In presenting this dissertation in partial fulfillment of the requirements for an advanced degree at Idaho State University, I agree that the Library shall make it freely available for inspection. I further state that permission to download, photocopy, and/or print my dissertation for scholarly purposes may be granted by the Dean of the Graduate School, Dean of my academic division, or by the University Librarian. It is understood that any copying or publication of this dissertation for financial gain shall not be allowed without my written permission.

Signature _____

Emerald D. Ryan

Date _____

Determination, Development, and Validation of a Fluid Height Analysis Method and Particle Spacing Protocol for the Smoothed Particle Hydrodynamic Code Neutrino

by

Emerald Dawn Ryan

A dissertation

submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy in Nuclear Science and Engineering

Idaho State University

Fall 2019

Copyright 2019 Emerald Dawn Ryan

To the Graduate Faculty:

The members of the committee appointed to examine the thesis of Emerald Ryan find it satisfactory and recommend that it be accepted.

_____

Dr. Chad L. Pope,

Major Advisor

_____

Dr. Mary Lou Dunzik-Gougar,

Committee Member

_____

Dr. Cristian Rabiti,

Committee Member

_____

Dr. Bruce Savage,

Committee Member

_____

Dr. S. Hossein Mousavinezhad,

Graduate Faculty Representative

To my mother, Debra Ryan, for always knowing I could achieve my goals and providing the support and encouragement for it to be possible

Acknowledgements

I have had the pleasure of working on the flooding risk analysis project for about 5 years. This project would not have been possible without Dr. Curtis Smith allowing ISU and myself to work in conjunction with him. Without his project, I would not have had the opportunity to learn about Smoothed Particle Hydrodynamics which resulted in my master's thesis and doctoral dissertation. I would also like to thank Steven Prescott and Michael Calley for being my mentor and manager during my INL Graduate Fellowship. They each provided guidance and support throughout this process and I have gained invaluable experience by being able to work with them.

Next, I would like to thank each of my committee members: Dr. Mary Lou Dunzik-Gougar for all of the support she has provided me throughout my time at ISU, Dr. Cristian Rabiti for providing help with RAVEN, Dr. Bruce Savage for providing guidance on fluid simulations, and Dr. S. Hossein Mousavinezhad for his support as my graduate faculty representative.

Lastly, I would like to thank Dr. Chad Pope for being my advisor and providing guidance throughout this entire process. I appreciate all the time he spent meeting with me to discuss my progress and changes to my dissertation. He provided excellent feedback, encouragement, and patience throughout my entire time at ISU and has helped me reach my goals.

Vita

Emerald Ryan was born on April 20, 1993 in Pocatello, ID. She received her undergraduate degrees in Nuclear Engineering and Mechanical Engineering with a minor in Applied Mathematics in 2015 from Idaho State University (ISU). She was awarded the 2016 Outstanding Student Achievement Award for the College of Science and Engineering – Engineering. While an undergraduate, she worked on several projects at ISU including the development of a GUI for a crystal furnace, development of a glovebox control system, development of a neutron unfolding suite, investigation of SPH codes, and modeling of riverine flooding using SPH. Upon finishing her undergraduate degrees, she was accepted into the master's program of Nuclear Science and Engineering at ISU. Her master's thesis was titled "Construction of a Smoothed Particle Hydrodynamic Model for Flow Over an Ogee Spillway Comparison to Determine Viability in Modeling Flooding Scenarios". She graduated with her M.S. degree in 2016 and immediately went to work on her Ph.D. in Nuclear Science and Engineering at ISU. She passed the Ph.D. qualifying exam in March 2017 and proposed her dissertation topic in March 2018. In the summer of 2018, Emerald was awarded an INL Graduate Fellowship which she started in August 2018. During her fellowship, she worked on validating and determining significant parameters for SPH simulations, helped in developing a fire simulation software to improve fire risk analysis, and continued work on her dissertation.

Table of Contents

List of Figures ix

List of Tables xiv

List of Acronyms xviii

Abstract xix

1 Introduction 1

2 Smoothed Particle Hydrodynamics 4

2.1 Theory 4

2.2 Neutrino 9

2.3 Particle Spacing Selection and Result Calculation 11

3 Literature Review 18

4 Importance of Particle Spacing 25

4.1 Neutrino and RAVEN Coupling 25

4.2 Sobol Index Calculation Using RAVEN 27

4.3 Example Calculation of Sobol Indices 29

5 Particle Spacing Convergence Optimizer 40

6 Optimum Particle Spacing Determination 55

6.1 Neutrino Models 56

6.2 Results 59

7 Sphere Packing Arrangement 64

7.1 Sphere Packing Analysis 66

8 Comparison Between Neutrino Results and Sphere Packing Results 71

9 Neutrino Fluid Height Bias 78

10 Improved Fluid Height Analysis Method 82

11 Validation of the Improved Fluid Height Analysis Method 94
12 Optimum Particle Spacing Determination Using the Improved Fluid Height Analysis Method 100
12.1 Adjustments 100
12.2 Results 101
12.3 Protocol Determination and Verification 105
13 Conclusion 117
14 References 119
Appendix A: RAVEN Files 126
Base Neutrino Coupling Interface 126
Neutrino/RAVEN Oscillating Tank Example Files 130
Appendix B: Convergence Optimizer Files 145
XML Editor Class 145
Command Prompt Class 146
Adjust Value Class 146
Output File Analysis Class 148
Analysis Class 149
Main File 149
Appendix C: Optimum Particle Spacing Details Using Neutrino Result 154
Appendix D: Sphere Packing Arrangement Analysis Script 171
Appendix E: Optimum Particle Spacing and Sphere Packing Arrangement Comparison Scripts 173
Appendix F: Improved Fluid Height Analysis Method Files 182

Per Particle Output File Example 182
2D Surface Profile Script 185
3D Surface Profile Script 187
Improved Average Height Script 188
3D Surface Profile Plotting Script 190
Appendix G: Spillway Files 192
Neutrino Height Results 192
Improved Height Results 227
Neutrino Velocity Results 262
Appendix H: Optimum Particle Spacing Details Using Improved Method 298
Appendix I: Particle Spacing Protocol Verification Test Details 314

List of Figures

Figure 1: Smoothing kernel and smoothing length [15]. 6

Figure 2: Neutrino GUI. 9

Figure 3: Particle spacing example: (left) 0.5 m spacing, (center) 0.25 m spacing, and (right) 0.125 m spacing. 12

Figure 4: Particle spacing example setup. 12

Figure 5: Particle spacing example with 0.25 m particle spacing and diameter. 13

Figure 6: Particle spacing example with 0.25 m particle spacing and point representation. 13

Figure 7: Particle spacing example with 0.125 m particle spacing and diameter. 14

Figure 8: Particle spacing example with 0.125 m particle spacing and point representation. 14

Figure 9: Particle spacing example with 0.0625 m particle spacing and diameter. 14

Figure 10: Particle spacing example with 0.0625 m particle spacing and point representation. 15

Figure 11: Particle spacing example with 0.03125 m particle spacing and diameter. 15

Figure 12: Particle spacing example with 0.03125 m particle spacing and point representation. 15

Figure 13: Scaled flooding experiment parametric study for different water depth gauges [19]. 19

Figure 14: Hypothetical flooding event parametric study [19]. 21

Figure 15: Neutrino/RAVEN coupling flow chart 26

Figure 16: Large-scale oscillating tank at the George Washington University. 30

Figure 17: Large-scale oscillating tank simulation setup. 31

Figure 18: SPH particle spacing convergence study [47]. 41

Figure 19: Hypothetical decreasing convergence simulation results. 42

Figure 20: Hypothetical increasing convergence simulation results. 42

Figure 21: Convergence optimizer flow chart. 47

Figure 22: Example converged study plot with converged result and optimum particle spacing. 49

Figure 23: Example optimizer result plot on the convergence study plot. 54

Figure 24: Model setup. 57

Figure 25: Small box fluid height vs. particle spacing plot using Neutrino result. 60

Figure 26: Medium box fluid height vs. particle spacing plot using Neutrino result. 60

Figure 27: Large box fluid height vs. particle spacing plot using Neutrino result. 61

Figure 28: Small box computational runtime vs. particle spacing plot using Neutrino result. 61

Figure 29: Medium box computational runtime vs. particle spacing plot using Neutrino result. 62

Figure 30: Large box computational runtime vs. particle spacing plot using Neutrino result. 62

Figure 31: Parametric study example. 63

Figure 32: Sphere arrangements [48]. 64

Figure 33: SC arrangement (left) and FCC arrangement (right) [50]. 65

Figure 34: Sphere packing analysis flow chart. 69

Figure 35: Sphere packing analysis example plot. 70

Figure 36: Small box optimum particle spacing and sphere packing arrangement comparison plot. 72
Figure 37: Medium box optimum particle spacing and sphere packing arrangement comparison plot. 73
Figure 38: Medium box optimum particle spacing and sphere packing arrangement comparison plot zoomed. 74
Figure 39: Large box optimum particle spacing and sphere packing arrangement comparison plot. 75
Figure 40: Small box simulation at the last time step. 77
Figure 41: Example of Neutrino height issue. 78
Figure 42: Particle position vs. Neutrino result comparison plot. 80
Figure 43: 2D improved fluid height analysis example. 83
Figure 44: 2D surface profile flow chart. 85
Figure 45: 3D surface profile flow chart. 86
Figure 46: Improved average height flow chart. 87
Figure 47: 2D surface profile using 1.5x particle spacing bin size. 88
Figure 48: 2D surface profile using 1x particle spacing bin size. 89
Figure 49: 2D surface profile using 0.5x particle spacing bin size. 89
Figure 50: 2D surface profile using 0.25x particle spacing bin size. 90
Figure 51: 3D surface profile using 2x particle spacing bin size. 91
Figure 52: 3D surface profile using 1.5x particle spacing bin size. 91
Figure 53: 3D surface profile using 1x particle spacing bin size. 92
Figure 54: 3D surface profile using 0.5x particle spacing bin size. 92

Figure 55: Neutrino spillway model. 96

Figure 56: Spillway total head vs. flow rate comparison plot. 98

Figure 57: Spillway relative error vs. flow rate comparison plot. 98

Figure 58: Small box fluid height vs. particle spacing plot using improved height result. 102

Figure 59: Medium box fluid height vs. particle spacing plot using improved height result. 103

Figure 60: Large box fluid height vs. particle spacing plot using improved height result. 103

Figure 61: Small box computational runtime vs. particle spacing plot using improved height result. 104

Figure 62: Medium box computational runtime vs. particle spacing plot using improved height result. 104

Figure 63: Large box computational runtime vs. particle spacing plot using improved height result. 105

Figure 64: Small test box fluid height vs. particle spacing plot using improved height result. 109

Figure 65: Medium test box fluid height vs. particle spacing plot using improved height result. 109

Figure 66: Large test box fluid height vs. particle spacing plot using improved height result. 110

Figure 67: Small test box computational runtime vs. particle spacing plot using improved height result. 110

Figure 68: Medium test box computational runtime vs. particle spacing plot using improved height result. 111
Figure 69: Large test box computational runtime vs. particle spacing plot using improved height result. 111
Figure 70: Small test box fluid height vs. particle spacing comparison with particle spacing protocol. 113
Figure 71: Medium test box fluid height vs. particle spacing comparison with particle spacing protocol. 113
Figure 72: Large test box fluid height vs. particle spacing comparison with particle spacing protocol. 114
Figure 73: Medium test box fluid height vs. particle spacing comparison with particle spacing protocol with adjusted particle emitter size. 115
Figure 74: Large test box fluid height vs. particle spacing comparison with particle spacing protocol with adjusted particle emitter size. 116

List of Tables

Table 1: Parameter Discussion..... 32

Table 2: Investigated Parameter Value Ranges 37

Table 3: Sobol Index Results for Oscillating Tank Simulation..... 39

Table 4: Convergence Optimizer Example Details 52

Table 5: Convergence Optimizer Example Particle Spacing Calculations..... 53

Table 6: Convergence Optimizer Example Convergence Calculations..... 53

Table 7: Model Details..... 58

Table 8: Model Convergence Optimizer Details 59

Table 9: Optimum Particle Spacing Using Neutrino Result..... 59

Table 10: Emitter Size Rounding of Number of Particles 67

Table 11: Emission Amount Rounding Details 68

Table 12: Spillway Results Comparison..... 97

Table 13: Relative Percent Improvement Using Improved Fluid Height Analysis Method 99

Table 14: Python File for Small Box Model..... 100

Table 15: Model Starting Particle Spacing for Improved Method Optimization 101

Table 16: Optimum Particle Spacing for Models Using Improved Method..... 102

Table 17: Model Smallest Dimension Details 105

Table 18: Model Particles Per Dimension Details..... 106

Table 19: Test Model Details..... 107

Table 20: Test Model Convergence Optimizer Details 108

Table 21: Optimum Particle Spacing and Protocol Particle Spacing for Test Model 108

Table 22: Particle Spacing Protocol Convergence Results for Test Models 112

Table 23: Particle Spacing Protocol Convergence Results for Test Models with Adjusted Particle Emitter Size 115

Table 24: Base Neutrino Coupling Interface 126

Table 25: Neutrino Interface for Oscillating Tank Example 130

Table 26: RAVEN Input File for Oscillating Tank Example 139

Table 27: RAVEN Output File for Oscillating Tank Simulation 143

Table 28: XML Editor Class 145

Table 29: Command Prompt Class 146

Table 30: Adjust Value Class 147

Table 31: Output File Analysis Class 148

Table 32: Analysis Class 149

Table 33: Convergence Optimizer Main File 150

Table 34: Small Box Convergence Optimizer Output File Details Using Neutrino Results 154

Table 35: Medium Box Convergence Optimizer Output File Details Using Neutrino Results 155

Table 36: Large Box Convergence Optimizer Output File Details Using Neutrino Results 155

Table 37: Small Box Printed Convergence Optimizer Information Using Neutrino Results 156

Table 38: Medium Box Printed Convergence Optimizer Information Using Neutrino Results 160

Table 39: Large Box Printed Convergence Optimizer Information Using Neutrino Results 165
Table 40: Sphere Packing Arrangement Analysis Script 171
Table 41: Small Box Optimum Particle Spacing and Sphere Packing Arrangement Comparison Script 173
Table 42: Medium Box Optimum Particle Spacing and Sphere Packing Arrangement Comparison Script 175
Table 43: Large Box Optimum Particle Spacing and Sphere Packing Arrangement Comparison Script 178
Table 44: Per Particle Output Example 182
Table 45: 2D Surface Profile Script 185
Table 46: 3D Surface Profile Script 187
Table 47: Improved Average Height Script 188
Table 48: 3D Surface Profile Plotting Script 190
Table 49: Neutrino Height Results for Spillway Comparison 192
Table 50: Improved Height Results for Spillway Comparison 227
Table 51: Neutrino Velocity Results for Spillway Comparison 262
Table 52: Small Box Convergence Optimizer Output File Details Using Improved Fluid Height Method 298
Table 53: Medium Box Convergence Optimizer Output File Details Using Improved Fluid Height Method 298
Table 54: Large Box Convergence Optimizer Output File Details Using Improved Fluid Height Method 299

Table 55: Small Box Printed Convergence Optimizer Information Using Improved Fluid Height Method 300

Table 56: Medium Box Printed Convergence Optimizer Information Using Improved Fluid Height Method 304

Table 57: Large Box Printed Convergence Optimizer Information Using Improved Fluid Height Method 308

Table 58: Small Test Box Convergence Optimizer Output File Details Using Improved Fluid Height Method 314

Table 59: Medium Test Box Convergence Optimizer Output File Details Using Improved Fluid Height Method 314

Table 60: Large Test Box Convergence Optimizer Output File Details Using Improved Fluid Height Method 315

Table 61: Small Test Box Printed Convergence Optimizer Information Using Improved Fluid Height Method 316

Table 62: Medium Test Box Printed Convergence Optimizer Information Using Improved Fluid Height Method 319

Table 63: Large Test Box Printed Convergence Optimizer Information Using Improved Fluid Height Method 324

List of Acronyms

2D	Two-dimensional
3D	Three-dimensional
CFEL	Component Flooding Evaluation Laboratory
CFL	Courant-Friedrichs-Lewy
CSV	Comma Separated Value
FCC	Face-Centered Cubic
GUI	Graphical User Interface
IISPH	Implicit Incompressible Smoothed Particle Hydrodynamics
ISU	Idaho State University
NPP	Nuclear Power Plant
RAVEN	Risk Analysis Virtual Environment
ROM	Reduced Order Model
SC	Simple Cubic
SESPH	State-Equation based Smoothed Particle Hydrodynamics
SPH	Smoothed Particle Hydrodynamics
STL	Stereolithography
UWRL	Utah Water Research Laboratory
WCSPH	Weakly Compressible Smoothed Particle Hydrodynamics
XML	Extensible Markup Language

Determination, Development, and Validation of a Fluid Height Analysis Method and Particle Spacing Protocol for the Smoothed Particle Hydrodynamic Code Neutrino

Dissertation Abstract – Idaho State University (2019)

Flooding is a concern to nuclear power plants and can cause extensive damage. Unfortunately, flooding cannot be prevented, but the damage can be mitigated. To mitigate the damage, potential flooding events can be simulated. One method for simulating fluid flow is Smoothed Particle Hydrodynamics (SPH). SPH is a particle-based method. However, one issue is that there is no guidance on particle spacing selection, which affects the resolution of the simulation. The selection of particle spacing for a simulation is typically chosen by a brief parametric study. While a parametric study provides some information, it does not guarantee that the optimum particle spacing is being used. To determine the optimum particle spacing, a convergence optimizer was developed in Python. The goal of the convergence optimizer was to determine the largest particle spacing that provided a converged result. A Neutrino model of fluid in a box was constructed with the fluid height analyzed using the convergence optimizer results as well as sphere packing arrangements. After analyzing the optimizer results and sphere packing results, it was determined that the Neutrino height method provides an assumption of a simple cubic arrangement resulting in overestimating the fluid height. An improved method for fluid height analysis was constructed in the form of a Python dynamic expression script and was validated using the flow over an ogee spillway comparison. The improved fluid height analysis method provides between a 15% to 43% improvement compared to the Neutrino results. Finally, the model was reanalyzed using the convergence optimizer and improved fluid height analysis method. From those results, a particle spacing protocol of 12 particles per the smallest dimension was determined and verified.

1 Introduction

Nuclear Power Plants (NPPs) are required to operate with minimal risk to the public's health and safety [1]. To categorize the risk of an initiating event, the consequence and the frequency of the event are used. However, one limiting issue with flooding initiating events related risk analysis is the lack of data. While some information is known, such as how long a component can operate until it fails under normal conditions, other information is not, such as how long a component can operate while submerged. Another example regarding component behavior during flooding conditions involves door performance. Most risk analysis suggests doors, including metal doors, will fail once the water level on the door reaches ~0.9 m. However, after conducting door failure experiments at Idaho State University (ISU), it was found that while simple hollow-core doors fail between 0.71 m and 0.99 m and metal doors fail between 0.84 m and 2.05 m [2]. These results shows that more information needs to be acquired so NPP flooding risk analysis can be improved.

Flooding is a concern for NPPs because it can cause extensive damage. The most widely known example of a devastating flood event is the Fukushima Daiichi accident that happened in March 2011. The accident was caused by a 14.5 m tsunami that inundated the north-east coast of Japan. The flooding from the tsunami disabled 12 of the 13 backup generators that were on-site as well as the ability to remove heat from the reactors [3]. This led to three reactor cores melting, radioactive material release to the environment, continued remediation efforts at the site, and increased public and industry concern regarding future flooding events.

Besides the Fukushima accident, there are many other NPP flooding events that have occurred. For example, on October 17, 1980, nearly 100,000 gallons of water collected in the containment building covering the lower 9 feet of the pressure vessel at Indian Point Unit 2 due to leakage from the system suppling water to air conditioners [4]. The surrounding water increased the thermal stresses on the pressure vessel which resulted in an increased chance of rupturing. Another example was at Calvert Cliffs where a winter storm dislodged a ventilation cover which allowed snow to enter an electrical cabinet [5]. As the snow melted, the water caused a short in the electrical cabinet causing both reactors to shut down.

Given the flooding concern in NPPs, improved characterization of the risk can be beneficial. There are two actions that can be performed to better understand the risk of flooding events. One action is to develop flooding failure data for components and the other action is to improve flooding simulations.

The action of obtaining flooding failure data for components is being pursued at ISU. ISU is in the process of developing the Component Flooding Evaluation Laboratory (CFEL). At CFEL, components will be flooded using different scenarios, such as rising water, spraying water, and eventually wave impact, to obtain data about when and how components fail. Experiments have already been done on small scale items, such as battery powered radios, and on full scale doors [2], [6]. In addition to conducting experiments, component fragility mathematical modeling is being developed for use in flood risk analysis. The results obtained from CFEL will be crucial for the second action.

The second action is to improve flooding simulations. Smoothed Particle Hydrodynamics (SPH) is a method for modeling fluid flow and is the main method being

investigated at ISU for flooding event simulations. The goal is to be able to either incorporate flooding failure data obtained from CFEL into an SPH code or to couple an SPH code with a risk assessment code that contains the flooding failure data. Incorporating SPH and flooding failure data would allow for flooding simulations at NPPs to be modeled before a flooding event occurs. These simulations would allow the NPP risk analyst to determine what potential damage would be done if the flooding event were to occur. Additionally, the analysis would help determine if certain actions need to be taken that could reduce the damage of the flooding event. For example, if a 14.5 m tsunami at the Fukushima Daiichi plant was simulated, the simulation might have showed that the backup generators would be impacted resulting in potential failure. Therefore, the plant could have relocated the generators or could have installed additional generators at an alternate location which might have resulted in a less devastating accident.

While simulating flooding events is the end goal, further work needs to be done in understanding very simple simulations so that complicated systems can be modeled correctly. A couple of the areas that need further research and understanding is SPH particle spacing and simulation result calculations. The following work presents a new method to more accurately calculate the fluid height as well as a particle spacing protocol for the SPH code Neutrino. The fluid height analysis method was validated and the particle spacing protocol was verified to show simulation improvement.

2 Smoothed Particle Hydrodynamics

2.1 Theory

Smoothed Particle Hydrodynamics is a particle-based method for obtaining approximate numerical solutions to the equations of fluid dynamics. SPH was originally developed for astrophysics applications in 1977 [7], [8]. The method was then extended to be used in fluid dynamics [9].

SPH uses the Lagrangian method to simulate flow as opposed to the Eulerian, grid-based, method such as Computational Fluid Dynamics (CFD). The Lagrangian method tracks a particle as it moves through the simulation. The Eulerian method determines properties at a location and is not specific to a particle. A simple example of the difference between Lagrangian and Eulerian methods is in bird migration. If one determines bird migration by counting the birds as they pass a location, then a Eulerian method is used. If instead, they place a radio transmitter on the birds, then a Lagrangian method is used [10].

SPH is an interpolation method meaning the property of one particle will depend on the surrounding particles. The interpolation technique is based on the interpolation integral, as shown in Eq. (1). However, since most properties are not known as a function, the interpolation integral must be rewritten into numerical form shown in Eq. (2). Additionally, Eqs. (3) and (4) show the gradient and Laplacian of the summation interpolant assuming the smoothing kernel is first and second order differentiable [11].

$$A_i(\boldsymbol{r}) = \int A(\boldsymbol{r}')W(\boldsymbol{r}-\boldsymbol{r}',h)d\boldsymbol{r}' \tag{1}$$

$$A_i(\boldsymbol{r}) = \sum_j A_j \frac{m_j}{\rho_j} W(\boldsymbol{r}-\boldsymbol{r}_j,h) \tag{2}$$

$$\nabla A_i(\boldsymbol{r}) = \sum_j A_j \frac{m_j}{\rho_j} \nabla W(\boldsymbol{r} - \boldsymbol{r}_j, h) \tag{3}$$

$$\nabla^2 A_i(\boldsymbol{r}) = \sum_j A_j \frac{m_j}{\rho_j} \nabla^2 W(\boldsymbol{r} - \boldsymbol{r}_j, h) \tag{4}$$

where A is the property of interest, i denotes the particle of interest, j denotes the surrounding particle, W is the smoothing kernel, r is the position of the particle, h is the smoothing length, m is the mass of the particle, and ρ is the density of the particle.

Smoothing kernels are weighting functions that determine how much a surrounding particle will affect the particle of interest. A particle closer to the particle of interest will have a larger influence than a particle far away. There are multiple different smoothing kernels each with their own equations. However, there are two properties a smoothing kernel must have [11]:

1. Kernel must be normalized: $\int W(\boldsymbol{r}, h) dr = 1$
2. Kernel must go to a Dirac function as h approaches 0: $\lim_{h \to 0} W(\boldsymbol{r}, h) = \delta(\boldsymbol{r})$

Besides these two properties, the smoothing kernel must also be positive and even to ensure it is an averaging function [12]. If the smoothing kernel is even and normalized, then the interpolation will have a second order accuracy or better. The accuracy is estimated by expanding A(**r**') in Eq. (1) using a Taylor series expansion about **r** and using the normalized kernel criterion to simplify the expansion result [13]. If the smoothing kernel is even, then the coefficients of the Taylor expansion are only non-zero for the even degree terms resulting in a second order accuracy or better [14].

The smoothing length specifies the distance away from the particle of interest where the particles affect the particle of interest. Therefore, a larger smoothing length means particles further away will influence the particle of interest. Figure 1 shows an example of the smoothing kernel, W, and smoothing length, h, on a set of particles.

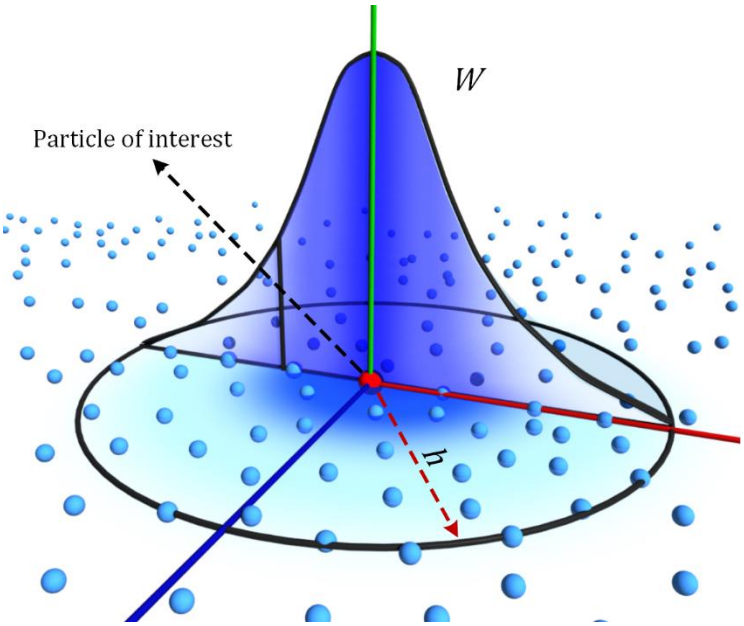


Figure 1: Smoothing kernel and smoothing length [15].

The smoothing kernel and smoothing length are then incorporated into the equations of motion. The momentum conservation equation calculates the change of momentum of the particle. It can account for pressure, gravity, viscosity, surface tension, and other forces. The momentum equation is shown in Eq. (5).

$$\frac{d\boldsymbol{v}_i}{dt} = -\sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2} + \Pi_{ij}\right) \nabla_i W_{ij} + \boldsymbol{g} \tag{5}$$

where **v** is the velocity vector, m is the mass, ρ is the density, P is the pressure, Π represents the viscosity term, and **g** is gravity.

Another equation of motion is the continuity equation which calculates the change in fluid density shown in Eq. (6) [16].

$$\frac{d\rho_i}{dt} = \sum_j m_j \boldsymbol{v}_{ij} \cdot \nabla_i W_{ij} \tag{6}$$

$$\boldsymbol{v}_{ij} = \boldsymbol{v}_i - \boldsymbol{v}_j \tag{7}$$

However, for incompressibility, the continuity equation must be equal to zero. The next equation of motion addresses moving the particles. Particles are moved using the XSPH variant [11]. The XSPH variant, where X is the unknown factor, ensures neighboring particles move with approximately the same velocity so particles do not occupy the same location [17]. The equation for moving particles is provided in Eq. (8).

$$\frac{d\boldsymbol{r}_i}{dt} = \boldsymbol{v}_i + \frac{1}{2}\sum_j \frac{m_j}{\rho_{ij}} \boldsymbol{v}_{ij} W_{ij} \tag{8}$$

$$\rho_{ij} = \frac{\rho_i + \rho_j}{2} \tag{9}$$

To move the particles as a function of time, a time stepping scheme is used. There are multiple ways of time stepping to determine the new property values of the particle. For example, the Euler-Cromer time stepping scheme equations, which are used in this work, are shown below.

$$\boldsymbol{v}_{t+\Delta t} = \boldsymbol{v}_t + \Delta t \boldsymbol{a}_t \tag{10}$$

$$\boldsymbol{r}_{t+\Delta t} = \boldsymbol{r}_t + \Delta t \boldsymbol{v}_{t+\Delta t} \tag{11}$$

where **r** is the position vector, **v** is the velocity vector, **a** is the acceleration vector, t is the time step, and Δt is the time step distance.

Lastly, compressibility is an important issue to address when dealing with fluids. There are multiple methods that can be used to account for the compressibility of a fluid. For example, two compressibility methods are the Implicit Incompressible SPH (IISPH) method and state-equation based SPH (SESPH) method. The SESPH method is a compressible approach employed using an equation of state such as the Tait equation. The IISPH method determines pressure forces by iteratively solving a pressure Poisson equation until the desired compressibility is reached [9]. For this work, the IISPH method was used since water can be considered incompressible. The IISPH method decouples the velocity and pressure by splitting the resolution of the momentum equation [9]. The decoupling is done by solving intermediate velocities using the momentum equation with only non-pressure force terms, shown in Eq. (12) and solving intermediate densities using the continuity equation shown in Eq. (13).

$$\boldsymbol{v}_i^* = \boldsymbol{v}_i(t) + \Delta t(-\sum_j m_j \Pi_{ij} \nabla W_{ij}(t) + \boldsymbol{g}) \tag{12}$$

$$\rho_i^* = \rho_i(t) + \Delta t \sum_j m_j(\boldsymbol{v}_i^* - \boldsymbol{v}_j^*) \cdot \nabla W_{ij}(t) \tag{13}$$

The Poisson equation is determined by combining the momentum equation with only the pressure force term, the mass equation, and the incompressibility condition [9]. Eq. (14) shows the Poisson equation.

$$\nabla^2 p_i(t) = \frac{\rho_0 - \rho_i^*}{\Delta t^2} \tag{14}$$

The Poisson equation is reformulated and iterated until the relative density error is below the specified threshold value. Finally, the new velocities and positions are computed by taking into account the pressure force term and using the Euler-Cromer time stepping scheme [9].

2.2 Neutrino

Neutrino is a general-purpose simulation and visualization environment developed by Neutrino Dynamics Initiative which includes an SPH solver [15]. Neutrino is being used for a variety of applications including risk assessment and mitigation, hydro-fracturing, energy research, and environmental sciences [15]. Neutrino is a proprietary, developmental code but is available to universities for research.

Neutrino has a graphical user interface (GUI) that allows for creation of models. Figure 2 shows the Neutrino GUI.

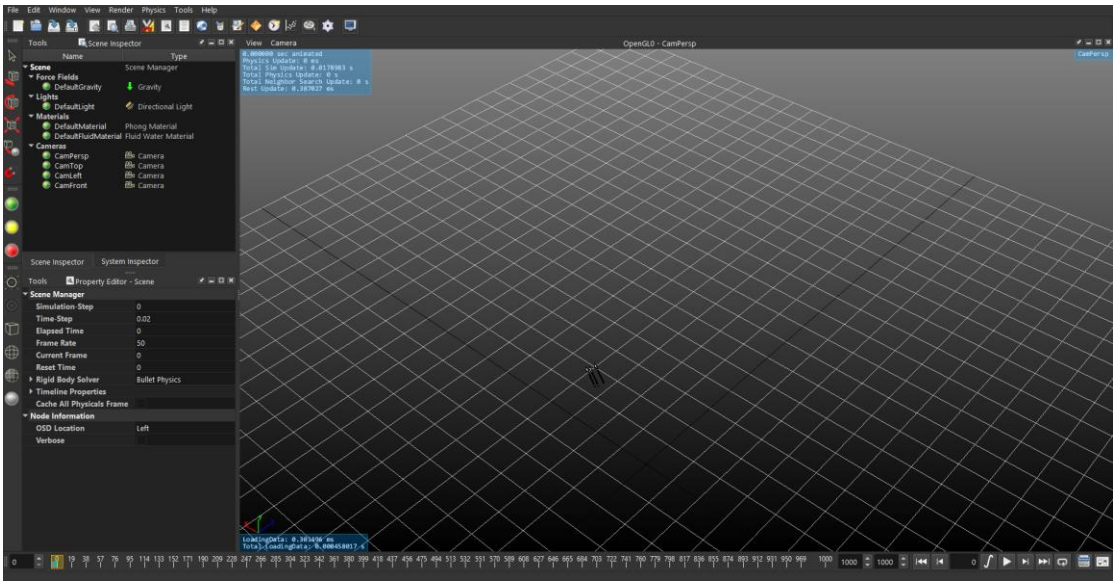


Figure 2: Neutrino GUI.

To create a simulation, the user must add the required rigid bodies, particle emitters, and the SPH solver. Neutrino has multiple native rigid bodies including boxes, cuboids, planes, cylinders, and spheres. It also allows for custom rigid bodies to be imported by using the Stereolithography (STL) file of the custom geometry. Neutrino also has different particle emitters including a square emitter, circle emitter, flow emitter, and block emitter and it has different SPH solvers including an IISPH solver and Weakly Compressible SPH (WCSPH) solver.

There are additional features in Neutrino that can be used depending on the scenario being modeled. Some of these features include measurement fields, particle killers, visualization tools, and custom Python dynamic expression scripts for specific components such as rigid bodies.

Once the input file is created, the user can simulate the scenario. After the scenario is simulated, the user can then play the simulation back to visualize the result, record results from the measurement fields, and create an animation of the result.

Besides using the GUI to create the input file, the input file can also be created or modified using a text editor since the Neutrino input file is an Extensible Markup Language (XML) file. The user can also run the Neutrino input file using the command prompt rather than the GUI. The XML input file and command prompt execution allows for the ease of modification and execution of simulations.

For this work, two version of Neutrino were used. The Neutrino version that was used for the initial optimum particle spacing work was released on November 05, 2018. The Neutrino version that was used for the rest of the work was released on

February 22, 2019. Two different versions of Neutrino were used because the initial optimum particle spacing work was started first and the older Neutrino version worked for the simulations being conducted. However, there was an issue with adding dynamic expression scripts in Neutrino through the command prompt which required using an updated Neutrino version that had the issue fixed.

2.3 Particle Spacing Selection and Result Calculation

While SPH is a useful method for simulating fluid flow, one major weakness of SPH is that there is no go-to method for determining the appropriate particle spacing. This is an issue because by choosing the wrong particle spacing there is a chance the simulation will provide unrealistic results. The following section provides additional detail on particle spacing.

Particle spacing represents the distance between the points that will be tracked throughout the simulation. Another way to think of particle spacing is to think of it as the particle diameter/size. Therefore, if the particle spacing is 0.5 m, then the particle diameter is 0.5 m. Additionally, the particle spacing also effects the number of particles. As the particle spacing decreases, then the number of particles increases.

To better understand particle spacing, suppose there is a 1 m by 1 m by 1 m box full of fluid. If the particle spacing is 0.5 m, this means the particle diameter is 0.5 m resulting in 8 particles in the box. If the particle spacing is 0.25 m, this means the particle diameter is 0.25 m resulting in 64 particles in the box. Lastly, if the particle spacing is 0.125 m, this means the particle diameter is 0.125 m resulting in 512 particles in the box. Figure 3 shows these different particles spacings in a 1 m by 1 m by 1 m box.

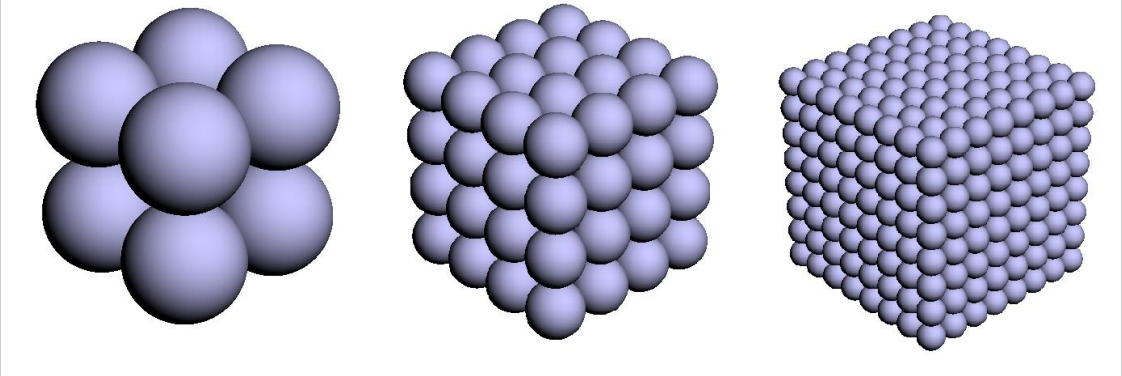


Figure 3: Particle spacing example: (left) 0.5 m spacing, (center) 0.25 m spacing, and (right) 0.125 m spacing.

Based on the figure above, the smaller the particle spacing becomes, the resolution of the fluid in the box increases. This shows that a certain particle spacing is needed to provide an accurate resolution of the situation be modeled.

To show how particle spacing affects the results of the simulation, a basic Neutrino model was created. The following example consists of a rigid box, a rigid cuboid, and a block of fluid. Figure 4 shows the setup of the example.

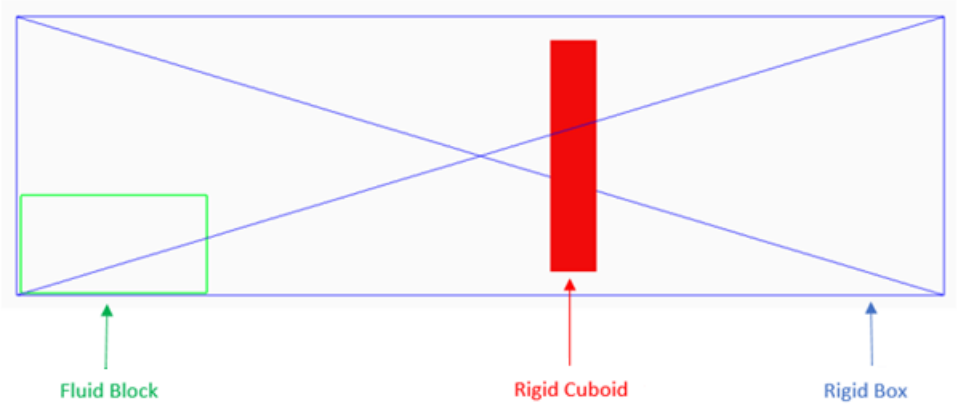


Figure 4: Particle spacing example setup.

For this setup, the fluid should collapse down due to gravity from the fluid block form and then flow underneath the rigid cuboid which has a gap of 0.25 m. However,

depending on the particle spacing, results may be different. For example, the following model was simulated using different particle spacings to show how the results change. Each simulation was simulated for 100 timesteps or for 2 seconds of real time. Figure 5 through Figure 12 shows the result for different particle spacings with the particles being rendered based on the particle spacing as the particle diameter and as points.

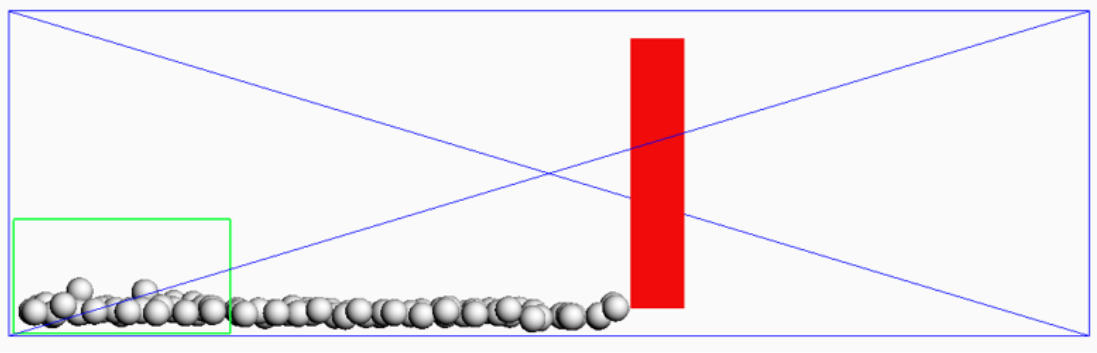


Figure 5: Particle spacing example with 0.25 m particle spacing and diameter.

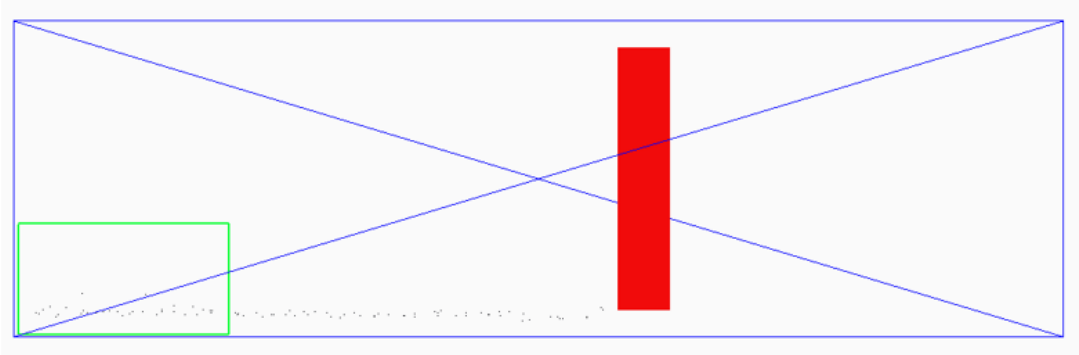


Figure 6: Particle spacing example with 0.25 m particle spacing and point representation.

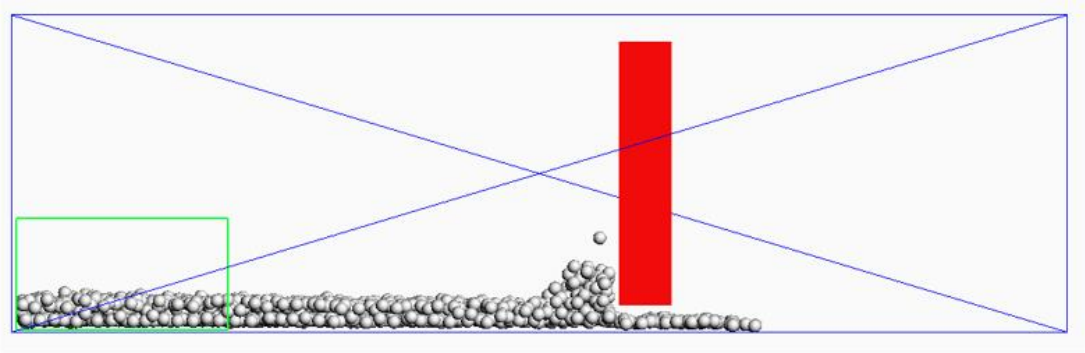


Figure 7: Particle spacing example with 0.125 m particle spacing and diameter.

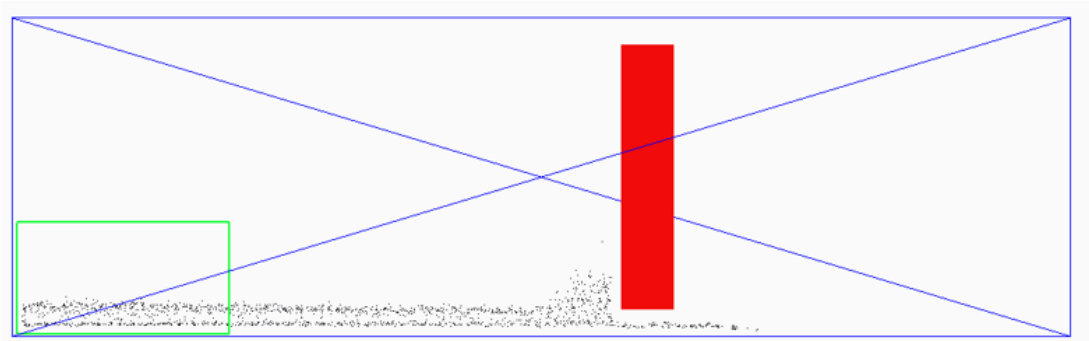


Figure 8: Particle spacing example with 0.125 m particle spacing and point representation.

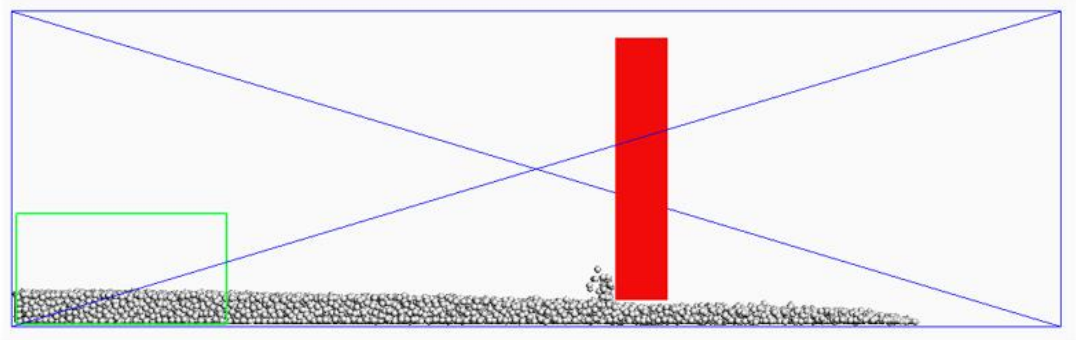


Figure 9: Particle spacing example with 0.0625 m particle spacing and diameter.

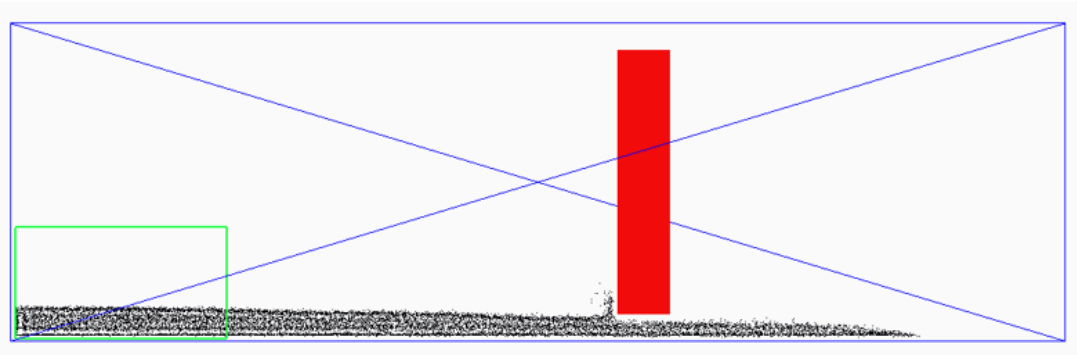


Figure 10: Particle spacing example with 0.0625 m particle spacing and point representation.

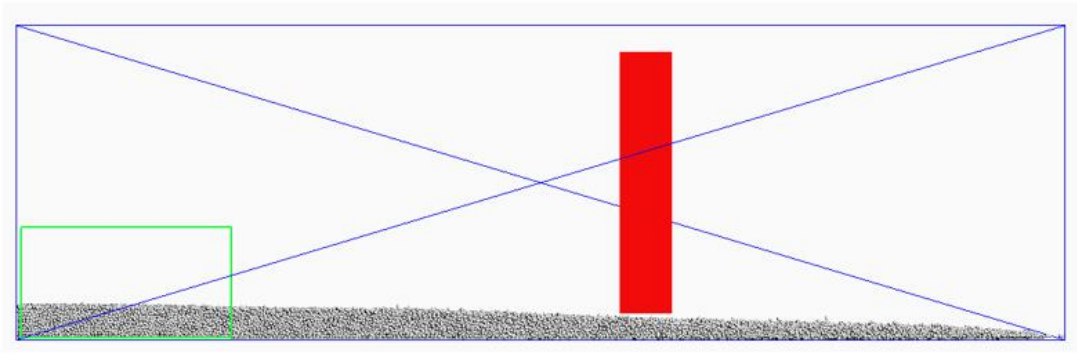


Figure 11: Particle spacing example with 0.03125 m particle spacing and diameter.

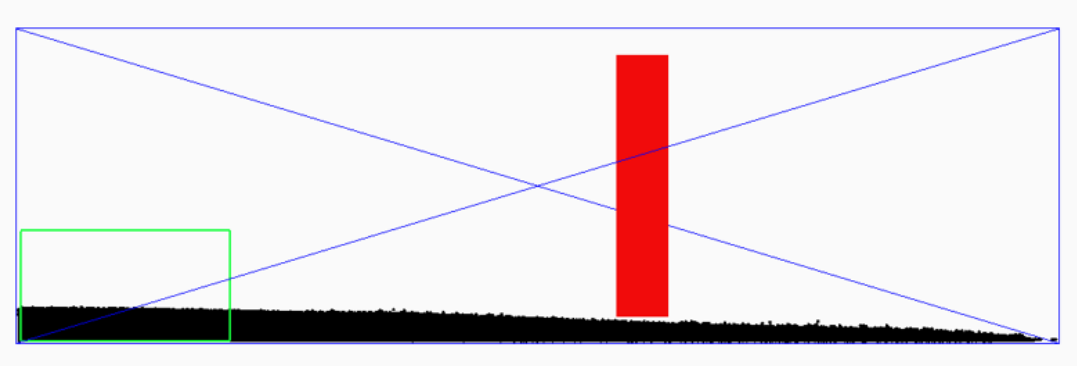


Figure 12: Particle spacing example with 0.03125 m particle spacing and point representation.

Based on the above results, each particle spacing provided a different result. The 0.25 m particle spacing did not allow any particle to flow underneath the rigid cuboid but

provided the quickest runtime of 1.36 seconds. The 0.125 m particle spacing allowed some particles to flow underneath the rigid cuboid, but only one row of particles could fit underneath. The 0.0625 m spacing allowed for multiple rows of particles to fit underneath the rigid cuboid, but some particles are still stopped by the rigid cuboid. Finally, the 0.03125 m particle spacing allows particles to flow underneath the rigid cuboid without interacting with the rigid cuboid but provided the longest runtime of 9.49 minutes.

The above results do not provide any guidance on what particle spacing should be used since each one provided a different result. Since the smaller particle spacing provides a more realistic simulation, it could be assumed the particle spacing of 0.03125 m should be used. However, there is a chance an even smaller particle spacing would provide a different result from the 0.03125 m particle spacing or a slightly larger particle spacing would provide the same result while decreasing the runtime. Therefore, a detailed parametric study is needed to determine the acceptable particle spacing for a simulation.

However, a parametric study of particle spacing does not guarantee the appropriate spacing will be selected. This is because the user must decide what particle spacings to try and when to stop the parametric study. Another issue is the fact that parametric studies take computational time as well as storage space. Therefore, if users have a deadline for results, there might not be enough time to conduct a thorough parametric study.

Another issue, particularly with proprietary codes, is not knowing how the code performs calculations and obtains results. Proprietary codes tend to only provide broad details about how the code works so that other code developers cannot duplicate their ideas. Typically, proprietary codes will provide extensive validation and verification examples to assure the users that the code is reliable. However, for a developmental, proprietary code,

there is no confirmation to the user that the code provides accurate results. Therefore, without the validation and verification, the lack of information regarding result calculation becomes troublesome and requires the user to either blindly trust the code or to extensively investigate the code to see what conclusions can be made.

3 Literature Review

A literature review was conducted to determine if there is a protocol for selecting particle spacing. From the articles investigated, there is not a specific protocol for selecting the particle spacing. Most of the articles rely on parametric or rudimentary convergence studies to select the particle spacing or number of particles for their specific simulation.

For example, the article *Two-phase smooth particle hydrodynamics modeling of air-water interface in aerated flows* by Yang HuiXia, et. al. (2017) conducted three case studies to determine the performance of their two-phase SPH model [18]. For the case studies, a parametric study was conducted to determine the particle spacing required to provide results close to experimental data but be computationally efficient. Based on their results, the smallest particle spacing provided results closest to the experimental results but had the highest computational time. For the rest of the simulations, they used the largest particle spacing because it provided results within their precision requirements and it had the quickest runtime [18]. However, if they would have had a protocol to select the particle spacing, then they would not have had to spend over 9 hours of computational runtime to determine an acceptable particle spacing value.

Another article, *Efficient urban flood simulation using a GPU-accelerated SPH model* by Qiuhua Liang, et. al. (2015) developed an SPH model using GPUs to efficiently model urban flooding events. The model was then tested by conducting a scaled flooding experiment and then a hypothetical flood [19]. For both events, a parametric study was conducted to determine the appropriate particle spacing. Figure 13 shows the parametric study results for the scaled flooding experiment. Figure 14 shows the parametric study results for the hypothetical flooding event where (a) shows the results with a particle

spacing of 20 m, (b) shows the results with a particle spacing of 10 m, and (c) shows the results with a particle spacing of 5 m. The fluid particles are shown in yellow while the blue background represents the buildings and roads of the location.

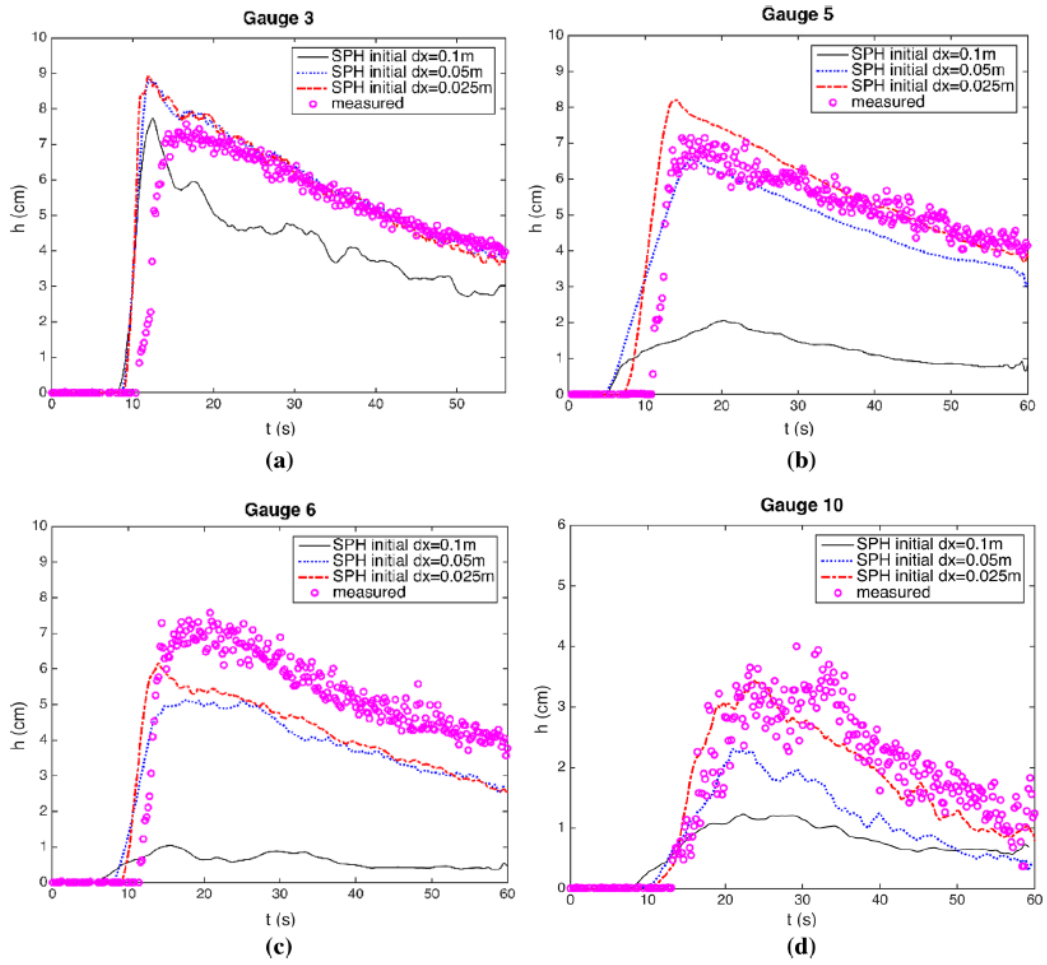
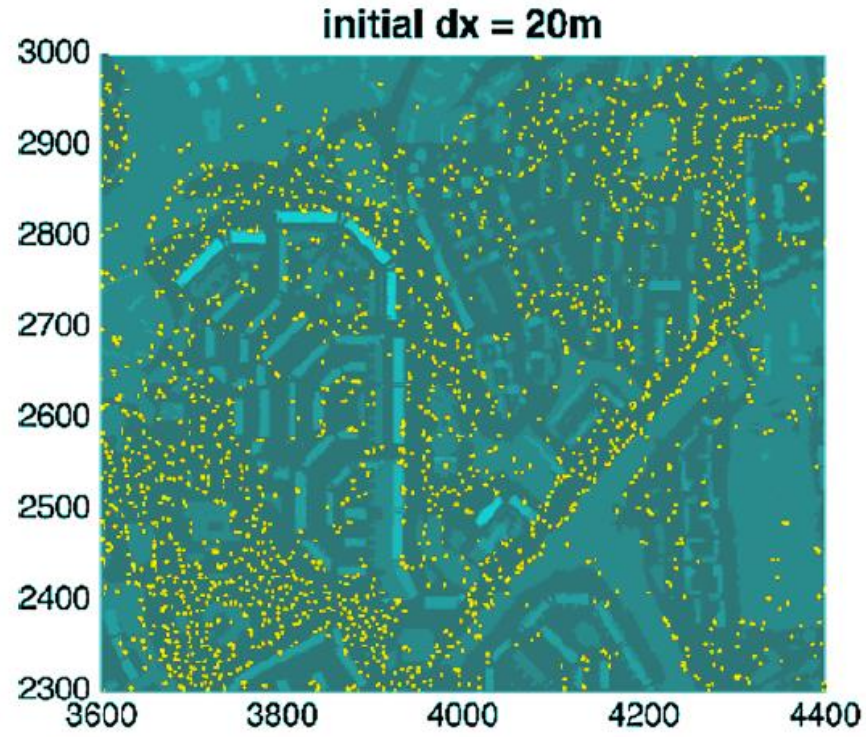
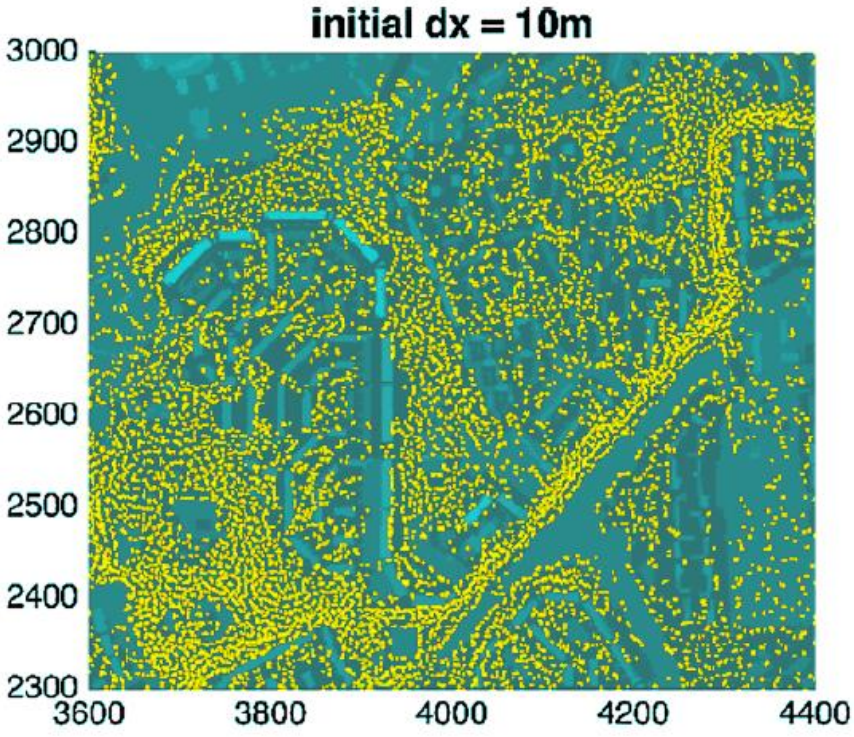


Figure 13: Scaled flooding experiment parametric study for different water depth gauges [19].


initial dx = 20m
3000
2900
2800
2700
2600
2500
2400
2300
3600
3800
4000
4200
4400

(a)


initial dx = 10m
3000
2900
2800
2700
2600
2500
2400
2300
3600
3800
4000
4200
4400

(b)

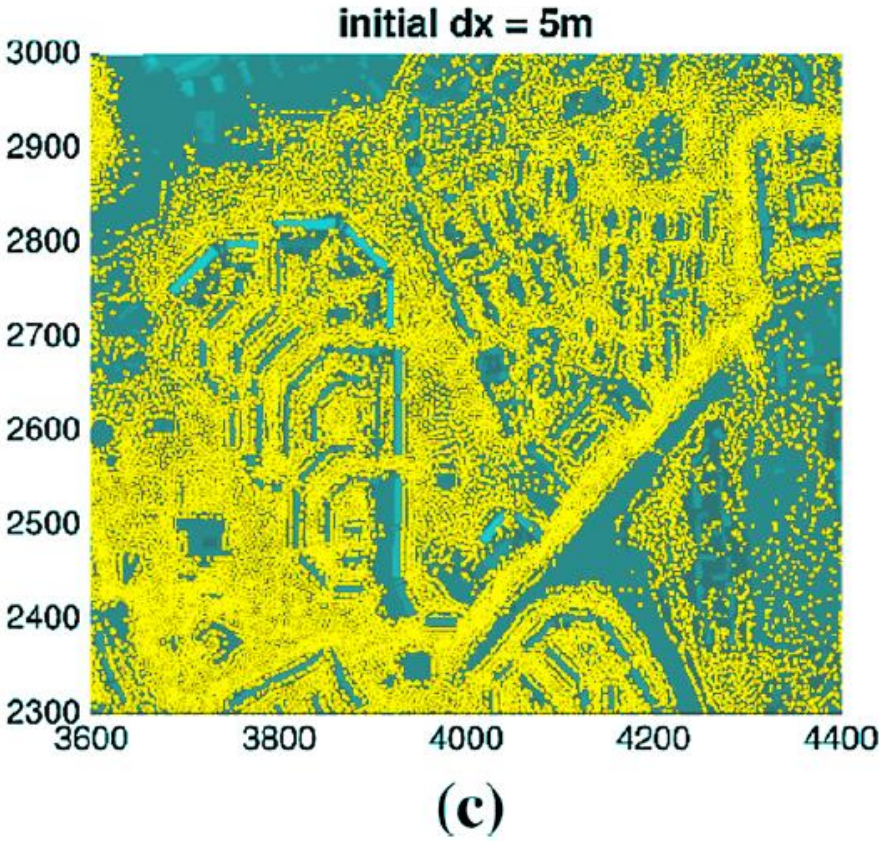


Figure 14: Hypothetical flooding event parametric study [19].

Based on their results, the smallest particle spacing was chosen for both flooding event simulations because it provided results closer to experimental data as well as provided a higher resolution. Additionally, they stated high resolution simulations are required to obtain accurate results when simulating flooding events and lower resolution simulations tend to give unacceptable results [19]. For both simulated flooding events, over 245 hours were spent conducting the parametric studies on a CPU system and around 140 hours on a GPU system [19].

The following are additional publications that have required a parametric study to determine the particle spacing:

- *GPUs, a New Tool of Acceleration in CFD: Efficiency and Reliability on Smoothed Particle Hydrodynamics Methods* by Alejandro Crespo, et. al. (2011): parametric study on number of particles for dam break flow impacting an obstacle [20].
- *Three-Gorges Dam Fine Sediment Pollutant Transport: Turbulence SPH Model Simulation of Multi-Fluid Flows* by J. H. Pu, et. al (2016): convergence analysis on multi-fluid flows [21].
- *Numerical simulation of ball-end milling with SPH method* by S. Ghafarizadeh, et. al. (2017): convergence study on particle spacing for cutting forces in ball-end milling process, 1,155 hours of computation [22].
- *Time domain simulation of sound waves using smoothed particle hydrodynamics algorithm with artificial viscosity* by Xu Li, Tao Zhang, and Yong Ou Zhang (2015): convergence study on particle spacing for sound propagation [23].
- *A 3-D SPH model for simulating water flooding of a damaged floating structure* by Kai Guo, et. al. (2016): convergence study on particle spacing for simulation of damaged and non-damaged floating object [24].
- *Comparison of free surface flow measurements and smoothed particle hydrodynamic simulation for potential nuclear power plant flooding simulation* by Emerald D. Ryan, Bruce M. Savage, Curtis L. Smith, and Chad L. Pope (2019): parametric study on particle diameter for ogee spillway comparison, 263 hours of computation [25].
- *Numerical modeling of Kelvin-Helmholtz instability using smoothed particle hydrodynamics* by M. S. Shadloo and M. Yildiz (2011): convergence study on number of particles for simulating Kelvin-Helmholtz instability [26].

- *SPH for 3D floating bodies using variable mass particle distribution* by Pourya Omidvar, Peter K. Stansby, and Benedict D. Rogers (2012): parametric study between coarse and fine particle spacing for different experiments, 21.5 days of computation [27].
- *Smoothed Particle Hydrodynamics for coastal engineering problems* by A. Barreiro, A.J.C. Crespo, J.M. Domínguez, and M. Gómez-Gesteira (2013): parametric study on particle resolution for wave-structure interaction [28].
- *Numerical simulation of shallow-water dam break flows in open channels using smoothed particle hydrodynamics* by Tsang-Jung Chang, Hong-Ming Kao, Kao-Hua Chang, and Ming-His Hsu (2011): parametric study on particle spacing with accuracy prediction showing the smallest spacing was the most accurate but with a high computational time [29].
- *Numerical modeling of dambreak-induced flood and inundation using smoothed particle hydrodynamics* by Hong-Ming Kao and Tsang-Jung Chang (2012): parametric study on number of particles for dam break flows through a dry flat floodplain which shows the L_2 error reduces with more particles but increases the computational runtime [30].
- *A treatment of zero-energy modes in the smoothed particle hydrodynamics method* by R. Vignjevic, J. Campbell, and L. Libersky (2000): parametric study on number of particles showing the error in kinetic energy is reduced as the number of particles increase [31].
- *Applicability of Smoothed Particle Hydrodynamics for estimation of sea wave impact on coastal structures* by Corrado Altomare, et. al. (2015): parametric study

on particle resolution stating that the coarser resolution was preferred since it provided similar results with a quicker computational runtime [32].

The above publications show there is a need for a protocol for selecting an appropriate particle spacing. Days of computational time were required for many of the convergence studies performed. However, with their convergence studies, there was no guarantee convergence was actually reached. Therefore, with a protocol for selecting particle spacing, days of computational runtime could be avoided and more time could be spent on the actual SPH simulation.

Since Neutrino is a proprietary code, there is limited information regarding the code's calculation methodology. Additionally, Neutrino has limited verification and validation results since it is developmental. After reviewing the Neutrino documentation, it was also found that there is no information regarding how Neutrino calculates results. For example, the Neutrino documentation does not state how the fluid height, pressure, etc. is calculated. The lack of this information regarding result calculations causes skepticism in the simulation results. Therefore, further research needs to be done to increase the reliability of Neutrino.

4 Importance of Particle Spacing

While particle spacing seems to be an important parameter in regards to SPH simulations, it would be beneficial to have quantitative proof. To provide quantitative proof of particle spacing importance, the Risk Analysis Virtual Environment (RAVEN) software was used. RAVEN is an INL developed software tool that is a modular component that can be coupled with other systems [33]. It is capable of running external codes and performing parametric and probabilistic analysis based on the results of the external codes [33]. RAVEN is written in Python and has a wide range of capabilities. Some of the capabilities of RAVEN include creation and use of reduced order models, forward propagation of uncertainties, advanced sampling methods, data post-processing, model parameter optimization, and data management [33]. Additionally, RAVEN provides the framework for users to add new capabilities that will fit their needs.

4.1 Neutrino and RAVEN Coupling

To make use of all of RAVEN's capabilities and provide a more efficient approach to conducting multiple simulation, the SPH code Neutrino was coupled with RAVEN [34]. The Neutrino/RAVEN coupling was achieved by developing a Neutrino interface so RAVEN can run Neutrino. The Neutrino interface is a Python class consisting of five functions. Two of the functions are used to check the input file type, one function creates new Neutrino input files, one function generates the command to run Neutrino, and one function moves and converts the Neutrino output so RAVEN can interpret the results. The *create new input* function tells RAVEN to create new Neutrino input files with new sampled values for the parameters of interest and changes the directories to where the results are stored. The *generate command* function tells RAVEN where to find the Neutrino

executable and then how and what Neutrino input file to run. The *finalize code output* function edits and moves the Neutrino results so RAVEN can find the results and interpret them. Figure 15 shows a flow chart of the Neutrino/RAVEN coupling. For the base Neutrino/RAVEN coupling interface, please refer to Appendix A: RAVEN Files.

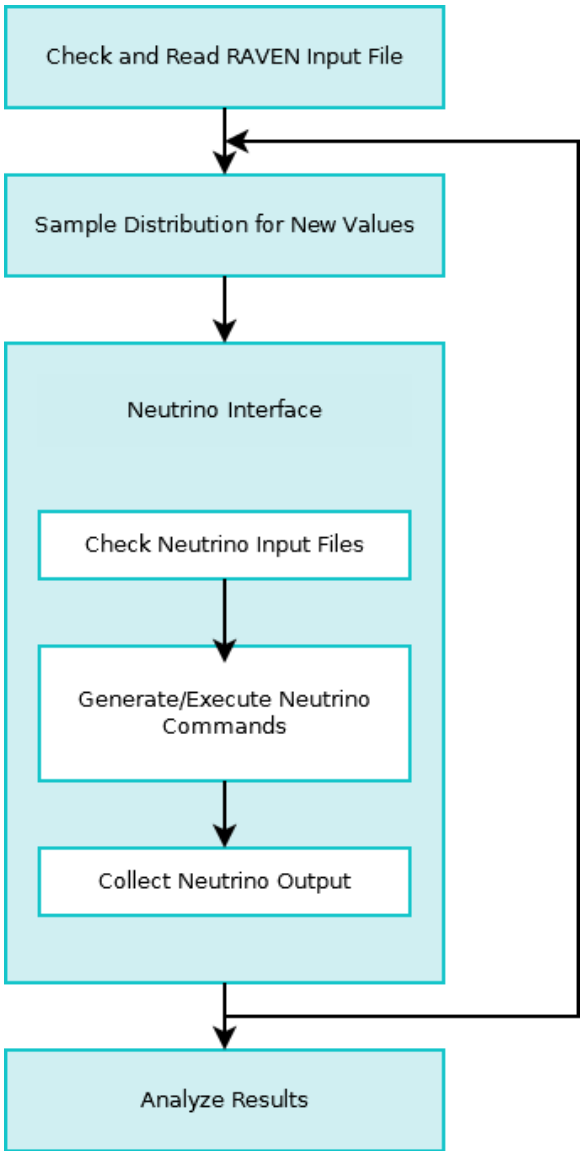


Figure 15: Neutrino/RAVEN coupling flow chart.

4.2 Sobol Index Calculation Using RAVEN

To determine parameter importance, a sensitivity analysis is typically performed. A sensitivity analysis identifies what parameters have the greatest effect on the model output [35]. The two types of sensitivity analysis are local and global. A local sensitivity analysis determines the change in the model output with respect to changes in one input parameter at a time while a global sensitivity analysis varies all input parameters simultaneously over the entire parameter space [35]. Since it is time saving to vary all input parameters simultaneously, a global sensitivity analysis was chosen.

There are several methods of global sensitivity analysis, but the one used for this analysis is a variance decomposition method. Variance based methods depend on estimating the quantity shown in Eq. (15).

$$\frac{Var_{X_i}[E(Y|X_i)]}{Var(Y)} \tag{15}$$

where Y is the output, X_i is an input parameter, $E(Y|X_i)$ is the expectation of output Y given the value of X_i, and Var denotes the variance [36]. The specific variance decomposition method used is the Sobol method. It determines the contribution of each input parameter and their interactions to the overall model output variance by decomposing the model output function into summations of variance using combinations of input parameters in increasing dimensionality [35], [36]. This is shown in Eq. (16).

$$f(x) = f_0 + \sum_{i=1}^{k} f_i(x_i) + \sum_{i=1}^{k}\sum_{j\neq i}^{k} f_{ij}(x_i, x_j) + f_{1,2,\dots,k}(x_1, \dots, x_k) \tag{16}$$

where f(x) is the output function, k is the number of input parameters, f_0 is the mean of the output function, $f_i(x_i)$ is a function of x_i, $f_{ij}(x_i, x_j)$ is a function of x_i and x_j, and so on [36]. Using this decomposition, the mean, f_0, is calculated using Eq. (17), the total variance is calculated using Eq. (18), the partial variance is calculated using Eq. (19), and the first-order Sobol index is calculated using Eq. (20).

$$f_0 = \int f(x)dx \approx \frac{1}{k}\sum_{i=1}^{k} f(x_i) \tag{17}$$

$$D = \int f^2(x)dx - f_0^2 \approx \sum_{i=1}^{k} D_i + \sum_{i<j} D_{ij} + \sum_{i<j<l} D_{ijl} + \cdots + D_{1,2,\ldots,k} \tag{18}$$

$$D_{i_1 \ldots i_s} = \int f^2_{i_1 \ldots i_s}(x_{i1}, \ldots, x_{is})dx_{i1}, \ldots, x_{is} \tag{19}$$

$$S_i = \frac{D_i}{D} \tag{20}$$

where D is the total variance, $D_{i_1 \cdots i_s}$ is the partial variance, s is the order contribution from the input parameter x_i to the output variance with s equal to one being first-order, s equal to two being second-order, etc., D_i being the first-order variance of the i^{th} input parameter, and S_i is the first-order Sobol sensitivity index for the i^{th} input parameter. The first-order Sobol indices determine the effect of a single parameter on the model output variance. If an input parameter has a large Sobol index, then it means that parameter greatly effects the output variance resulting in it being important.

While calculating the Sobol indices when the output function is known is straightforward, it becomes more complicated if your output function is a black box, such as with a Neutrino simulation. The output function is considered a black box because the input parameters are known as well as the output value, but the function to get the output is unknown.

To determine the Sobol indices for an unknown output function, there are different methods. One method is to use a Monte Carlo integration. However, the issue with this approach is that it requires, at a minimum, 1,000 times the number of input parameter sample points to calculate the Sobol indices [37]. Therefore, if there are two input parameters, then at least 2,000 sample points would be needed. This is an issue because performing thousands of runs of a single simulation would be very computationally expensive. Another approach is needed to calculate the Sobol indices for SPH simulations.

Since Neutrino and RAVEN were coupled, the capabilities of RAVEN were investigated. One method to obtain the Sobol indices with RAVEN is by using the Sparse Grid Collocation sampler and the GaussPolynomialRom reduced order model (ROM). The Sparse Grid Collocation sampler selects evaluation points based on characteristic quadratures as part of stochastic collocation for generalized polynomial chaos uncertainty quantification [33]. The GuassPolynomialRom is a ROM based on the characteristic Gaussian polynomial fitting scheme of generalized polynomial chaos expansion [33]. For more information on Sparse Grid Collocation and GaussPolynomialRom, please refer to these references [38], [39], [40], [41], [42], [43], [44].

4.3 Example Calculation of Sobol Indices

Once a method for calculating the Sobol indices for an SPH simulation was identified, an example calculation was performed to quantify the importance of particle spacing. The example used was an oscillating tank experiment.

A large-scale oscillating tank was designed and constructed at the George Washington University. The tank measures 5.951 m long by 1.2 m high by 2.468 m wide

[45]. The tank is constructed of a steel frame and acrylic walls and bottom. The tank is oscillated through a sine-forcing function using a hydraulic actuator capable of amplitudes up to 0.25 m and velocities up to 0.5 m/s [45]. Additionally, a pressure transducer is located at the end-wall of the tank, 0.1016 m above the tank bottom, to measure the pressure from wave impacts. The experiment for the end-wall pressure measurement consisted of a water depth of 0.1524 m. The forcing function had a 0.1016 m amplitude, with a 0.11 Hz frequency, and ran for 60 cycles. Figure 16 shows the large-scale oscillating tank at the George Washington University.



Figure 16: Large-scale oscillating tank at the George Washington University.

The Neutrino model was constructed to match the experimental setup as closely as possible. The oscillating tank experiment can be characterized as a two-dimensional (2D) experiment, resulting in the simulation tank having the same length and height dimensions, but a smaller width of 0.2 m. By reducing the width of the simulation tank, the computational runtime of the simulation is also reduced without compromising accuracy since the width was selected large enough to avoid edge effects due to the boundary conditions [25].

The Neutrino simulation consisted of a rigid box to act as the simulation tank, two block particle emitters to fill the tank with the correct number of particles, and one measurement field. The measurement field was used to measure the pressure on the end-wall. This measurement field compares to the pressure transducer used in the experimental setup. Figure 17 shows the Neutrino simulation setup for the oscillating tank.

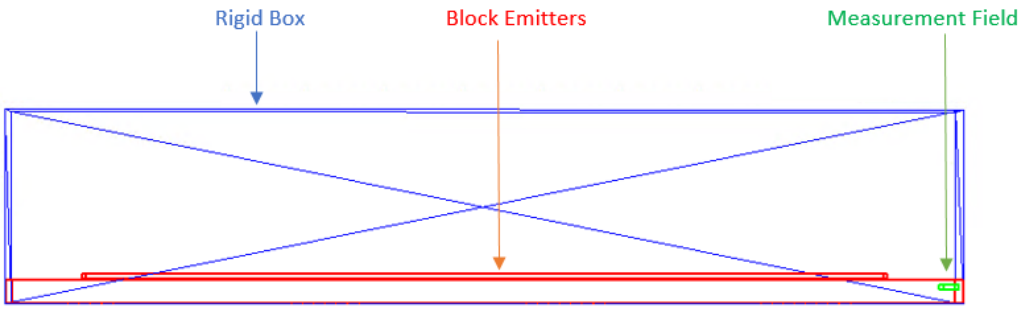


Figure 17: Large-scale oscillating tank simulation setup.

Once the simulation setup was complete, the simulation tank and end-wall measurement field needed to be oscillated at the same forcing function as the experiment. To create these oscillations, a Python script for each object was created and added to the position of the object as a dynamic expression. The Python scripts adjust the position of the items based on Eq. (21).

$$z(t) = A\sin(2\pi f t) \tag{21}$$

where z is the new position, t is the time, A is the amplitude, and f is the frequency. The amplitude and frequency values are set in the Python script to match the experimental values, and the time is extracted from the simulation. The equation allows for the movement of each object to be continuous and evaluated for every time step of the simulation. Without using the equation and, instead, using a dataset of positions, the

movement of the objects may not shift smoothly with the time step, causing jumps with overlapping particles, which results in erratic or explosion-like behavior from the simulation tank impact [46].

Since there are more than 30 parameters associated with the simulation and experimental measurement uncertainty, these parameters were investigated to determine their applicability to this experiment. Table 1 shows a list of the parameters and an initial discussion on whether the parameter was investigated to determine its significance [46]. Highlights show the parameters that were selected for investigated.

Table 1: Parameter Discussion

Parameters	**Importance Discussion**
Time step pressure iteration coefficient	Important, but not investigated because it is expected to be soon deprecated after replacing the pressure solver
Non-static reference frame	Not important because it only affects simulations with fast-moving dynamic solid objects
Stop criterion	Important, but not investigated because the default value is known to be optimal
Stop threshold	Important because it affects the level of incompressibility enforcement. It is used in the IISPH solver to ensure the density change between time steps is less than this threshold.

Minimum number of iterations	Important, but not investigated because the default value is known to be optimal
Maximum number of iterations	Not important because it is only used to prevent occurrence of infinite loops due to non-convergence of the pressure solver
Relaxation factor	Not important because default value leads to optimum convergence
Old pressure weight	Important, but not investigated because the default value is known to be optimal
Particle shifting	Not important because particles should not be enforced to a particular configuration
Buoyance model	Not important because it only affects simulation of non-isothermal flows
Density computation	Important, but not investigated because the default value is known to be optimal
Enable negative pressures	Not important because it only affects the fluid-solver interaction with dynamic solid objects
Enable tensile instability control	Not important because it only affects the fluid-solver interaction with dynamic solid objects
SPH Laplacian for viscosity	Not important because it is known to yield very similar simulations
Wall boundary pressure method	Important, but not investigated because the default value is known to be optimal

Wall boundary extrapolation	Important, but not investigated because the default value is known to be optimal
Wall hydrostatic pressure correction	Important, but not investigated because the default value is known to be optimal
Wall viscous correction	Important, but not investigated because the default value is known to be optimal
Free-surface density correction	Important, but not investigated because the default value is known to be optimal
Free-surface pressure correction	Important, but not investigated because the default value is known to be optimal
Near-free-surface identification coefficient	Important, but not investigated because it only affects simulations with free-surface correction, which is not used
Open boundary extrapolation	Not important because there is currently only a single choice and would only affect simulations with open boundaries
Courant-Friedrichs-Lewy (CFL) number	Important, but not investigated because the default value is known to be close to optimal
Time step diffusion coefficient	Not important because it only affects simulations with highly viscous flow
Clamp to multiple of Solid-Solver time step	Not important because it only affects fluid-solver interaction with dynamic solid objects

Adaptive time step	Important, but not investigated because the default value is known to be optimal
SPH kernel	Important, but not investigated since the same methodology can be used with other smoothing kernels.
Particle spacing	Important because it changes the resolution of the simulation.
Interaction-Radius to Particle-Size Ratio	Important because it changes the number of particles influencing the particle of interest. It is used to calculate the smoothing length for the smoothing kernel. The smoothing length is this ratio multiplied by the particle spacing.
Fluid settling uncertainty	Important because it effects the fluid depth. It accounts for the constant movement of the particles regarding the fluid depth.
Fluid depth uncertainty	Not important because the correct number of particles for a given fluid depth can be calculated.
Fluid properties	Not important because they will match the fluid properties of the experiment.
Measurement field size	Important, but not being investigated due to initial research results.

Dimension uncertainty	Important, but not investigated due to very small uncertainty range.
Forcing function amplitude uncertainty	Important because it effects the forcing function of the simulation and is based on the experimental uncertainty.
Forcing function frequency uncertainty	Important because it effects the forcing function of the simulation and is based on the experimental uncertainty.
Pressure transducer location uncertainty	Important because it effects the measurement field location in the simulation and is based on the experimental uncertainty.

Based on Table 1, 7 of the 37 parameters were investigated to determine their significance. The seven parameters were selected based on discussion with the George Washington University experimentalists and the Neutrino developers. These seven parameters were analyzed using the Sparse Grid Collocation and GaussPolynomialRom in RAVEN to obtain the first order Sobol indices for these parameters.

To perform the analysis, each parameter needed a range of values for RAVEN to sample. The range of values selected for each parameter were based on either the experimental measurement uncertainty or a range provided by the Neutrino developers for the simulation parameters. Table 2 shows the range of values for each parameter as well as the relative percentage of the value range [46]. These ranges are all considered to have a uniform distribution so each value in that range has the same probability of being selected.

Table 2: Investigated Parameter Value Ranges

Parameters	**Value Range**	**Relative Percentage Value Range (%)**
Stop threshold	0.0001 to 0.01	99
Particle spacing	0.007 m to 0.02 m	65
Interaction-radius to particle-size ratio	2.0 to 2.4	16.67
Fluid settling uncertainty	$-0.4\delta_r$ to $0.4\delta_r$ δ_r = particle spacing	200
Forcing function amplitude uncertainty	0.1012 m to 0.102 m	0.78
Forcing function frequency uncertainty	0.1099868 Hz to 0.110011 Hz	0.022
Pressure transducer location uncertainty	0.1006 m to 0.1026 m	1.95

Once the parameter value ranges were selected, the Neutrino interface needed to be updated to include changing all these parameters and not just the particle spacing. The stop threshold and interaction-radius to particle-size ratio were changed by searching for the correct parameter in the Neutrino input file and changing the value. The forcing function amplitude and frequency were changed by writing the dynamic expression scripts for the simulation tank and measurement field with the new amplitude and frequency. The pressure transducer location required changing the position of the measurement field as

well as writing the correct height location to the dynamic expression script for the oscillation.

The particle spacing parameter was changed by adjusting the parameter in the input file. However, depending on the particle size, other parameters needed to be adjusted. The particle size parameter affected the measurement field size, location, and the required number of particles for the specified fluid depth. Lastly, the fluid settling uncertainty was accounted for by adding or subtracting particles from the required number of particles calculated for the specified depth. For the modified Neutrino interface and RAVEN input file, please refer to Appendix A: RAVEN Files.

After running the RAVEN input file for the oscillating tank simulation, the first order Sobol indices for the seven parameters were calculated after conducting 15 simulations with differing input parameter values to train the ROM. Table 3 shows the resulting Sobol index for each parameter. For the complete RAVEN output, please refer to Appendix A: RAVEN Files.

Table 3: Sobol Index Results for Oscillating Tank Simulation

Parameters	First Order Sobol Index
Stop threshold	0.0284
Particle spacing	0.8036
Interaction-radius to particle-size ratio	0.0364
Fluid settling uncertainty	0.0171
Forcing function amplitude uncertainty	0.0006
Forcing function frequency uncertainty	0.0961
Pressure transducer location uncertainty	0.0178

As can be seen from the Sobol index results, the particle spacing contributes the most to the output variance resulting in it being the most important parameter. Further research could be done to determine higher order Sobol indices as well as using different methods of sensitivity analysis to better quantify the importance of particle spacing. However, this was done to show that particle spacing is an important parameter even when other simulation and experimental parameters are being investigated.

5 Particle Spacing Convergence Optimizer

Since particle spacing selection is an issue with SPH simulations, a method, besides parametric study, needs to be developed to determine the maximum particle spacing that provides approximately the same result as a simulation with a small particle spacing. By determining this optimum particle spacing value, it would provide the user converged results with the quickest computational runtime. However, to determine the optimum particle spacing for a simulation, it will require running many simulations of different particle spacings. Therefore, the goal would be to determine the optimum particle spacing for a few simulations and then draw a conclusion that can apply to similar simulations.

Additionally, the optimum particle spacing could depend on different scenario or simulation parameters, i.e. fluid height, pressure, flow rate, smoothing kernel, etc. The dependence of optimum particle spacing on simulation/scenario parameters increases the complexity of identifying an optimum particle spacing protocol that could be applied to many simulations. Therefore, understanding the affect each parameter has on the optimum particle spacing would allow for a more universal particle spacing protocol.

To determine the optimum particle spacing, a convergence optimizer was created using Python. The goal of the convergence optimizer, regarding SPH particle spacing, is to find the largest particle spacing that provides converged results. This will allow for the results to be the same as a very small particle spacing, but with a quicker runtime. To understand how SPH particle spacing converges, Figure 18 shows an example of a convergence study done in the paper *Computational evaluation of smoothed particle hydrodynamics for implementing blood flow modeling through CT reconstructed arteries* by Yi Qin, et. al [47].

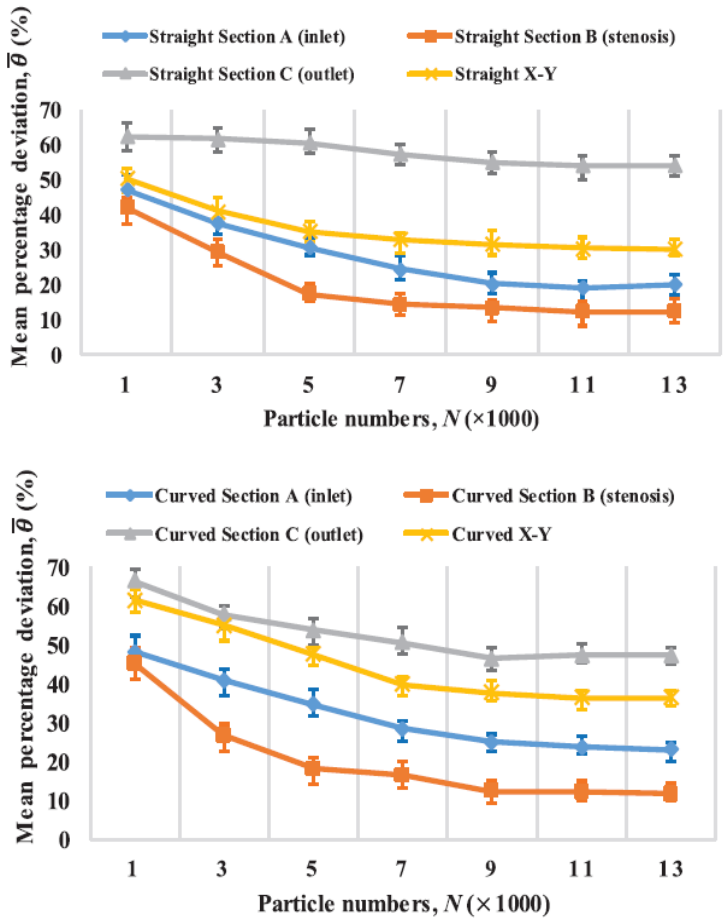


Figure 18: SPH particle spacing convergence study [47].

Based on the above plots, the results converge as the number of particles increase, or the particle spacing decreases. Therefore, 9,000 particles could be used instead of 13,000 particles since 9,000 particles provides essentially the same results, but with a shorter runtime.

For the above plots, the simulation results decrease to convergence as the number of particles increase, or the particle spacing decreases. However, there is a chance that simulation results could increase to convergence instead. Additionally, there could be fluctuation in the simulation results before convergence is reached as well as an early plateau. Figure 19 and Figure 20 show hypothetical simulation result plots.

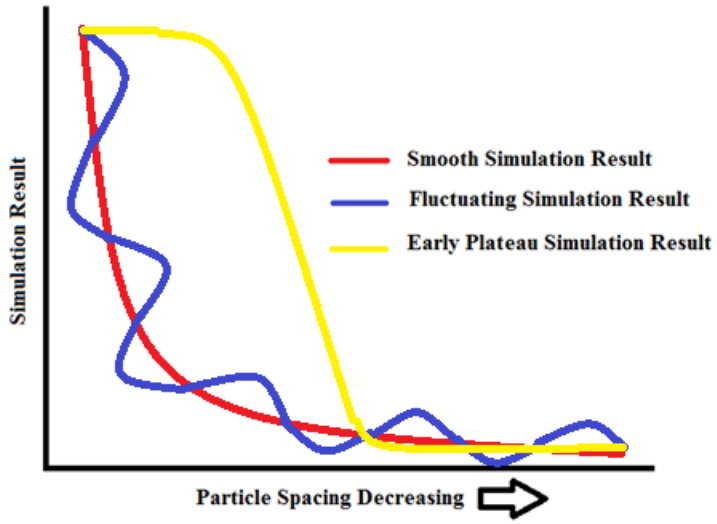


Figure 19: Hypothetical decreasing convergence simulation results.

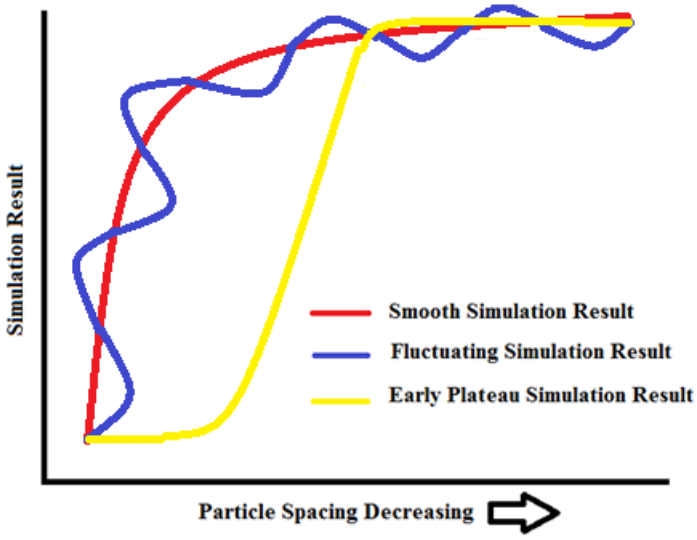


Figure 20: Hypothetical increasing convergence simulation results.

Since there is significant variability in the SPH simulation results, the convergence optimizer needs to be able to work for all these possible results. Therefore, a convergence

optimizer was developed in Python to be able to determine the optimum particle spacing for a simulation with all these possible variations in mind. The convergence optimizer consists of five Python classes and a main file. The following are the details of the five Python classes:

- XML editor class
    - This class contains one method that changes the particle spacing in the Neutrino input file.
- Command prompt class
    - This class contains two methods: one that executes the command to run Neutrino and the other to copy the results to a new file location so they are not overwritten.
- Adjust value class
    - This class contains two methods: one that determines the new particle spacing based on the user provided reduction method and the other determines the new particle spacing when optimization is occurring.
- Output file analysis class
    - This class contains three methods: one method reads the entire comma separated value (CSV) file containing the time and results at that time then stores it to an array, the next method returns an array with the specific data wanted (i.e. fluid height) from the total result array, and the last method determines the average result for the specific data.
- Analysis class
    - This class contains one method that checks if the result has converged.

    - At this point, it does not perform an analysis for every time step, but, with the framework, it could be added if it was needed in the future.

Each of the class files can be found in Appendix B: Convergence Optimizer Files. These classes typically do not need to be modified by the user. The only time the user might need to modify one of these classes is to change the Neutrino run command or to add the analysis for every time step. The file that does require user modification is the main file. The main file contains the function to perform the convergence optimization.

The first thing the user needs to do is update the input parameters. The following is a list of input parameters for the convergence optimizer:

- Input file path
    - Specifies the Neutrino input file directory location.
- Results path
    - Specifies the directory location of the Neutrino results.
- Output path
    - Specifies the directory location of where to move the Neutrino results.
- Starting particle spacing value
    - Specifies what particle spacing should be used for the first iteration.
- Minimum number of iterations
    - Decreases the chance of the optimizer reaching convergence on a plateau with a large particle spacing rather than on the actual converged result.
- Maximum number of iterations
    - Provides an exit for the optimizer.

- Convergence criterion
    - Specifies how close the results should be for convergence to be reached.
    - It is used in Eq. (22) as shown below.

$$\left|\frac{New\ Result - Previous\ Result}{Previous\ Result}\right| < Convergence\ Criterion \quad (22)$$

- Particle spacing reduction method
    - Specifies how the particle spacing will be reduced at the start of the optimization (by half, by a quarter, etc.).
- Number of converged iterations
    - Specifies how many iterations should be converged before the optimizer starts looking for the optimum particle spacing.
    - It decreases the chance of the optimizer saying convergence was reached on a plateau with a large particle spacing rather than on the actual converged result.
    - Is also adds the capability to handle the fluctuating result since there would be a very small probability that multiple particle spacings would provide the same result during a fluctuating convergence.
- Minimum particle spacing change
    - Specifies how small the change in particle spacing could be while searching for the optimum particle spacing before the optimizer exits.
    - It provides an exit for the optimizer.
- Time step to start averaging results
    - Specifies the first time step that the results will begin to be averaged.

    - The averaging will continue until the end of the simulation.
- The maximum simulation time
    - Specifies the end time of the simulation in seconds.
    - It is used to ensure that an extra time step result with inaccurate information is not included in the averaging.

Once the input parameters have been selected, the user can run the convergence optimizer and it will start looking for the optimum particle spacing. Figure 21 depicts the flow of the convergence optimizer and the main file can be found in Appendix B: Convergence Optimizer Files.

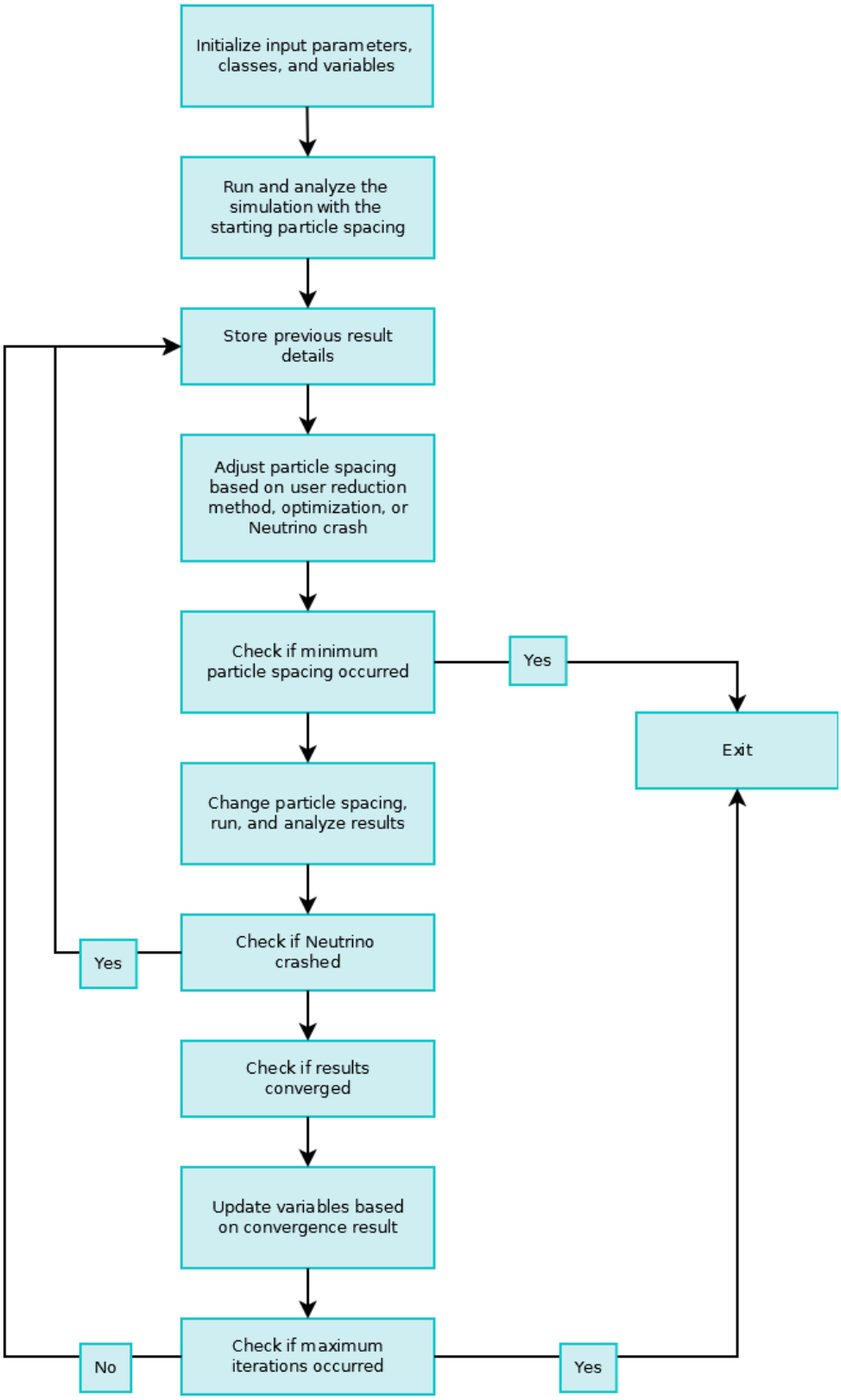


Figure 21: Convergence optimizer flow chart.

To better understand how the convergence optimizer works, the following is an example of the convergence optimizer process. Before the example is detailed, a couple of the variables in the optimizer need to be discussed. The following are important variables in the converged optimizer:

- Maximum converged particle spacing
    - Specifies the largest particle spacing that provided a converged result.
- Maximum converged average
    - Specifies the result of the largest particle spacing that provided a converged result.
- Minimum converged particle spacing
    - Specifies the smallest particle spacing that provided a converged result.
- Minimum converged average
    - Specifies the result of the smallest particle spacing that provided a converged result.
- Minimum non-converged particle spacing
    - Specifies the smallest particle spacing that did not provide a converged result.
- Minimum non-converged average
    - Specifies the result of the smallest particle spacing that did not provide a converged result.

With the important variables discussed, assume the convergence study plot looks like Figure 22 which shows the convergence study plot of any simulation result vs. time, the converged result, and the optimum particle spacing. The convergence information would not be known before hand, but for the example it will help demonstrate the optimizer.

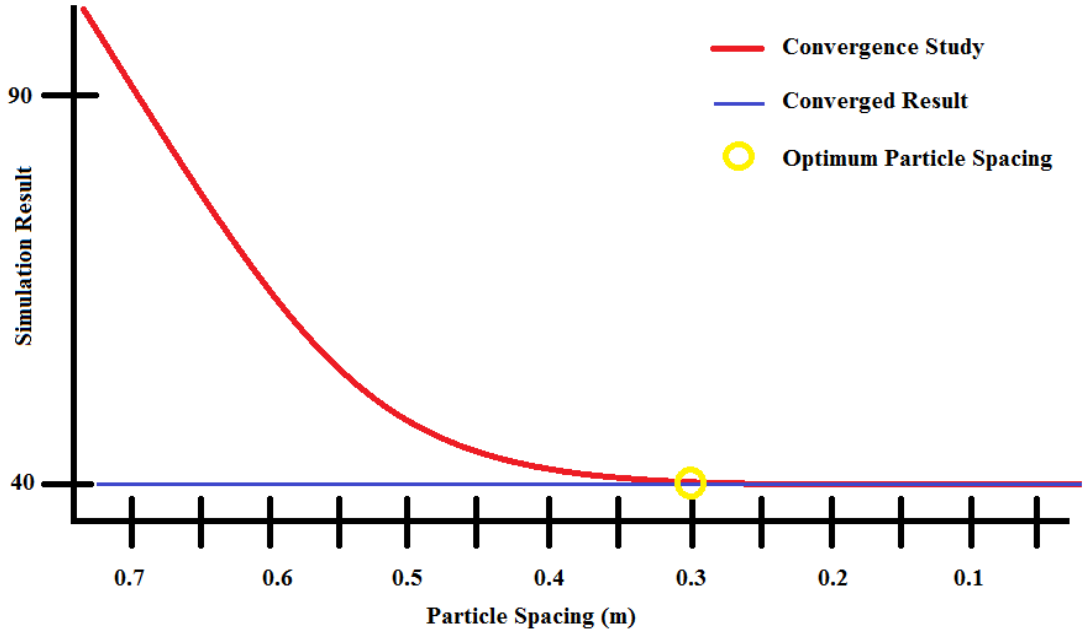


Figure 22: Example converged study plot with converged result and optimum particle spacing.

Based on the convergence optimizer details above, the user might select the following input parameters:

- Starting particle spacing value = 0.7 m
- Minimum number of iterations = 4
- Maximum number of iterations = 10
- Convergence Criterion = 0.01
- Reduction method = 0.5
- Number of converged iterations = 2
- Minimum particle spacing change = 0.015 m

The other input parameters, such as directory locations, time step to start averaging, and maximum simulation time, would need to be included, but, for this example, they are not needed. Additionally, the simulation result could be any result, i.e. height, pressure,

velocity, etc., but for this example, it is just a generic result to show the convergence optimizer process.

The convergence optimizer would start by running the simulation with the 0.7 m particle spacing. The optimizer would then gather the result of the simulation with the 0.7 m particle spacing, which is assumed to be 90, and stores the result and particle spacing as the previous values. Since the reduction method is by half and optimization has not occurred, the next particle spacing calculated would be 0.35 m. The optimizer would determine the change in particle spacing, which is 0.35 m, and make sure it is not less than the minimum particle spacing change of 0.015 m. Next, the simulation with the 0.35 m particle spacing would be ran and the output gathered. The optimizer would check to make sure Neutrino did not crash by checking the length of the simulation results.

Assuming Neutrino did not crash, the optimizer determines if the results converged. The 0.35 m particle spacing provided a result of 43. Therefore, the difference between the 0.7 m and 0.35 m particle spacing results is 0.52 which is greater than the convergence criterion. Since convergence did not occur, the number of converged iterations is reset back to zero, the minimum converged average is changed to the 0.35 m result, since optimization has not occurred, and the minimum non-converged particle spacing and average is changed to the 0.7 m particle spacing and result. The optimizer checks that the maximum number of iterations has not been reached, increases the number of iterations, and restarts the process.

Once the number of converged iterations has been met, the optimizer would start searching for the optimum particle spacing. The next particle spacing used would be the midpoint between the minimum non-converged particle spacing and the maximum

converged particle spacing. The optimization process continues until the maximum number of iterations or the minimum particle spacing change is reached. Table 4 shows the details of the entire optimization process for this example, Table 5 shows the particle spacing change calculations, and Table 6 shows the convergence result calculations. Figure 23 shows a plot of the optimizer results for this example on the convergence plot.

Table 4: Convergence Optimizer Example Details

Iter.	Particle Spacing (m)	Result	Particle Spacing Change (m)	Converged Result Using Left Hand Side Of Eq. (22)	Converged Iterations	Min Converged Particle Spacing (m) and Result Average	Max Converged Particle Spacing (m) and Result Average	Min Non-Converged Particle Spacing (m) and Result Average	Has Optimization Started?
0	0.7	90	-	-	0	0, 0	0, 0	5, 0	No
1	0.35	43	0.35	0.52	0	0.35, 43	0, 0	0.7, 90	No
2	0.175	40	0.175	0.07	0	0.175, 40	0, 0	0.35, 43	No
3	0.0875	40	0.0875	0	1	0.0875, 40	0.175, 40	0.35, 43	No
4	0.04375	40	0.04375	0	2	0.04375, 40	0.175, 40	0.35, 43	No
5	0.2625	40	0.0875	0	-	0.04375, 40	0.2625, 40	0.35, 43	Yes
6	0.30625	40.5	0.04375	0.013	-	0.04375, 40	0.2625, 40	0.30625, 40.5	Yes
7	0.284375	40	0.021875	0	-	0.04375, 40	0.284375, 40	0.30625, 40.5	Yes
8	0.295313		0.010938						Exit

Table 5: Convergence Optimizer Example Particle Spacing Calculations

Iter.	Particle Spacing Change Calculation (m)	New Particle Spacing Calculation (m)
1	0.7 / 2 = 0.35	0.7 – 0.35 = 0.35
2	0.35 / 2 = 0.175	0.35 – 0.175 = 0.175
3	0.175 / 2 = 0.0875	0.175 – 0.0875 = 0.0875
4	0.0875 / 2 = 0.04375	0.0875 – 0.04375 = 0.04375
5	(0.35 – 0.175) / 2 = 0.0875	0.35 – 0.0875 = 0.2625
6	(0.35 – 0.2625) / 2 = 0.04375	0.35 – 0.04375 = 0.30625
7	(0.30625 – 0.2625) = 0.021875	0.30625 – 0.021875 = 0.284375
8	(0.30625 – 0.284375) = 0.010938	0.30625 – 0.010938 = 0.295313

Table 6: Convergence Optimizer Example Convergence Calculations

Iter.	Convergence Calculation
1	| (43 – 90) / 90 | = 0.52
2	| (40 – 43) / 43 | = 0.07
3	| (40 – 40) / 40 | = 0
4	| (40 – 40) / 40 | = 0
5	| (40 – 40) / 40 | = 0
6	| (40.5 – 40) / 40 | = 0.013
7	| (40 – 40) / 40 | = 0

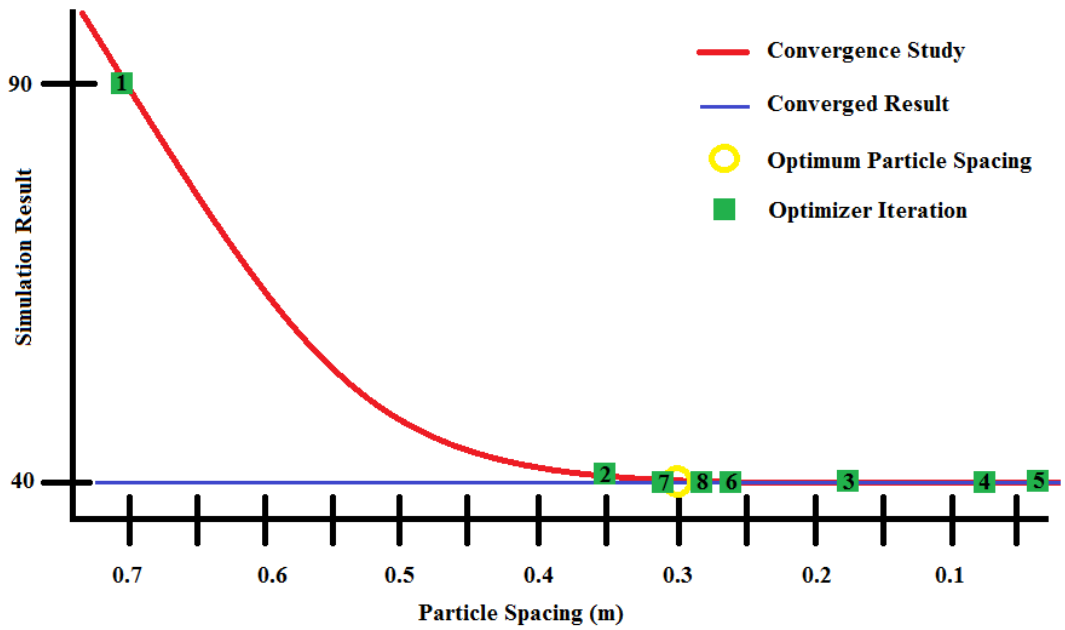


Figure 23: Example optimizer result plot on the convergence study plot.

Based on the details in Table 4, the optimizer exits on the eighth iteration due to the particle spacing change being less than the input parameter. The optimum particle spacing found by the optimizer was 0.284375 m. This is because it is the largest converged particle spacing tried before the optimizer was required to exit.

The above example provides a hypothetical example of how the convergence optimizer works. Depending on the scenario being simulated, the required input parameters could vary as well as increase or decrease runtime of the optimizer. For example, if the user selected the reduction method as ¾, it would take longer for the optimizer to reach the converged particle spacing than the ½ reduction method would. The goal of the optimizer is to determine the optimum converged particle spacing while also allowing the user the ability to modify the parameters of the optimizer.

6 Optimum Particle Spacing Determination

Since there is no guidance on particle spacing, the optimum particle spacing for very simple simulations needs to be determined before complex simulations are analyzed. This allows for a few parameters to be varied and the results analyzed rather than many parameters being varied and not knowing what is contributing to the result. The goal of this process is to understand the results and determine a protocol for a simple case before trying to analyze a complicated system.

The simple simulation used static fluid in a rigid box. The model consisted of filling a rigid box with fluid, letting the fluid settle such that there was very little motion of the particles, and then obtaining the fluid height. Three models were analyzed. The parameters that were varied among the three models, besides particle spacing, were:

- Rigid box width and length
- Fluid height
- Flow rate
- Simulation time

From the above parameters, only the rigid box width, length, and fluid height parameters are of importance for determining the particle spacing protocol. The flow rate does not contribute to the protocol determination because it is not a continuous flow rate and is only used to fill the rigid box to a particular fluid height and the simulation time does not contribute to the protocol determination because it varies, based on the simulation, to provide enough settling time for the particles. However, the rigid box size and fluid height are important because both parameters affect the required particle spacing. If the rigid box

length is 1 m, a 2 m particle spacing cannot be used because it is larger than the box itself. Therefore, this model only varies a single parameter which is the scale. This allows for a protocol to be determine based on the scale of the simulation.

6.1 Neutrino Models

For each simulation, the model consisted of the following Neutrino items:

- One rigid box
- One rigid plane
- One flow particle emitter
- One measurement field
- Neutrino IISPH solver

The rigid plane was placed above the rigid box to keep the particles inside the rigid box. As the particles fill the box, some particles might splash above the rigid box and the rigid plane keeps the particles from leaving the rigid box.

For the rigid box, the only property that was changed was the scale. All other properties were left as the default values. The flow particle emitter properties that were changed were the flow rate, stop time, position, scale, allow compressibility, and allow permeability. The allow compressibility causes the particles to be slightly compressed as they are emitted and the allow permeability results in ghost particles above the particle emitter resulting in more particles being emitted than expected. Both parameters were turned off to prevent any particle explosions from occurring and to prevent additional particles from being emitted. The position and scale of the flow particle emitter were varied during the initial simulation setup to determine values that reduced the amount of splashing and particle explosions. The measurement field properties that were changed were the scale

and the export information. The scale of the measurement field was selected to be in the middle of the rigid box so that side wall effects were not included in the results and the information extracted is the average height in the measurement field. Lastly, all Neutrino IISPH solver properties stayed the same except for the particle spacing. Figure 24 shows the Neutrino setup of the model.

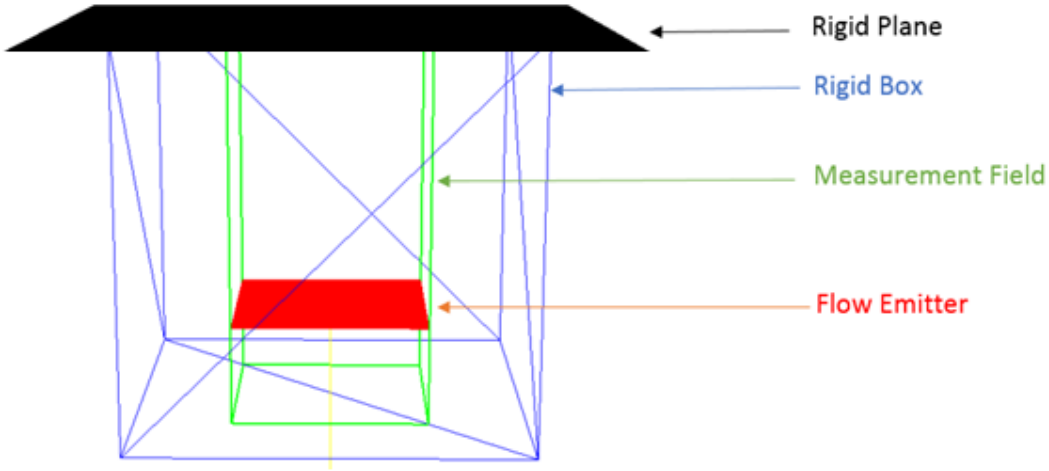


Figure 24: Model setup.

Three simulation models, each of a different scale, were analyzed using the convergence optimizer. The three simulation models are called the following:

- Small box
- Medium box
- Large box

Table 7 provides the details of each of the three simulation models.

Table 7: Model Details

Model	Rigid Box Length and Width (m)	Analytical Fluid Height (m)	Flow Particle Emitter Size (m)	Flow Rate (m^3/sec)	Emitter Time On (sec)	Simulation Time (sec)
Small	0.5 x 0.5	0.1	0.25 x 0.25	0.0125	2	6
Medium	7.5 x 7.5	5	5 x 5	93.75	3	12
Large	30 x 30	40	20 x 20	12,000	3	18

For each model, some of the convergence optimizer parameters were the same. The following is a list of the parameters and their values that were the same for each model:

- Minimum number of iterations = 3
- Maximum number of iterations = 25
- Convergence criterion = 0.05
- Number of converged iterations = 3
- Reduction method = 0.5
- Minimum particle spacing change = 0.000001

The rest of the convergence optimizer details can be found in Table 8.

Table 8: Model Convergence Optimizer Details

Model	Starting Particle Spacing (m)	Time Step to Start Averaging
Small	0.2	200
Medium	5	500
Large	5	800

6.2 Results

Each model was analyzed using the convergence optimizer. The convergence optimizer outputs details of the run, the particle spacing and average height results, and the runtime of each simulation. Table 9 shows the optimum particle spacing found for each model, Figure 25 through Figure 27 show plots of fluid height vs. particle spacing for each model with the optimum particle spacing shown in red, and Figure 28 through Figure 30 show plots of computational runtime vs. particle spacing for each model with the optimum particle spacing shown in red. Further information regarding Figure 25 through Figure 27 is provided in the Sphere Packing Analysis section. The details of each model can be found in Appendix C: Optimum Particle Spacing Details Using Neutrino Result.

Table 9: Optimum Particle Spacing Using Neutrino Result

Model	Optimum Particle Spacing (m)
Small	0.035188
Medium	1.635864
Large	5.000001

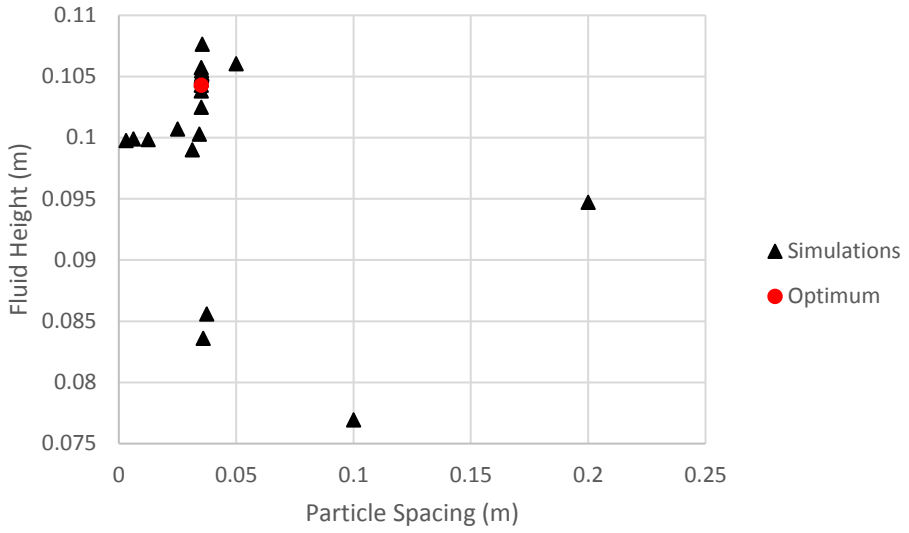


Figure 25: Small box fluid height vs. particle spacing plot using Neutrino result.

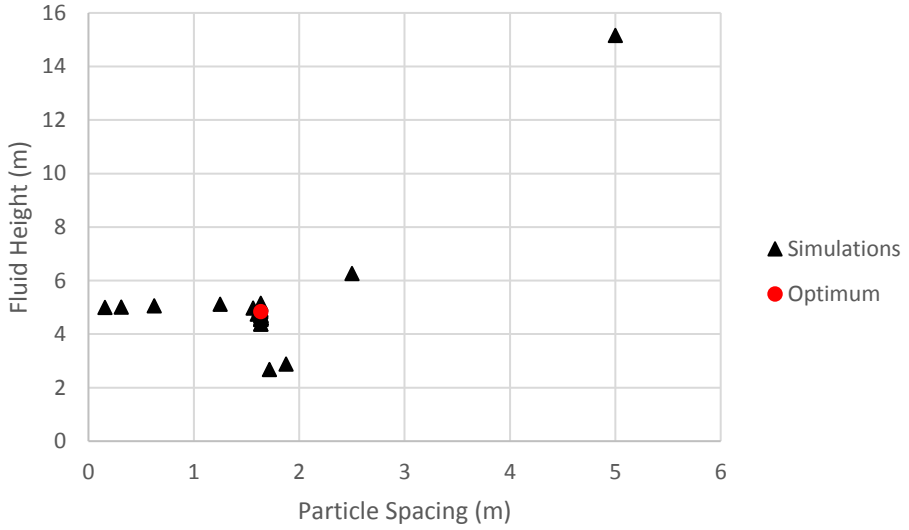


Figure 26: Medium box fluid height vs. particle spacing plot using Neutrino result.

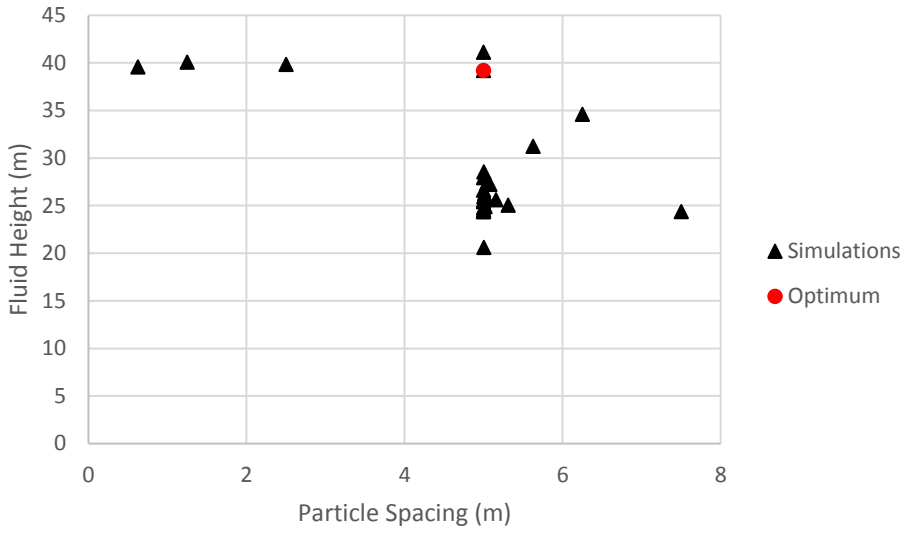


Figure 27: Large box fluid height vs. particle spacing plot using Neutrino result.

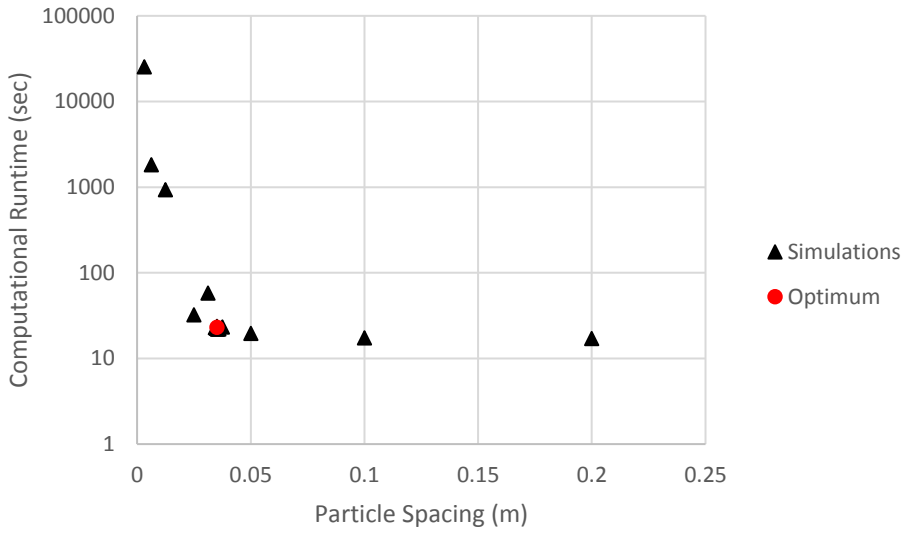


Figure 28: Small box computational runtime vs. particle spacing plot using Neutrino result.

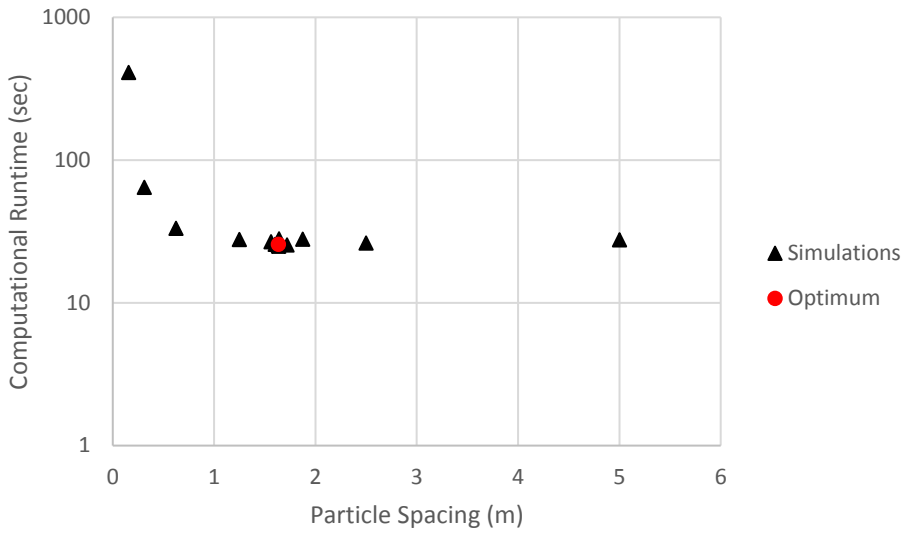


Figure 29: Medium box computational runtime vs. particle spacing plot using Neutrino result.

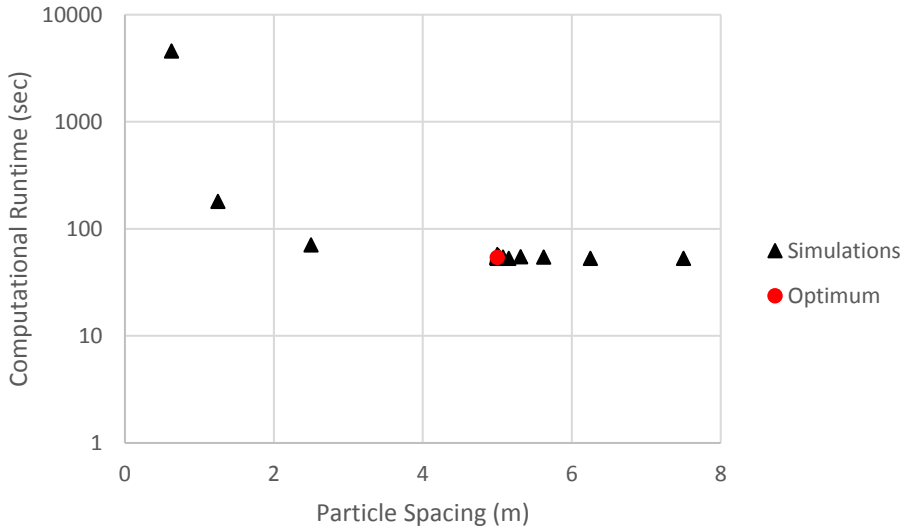


Figure 30: Large box computational runtime vs. particle spacing plot using Neutrino result.

From the results above, the convergence optimizer was able to find an optimum particle spacing for each model. Additionally, the optimum particle spacing provides a much shorter computational runtime than choosing a smaller particle spacing.

The optimum particle spacing determination shows how unrefined a traditional parametric study is. By only choosing several particle spacings, the user has limited knowledge regarding the convergence and no knowledge about the optimum particle spacing. As an example, consider a parametric study was conducted on the small box model. The user selected particle spacings of 0.2 m, 0.1 m, and 0.05 m. Figure 31 shows the results for this parametric study circled with blue in regard to the optimum particle spacing determination.

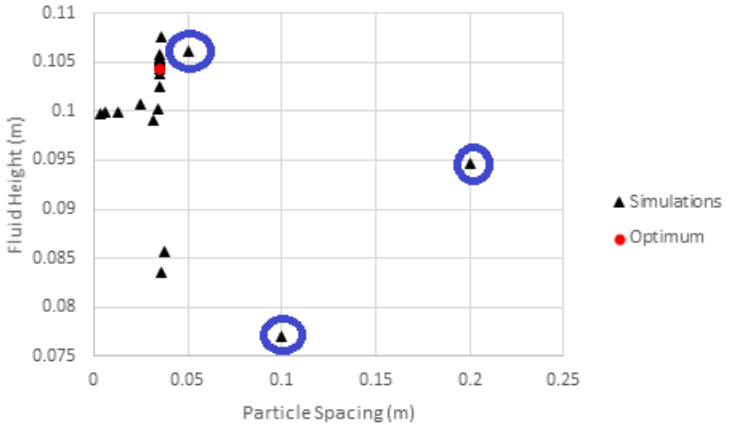


Figure 31: Parametric study example.

Based on Figure 31, the user would select the 0.05 m particle spacing since it provided a fluid height closest to the analytical fluid height of 0.1 m. However, based on the optimum particle spacing determination, convergence is not reached until a particle spacing of ~0.035 m or smaller is used. Additionally, no optimization information can be gained from this parametric study.

While optimum particle spacing determination requires many simulations, the results from the optimization provide valuable information regarding convergence and optimization that traditional parametric studies cannot.

7 Sphere Packing Arrangement

Since the particle spacing acts as the particle diameter, the position and movements of the particles are dependent on how spheres can move and position themselves. With sphere packing, there are different ways that spheres can arrange. When static spheres are investigated, the different sphere packing arrangements can provide different heights. For example, spheres arranged in a cubic form are stacked higher than spheres in a hexagonal form. Figure 32 shows different forms of sphere arrangement.

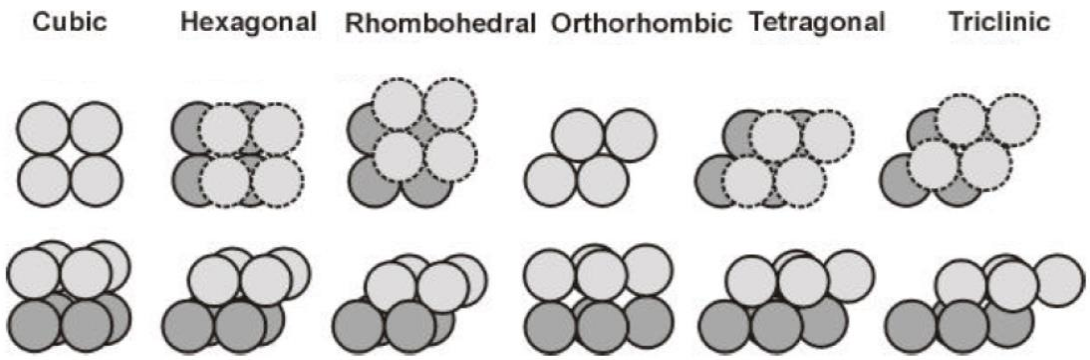


Figure 32: Sphere arrangements [48].

Based on the arrangements above, the height of the particles depends on how the particles are arranged. Besides the static arrangement of spheres, the SPH particles move throughout the simulation. Hence, the particles would move from one arrangement to another arrangement changing the height of the particles based on the sphere packing arrangement.

To determine a range of possible heights, two different sphere packing arrangements were investigated. The first arrangement is the simple cubic (SC). The SC arrangement contains particles in straight, neat layers with each layer directly above another layer providing large spaces between particles [49]. The second arrangement is the

face-centered cubic (FCC). This arrangement contains layers of particles that are altered and stacked to fit into the gaps [49]. It is more compact than the SC arrangement. These two arrangements were chosen because one provides large spaces between particles which acts as the maximum height and the other is closely packed which acts as the minimum height. Figure 33 shows the SC arrangement on the left and the FCC arrangement on the right.

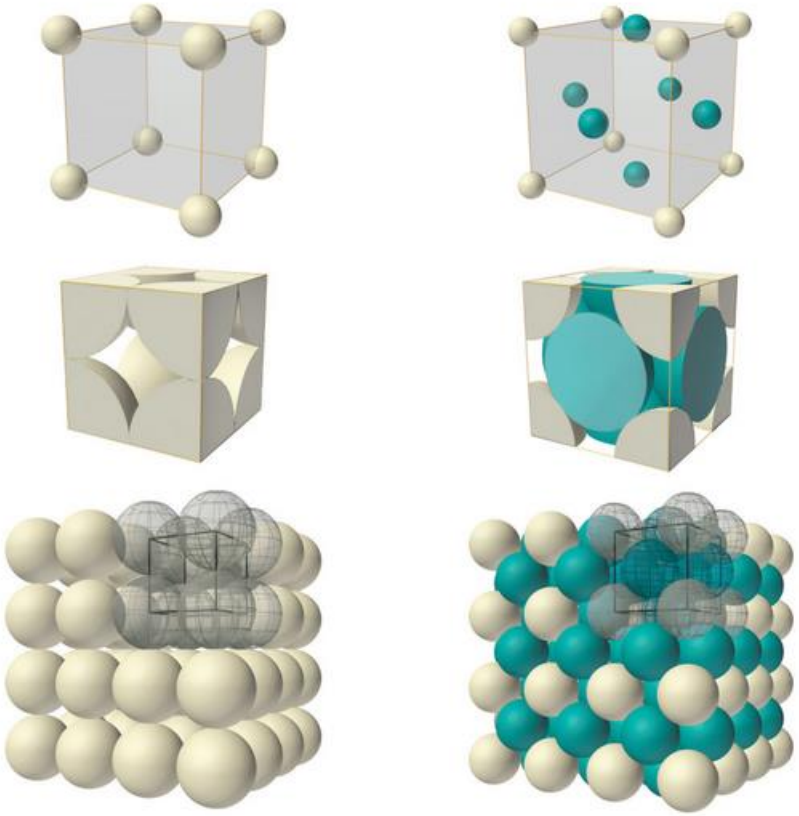


Figure 33: SC arrangement (left) and FCC arrangement (right) [50].

With sphere packing arrangements, the unit cell, which is the basic repeating unit, is analyzed [50]. The SC unit cell consists of one particle and only provides a packing density of 52% while the FCC unit cell consists of four particles and provides a packing density of 74% [49]. The packing density specifies the amount of space occupied by

particles [50]. The volume of the unit cell for the SC arrangement is shown in Eq. (23) and the volume of the unit cell for the FCC arrangement is shown in Eq. (24) [51].

$$V_{SC} = 8 * r^3 \tag{23}$$

$$V_{FCC} = 16\sqrt{2} * r^3 \tag{24}$$

7.1 Sphere Packing Analysis

Once the two arrangements were identified, the next step was to determine what the fluid height would be if the particles were in each of these arrangements. To determine this, the number of particles in the simulation needed to be determined. While Neutrino does provide the number of fluid particles in a simulation, the sphere packing analysis needed the number of particles for simulations that have not been conducted. Therefore, a more detailed understanding of the flow particle emitter was needed.

Since the flow particle emitter has an area where particles are emitted, how Neutrino rounded the number of particles emitted needed to be determined. This was determined by selecting a flow particle emitter size of 1 m by 1 m, resulting in a 1 m^2 area, and then changing the particle spacing. Table 10 shows the particle spacings used, the number of particles along the length and width of the emitter, the area of the particles, and what direction the number of particles were rounded.

Table 10: Emitter Size Rounding of Number of Particles

Particle Spacing (m)	Number of Particles Along Width and Length of Emitter	Area of Particles (m^2)	Rounding Direction
0.3	3 x 3	0.81	Down
0.23	4 x 4	0.85	Down
0.22	4 x 4	0.77	Down
0.17	5 x 5	0.72	Down
0.1	10 x 10	1	Equal

Based on the above table, Neutrino rounds the number of particles down to the nearest integer based on the particle emitter size. Next, a similar analysis needed to be done on the number of emissions of the particle emitter based on the flow rate and the time the particle emitter is actively emitting particles. For this analysis, the 1 m by 1 m particle emitter flow rate was set to 0.2 m^3/s for 2 sec. Analytically, this should result in a volume emitted of 0.4 m^3. Again, the particle spacing was varied and the emission volume was analyzed by counting the number of emissions that occurred. Table 11 shows the details of the emission amount rounding.

Table 11: Emission Amount Rounding Details

Particle Spacing (m)	**Number of Emissions**	**Emission Volume (m^3)**	**Volume per Emission (m^3/emission)**	**Rounding Direction**
0.3	2	0.49	0.24	Up
0.23	2	0.39	0.19	Down
0.22	2	0.35	0.18	Down
0.17	3	0.36	0.12	Down
0.1	4	0.4	0.1	Equal

Based on the results in Table 11, Neutrino will round up, emitting more particles than wanted, if the wanted volume, in this case 0.4 m^3, divided by the volume per emission, 0.243 m^3, results in a number with a decimal point of 0.5 or higher. Otherwise, Neutrino will round down and not emit enough particles.

Once the details of how Neutrino rounds was known, the sphere packing arrangement analysis could be conducted. This analysis was conducted by developing a Python script to determine the expected fluid height using both the SC arrangement and the FCC arrangement. Figure 34 shows the flow chart of the sphere packing analysis script. The file can be found in Appendix D: Sphere Packing Arrangement Analysis Script.

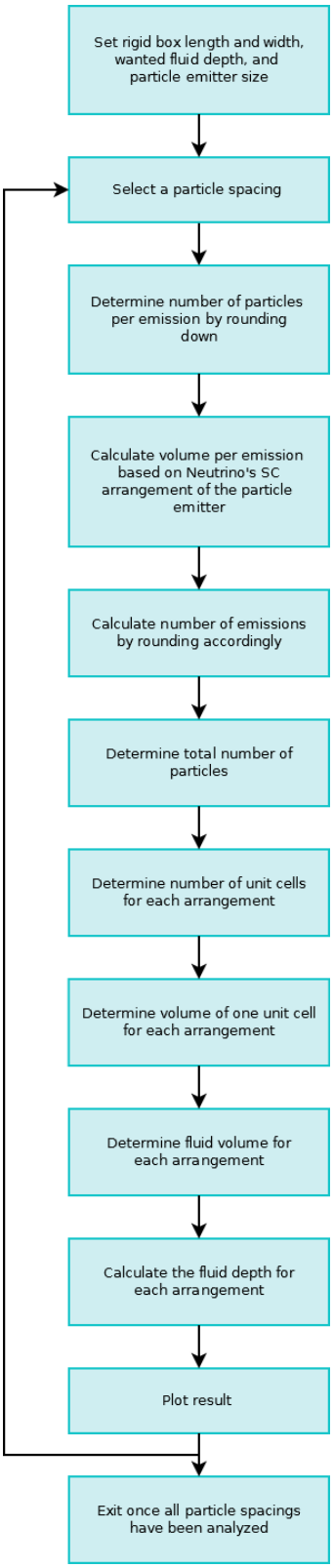


Figure 34: Sphere packing analysis flow chart.

To show how the sphere packing analysis script works, the small box model was analyzed to determine the expected fluid depths for the two different packing arrangements. Figure 35 shows the sphere packing analysis script plot for the small box model. The plot represents the analytical results using the Neutrino rounding protocol.

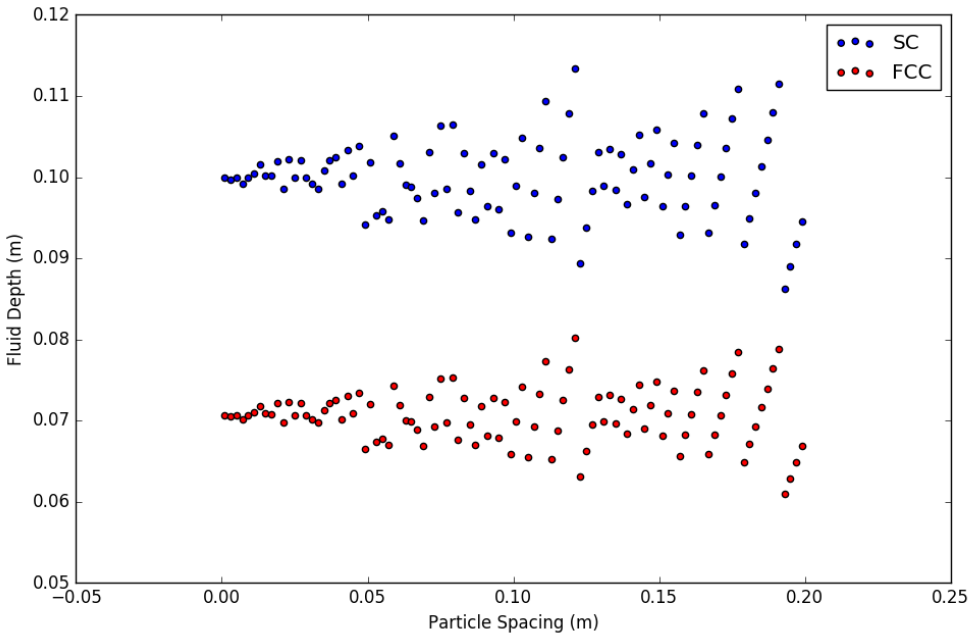


Figure 35: Sphere packing analysis example plot.

As seen in the above plot, the fluid depth range for both arrangements widens as the particle spacing becomes larger due to the Neutrino rounding. At some particle spacings, too many particles are emitted resulting in a higher fluid depth and vice versa with too few particles emitted. This also explains the widened range of results in the optimum particle spacing plots (Figure 25 through Figure 27) as the optimizer attempted to find the optimum particle spacing. The slight change in particle spacing affects the number of particles being emitted due to the Neutrino rounding protocol. Therefore, this rounding results in different fluid depths even at slight particle spacing changes.

8 Comparison Between Neutrino Results and Sphere Packing Results

The results from the optimum particle spacing were compared with the sphere packing arrangement results for the three models. The goal was to determine how the Neutrino results compared to the different sphere arrangement results. Figure 36 through Figure 39 show plots of the optimum particle spacing results and sphere arrangement results for the three models. The Python scripts for these comparisons can be found in Appendix E: Optimum Particle Spacing and Sphere Packing Arrangement Comparison Scripts.

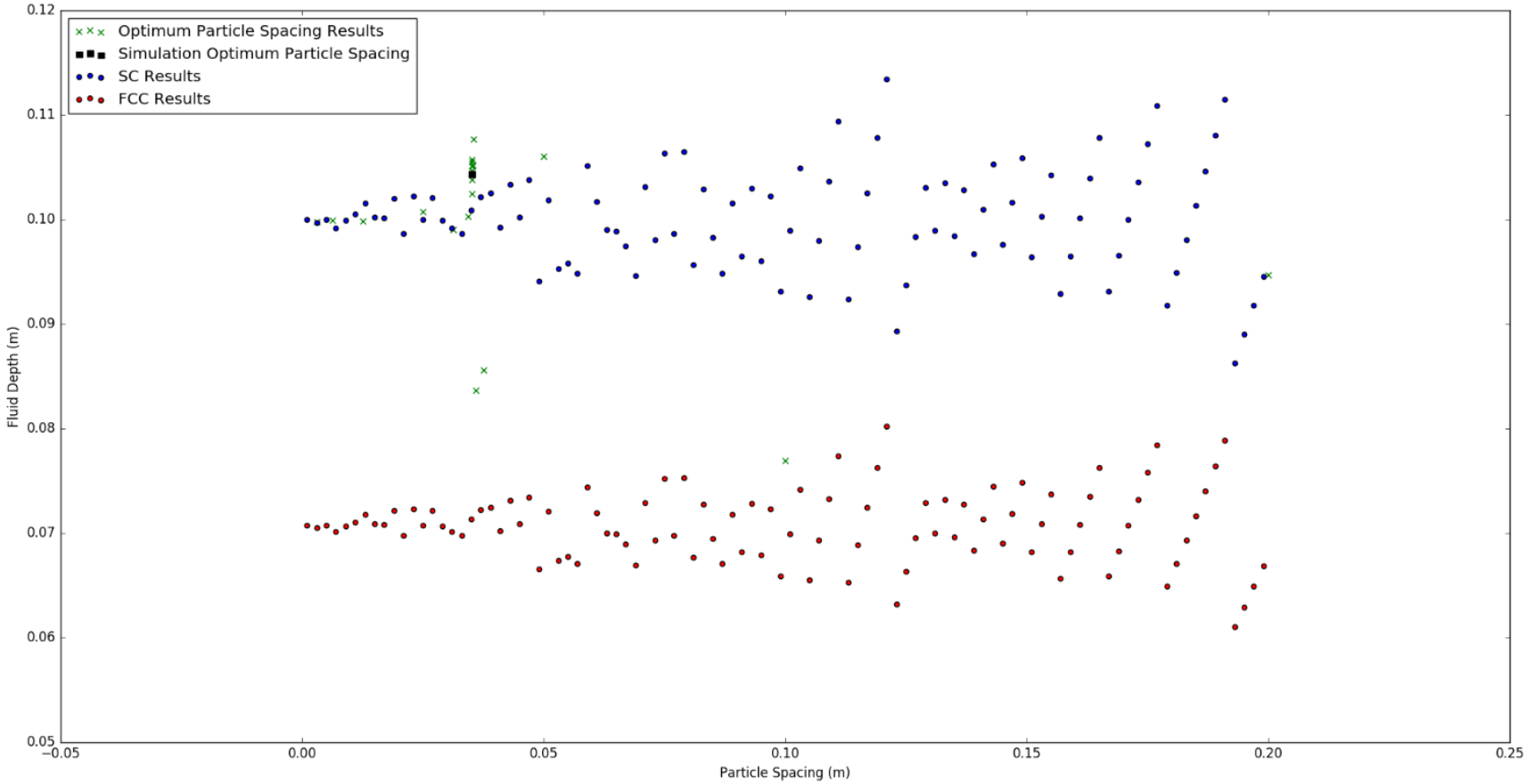


Figure 36: Small box optimum particle spacing and sphere packing arrangement comparison plot.

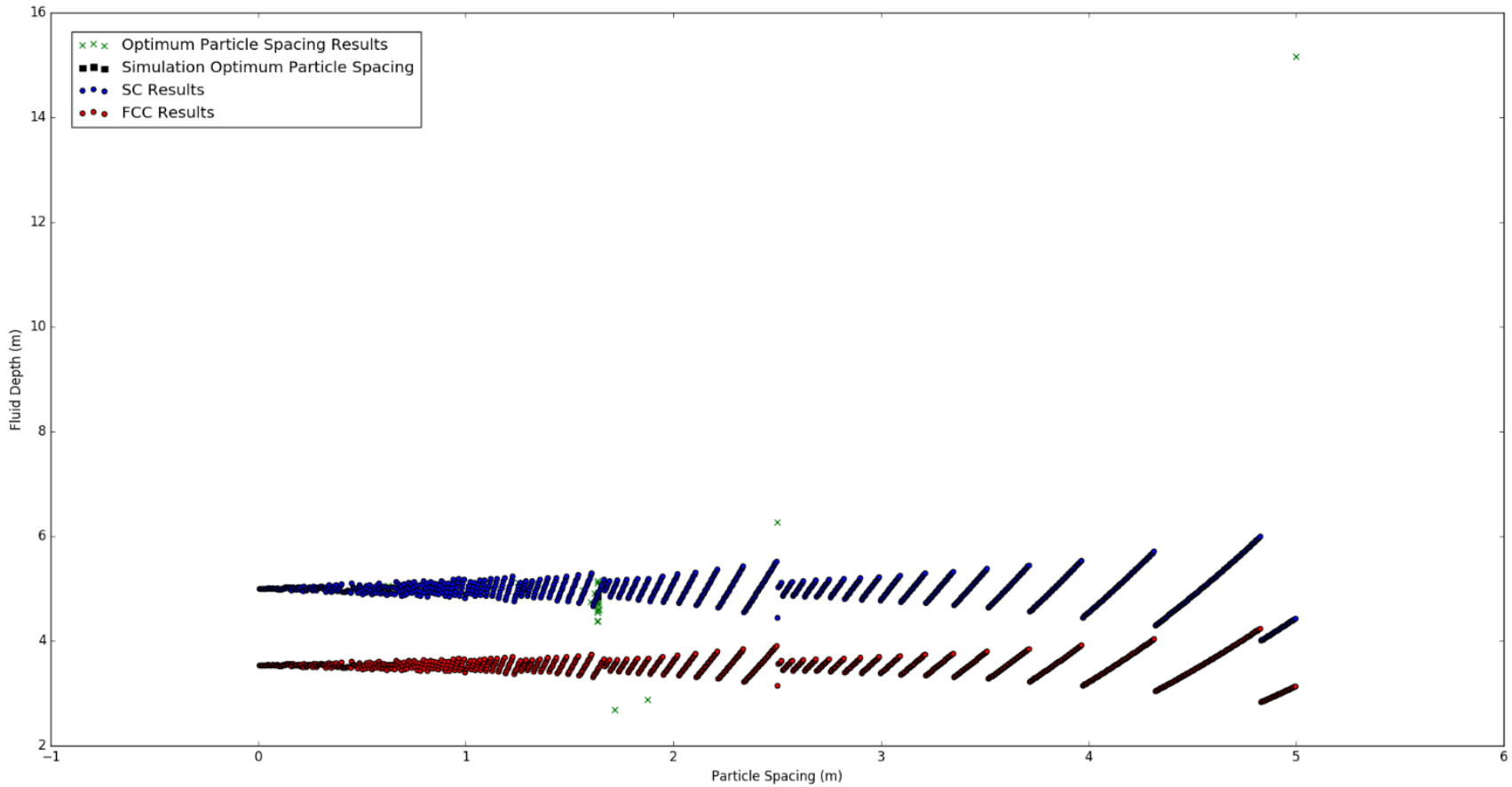


Figure 37: Medium box optimum particle spacing and sphere packing arrangement comparison plot.

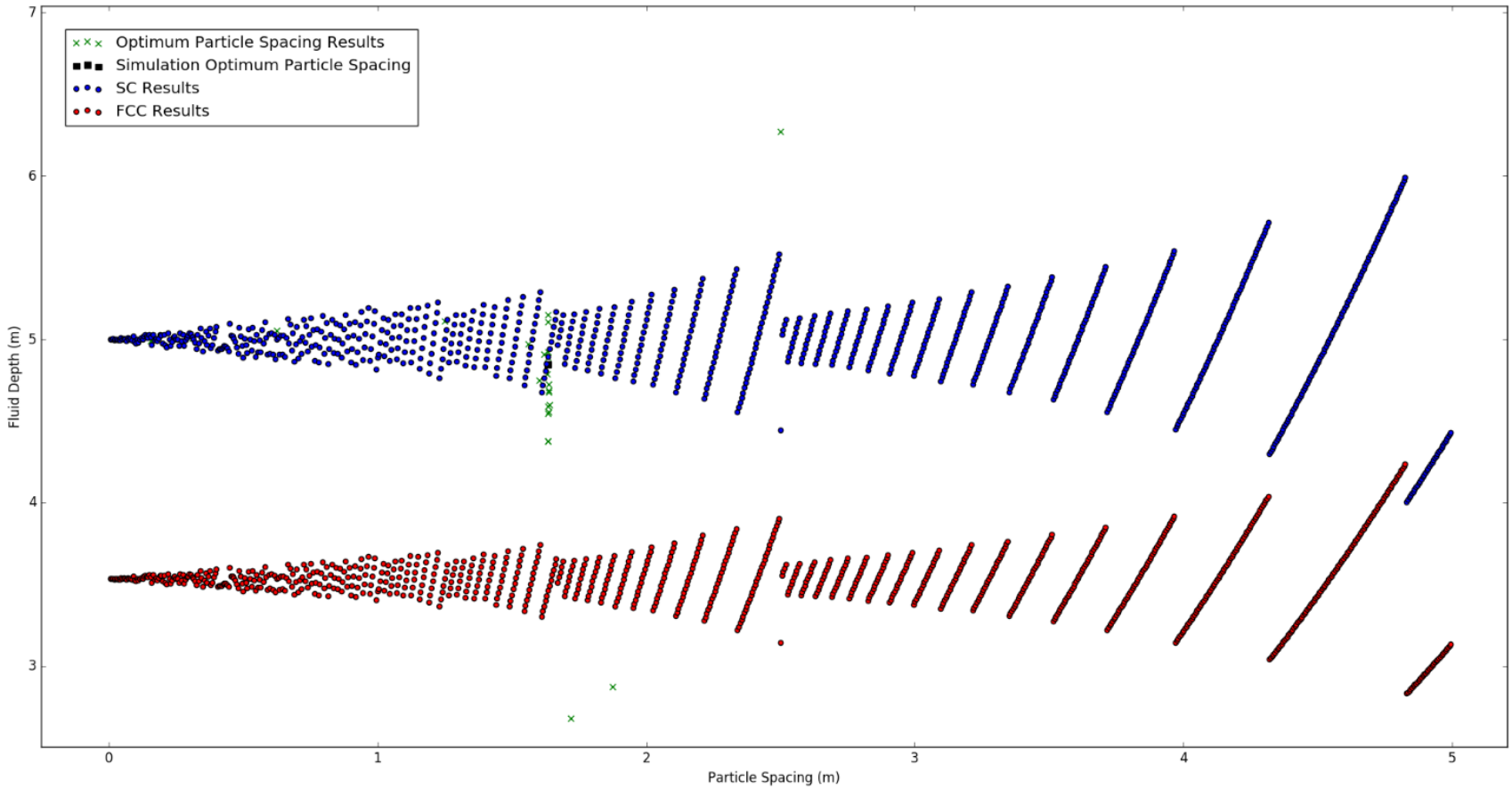


Figure 38: Medium box optimum particle spacing and sphere packing arrangement comparison plot zoomed.

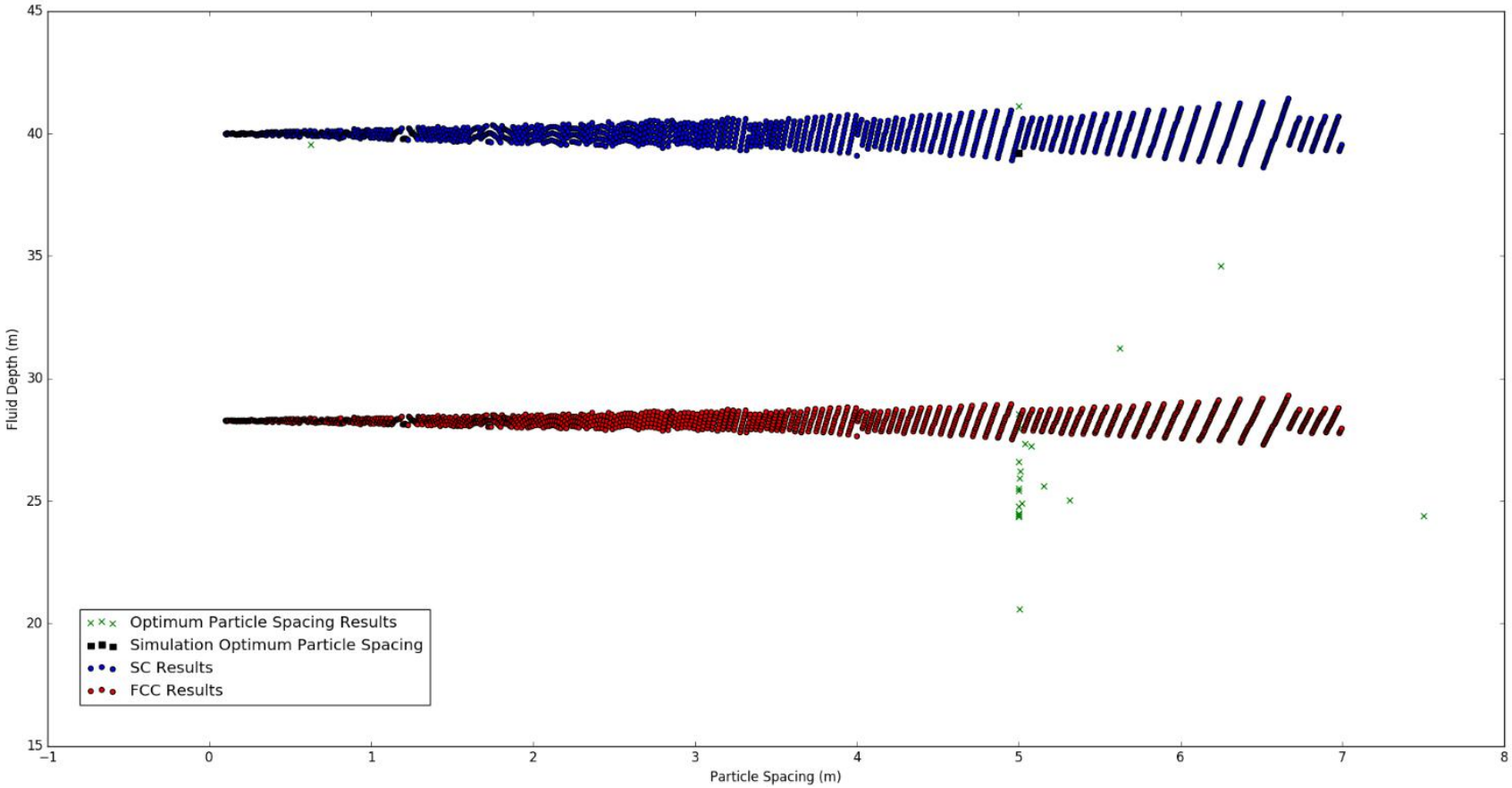


Figure 39: Large box optimum particle spacing and sphere packing arrangement comparison plot.

The plots show the Neutrino simulation results tend to follow the SC arrangement more often than the FCC arrangement. In Figure 36, there are only two simulation points between the two arrangements and one that is close to the FCC arrangement. The rest are within the SC arrangement values. In Figure 37 and Figure 38, there are only two simulation points close to the FCC arrangement. However, in Figure 39, there are more simulation points close to the FCC arrangement.

Based on these observations, the Neutrino height results tend to follow an SC arrangement. However, this does not make physical sense given the nature of particles. While the particles might occasionally align themselves in an SC arrangement, it would make more sense that they arrange themselves in a more compact manner such as an FCC arrangement. To confirm, the simulation from the small box model was looked at visually to see how the particles arranged themselves. Figure 40 shows the small box model at the last time step with the optimum particle spacing to visually analyze the particles' arrangements. Only one-time step was analyzed since there is little movement of the particles at the end of the simulation.

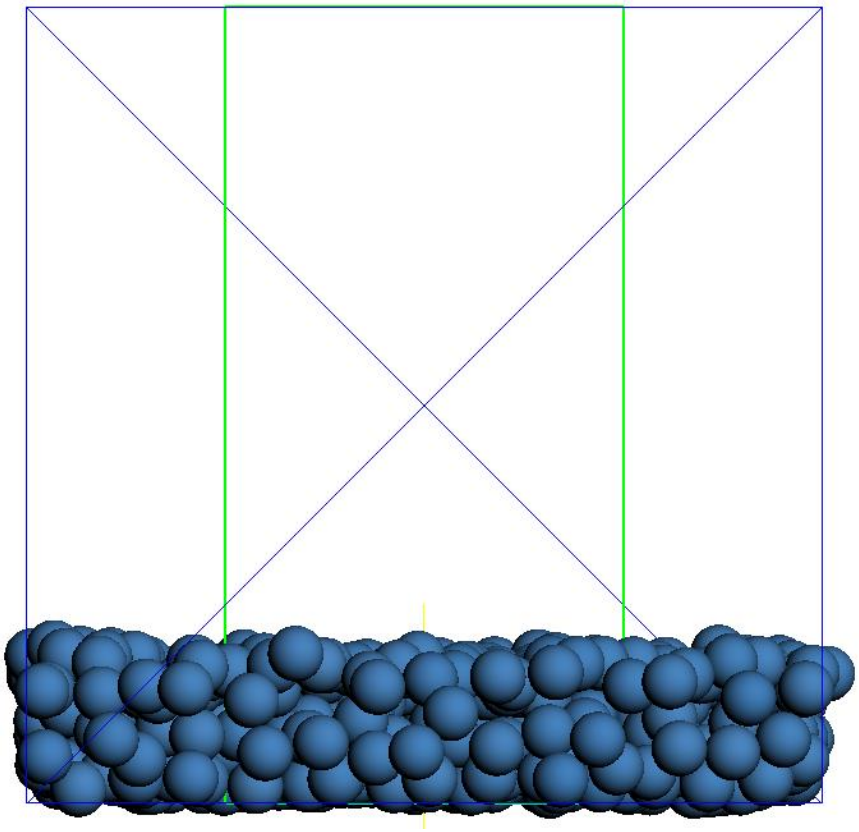


Figure 40: Small box simulation at the last time step.

Based on Figure 40, the particles are clearly not arranged in a SC arrangement. The particles do not seem to align themselves with any specific arrangement structure. Therefore, it does not make sense why the Neutrino measurement tends to follow the SC arrangement.

9 Neutrino Fluid Height Bias

The Neutrino height result was analyzed to confirm the suspicion that it uses an SC arrangement. To confirm, a single time step for the small box model was analyzed. The height at the last time step obtained from Neutrino was 0.101 m. Next, the number of particles in the measurement field at the last time step was obtained from the Neutrino GUI. There were 145 particles inside the measurement field. Now, assuming the SC arrangement, the total volume of the fluid was calculated using Eq. (23), the optimum particle spacing, and the number of particles. This resulted in a total volume of ~0.006 m^3. Using the measurement field width and length (0.25 m each), the fluid depth was calculated to be 0.101 m. Therefore, the Neutrino height result is calculated by determining the number of particles inside the measurement field and assuming a SC arrangement to determine the height.

Now that it is known how the Neutrino height is calculated, it can be shown that this proves to be an issue. The first issue is the fact that it is based on the number of particles in the field. To show this issue, Figure 41 shows a 2D set of particles with different measurement fields.

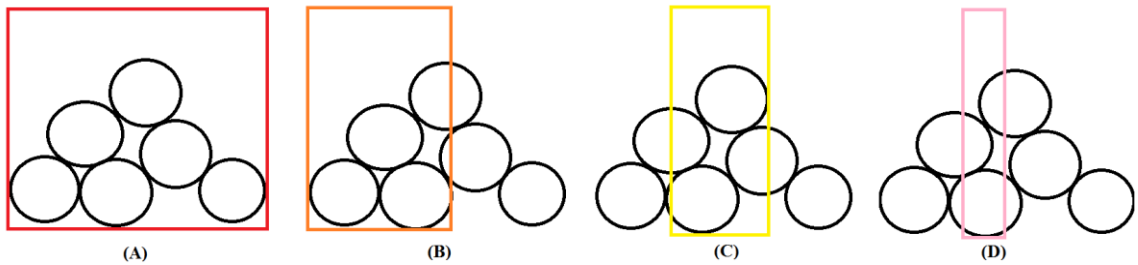


Figure 41: Example of Neutrino height issue.

In image (A), all particles are within the red measurement field. Therefore, both visually and with Neutrino, there would be six particles inside the measurement field. In image (B), there are approximately 3.75 particles inside the orange measurement field. However, Neutrino would say that there are 4 particles inside the field since more than half of one particle is inside the measurement field and less than half of the other is outside the measurement field. In (C), there are approximately 3 particles inside the yellow measurement field. However, Neutrino would say that there are 4 particles since half of the two particles are inside the measurement field. Lastly, in (D), there are approximately 1.5 particles inside the pink measurement field. However, Neutrino would say there is one particle. This shows that by basing the height result off the number of particles, the Neutrino result will be an estimation and not accurately account for the number of particles inside the measurement field.

The next issue is the simple cubic assumption. This assumption causes the Neutrino height result to be higher than the actual particle positions since the particles do not arrange themselves in the SC arrangement as shown in Figure 40. To show this overestimation, the last time step of the small box was analyzed. Again, Neutrino says that the fluid height is 0.101 m. To prove the overestimation, each particle's position in the measurement field was obtained. Figure 42 shows a plot of the Neutrino height result compared to all of the particles positions. Particles may look like they are on top of each other in the plot, but that is due to the fact that the plot is in 2D instead of 3D.

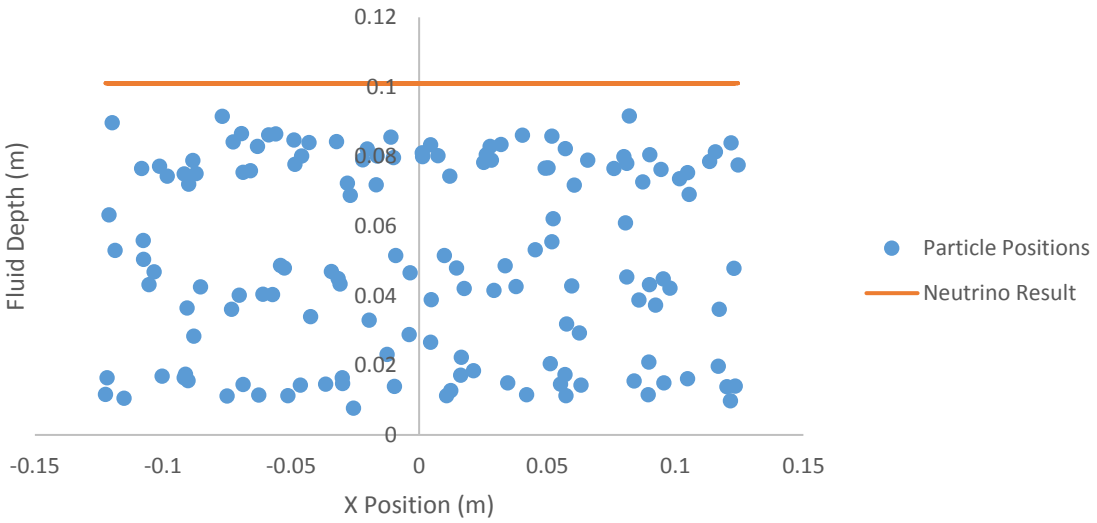


Figure 42: Particle position vs. Neutrino result comparison plot.

As shown above, none of the particles are on the Neutrino result line. They are all below. This shows that the SC arrangement assumption overestimates the fluid height.

Overestimating the fluid height is an issue for several reasons. First, it is not providing a result that matches the actual simulation. This can cause an issue with validating Neutrino either by providing a result that is close to experimental data when in fact it is off or vice versa. Second, if Neutrino is to be used to simulate flooding at NPPs, the water level needs to be accurate to determine the possible damage. Some might argue that overestimating the fluid height would provide a conservative answer and therefore is okay. However, overestimating the fluid height could cause additional damage.

To understand the impact of overestimating the fluid height, two examples are provided. The first example is simulating a flooding event with a door failure given by a specific height. By overestimating the fluid height, the simulation would say that the door would fail earlier than it should. This could cause an issue especially if people are involved. Assume that an internal flood is being simulated and that the Neutrino simulation says the

door fails at 5 minutes. This would tell first responders that after 5 minutes, they could enter the building because most of the water would have exited through the doorway. However, in reality, the door does not fail until 7 minutes. This could result in first responders being at the doorway the moment that the door fails resulting in injuries or worse.

The second example is regarding the water level inside a pressure vessel. Assume that a leak in a pressure vessel is being simulated using Neutrino with the goal being to determine when core damage would occur. With Neutrino's overestimation, it might say that core damage occurs at 30 minutes. However, the water level is actually lower than the Neutrino result causing core damage to occur at 20 minutes. These two examples show that overestimating the fluid height is an issue and can cause serious consequences to occur.

The last issue is the fact that Neutrino is a proprietary code. This means that only the Neutrino developers have access to the source code and could change the SC assumption. Since Neutrino is proprietary, it limits what can be done and how since only the existing capabilities can be used.

10 Improved Fluid Height Analysis Method

Since it is now known that there is a bias with the Neutrino height result, an improved fluid height analysis method needed to be developed. However, with Neutrino being proprietary, it was a challenge to come up with a method using Neutrino's existing capabilities. To determine an improved height method, the different measurement field output options were analyzed. The measurement field can output velocities, pressures, flow rates, temperatures, etc., but it is unknown if these measurements also make an assumption when calculating the result. Additionally, you can export information per particle. The per particle information provides the particle ID, x, y, z position, x, y, z velocity, density, pressure, and temperature for every particle in the measurement field. An example of a per particle output file is provided in Appendix F: Improved Fluid Height Analysis Method Files.

The position of each particle in the measurement field can be used to determine the height. The position of each particle comes from the equations of motion so it is expected that there should not be any assumption made on the arrangement of the particles.

While the position of each particle can be obtained, it does not provide the height of the fluid. Therefore, a method needed to be developed to determine the height of the fluid based on the particles' positions. Another issue is the fact that a new file is created every time step with all the particle's information. This is an issue because it will increase the storage space required for each simulation and it will take a longer amount of time to post process the data to get the result.

Fortunately, one of the capabilities of Neutrino is the ability to add a dynamic expression script. A dynamic expression script is ran every time step of the simulation. Therefore, an improved fluid height analysis method can be developed as a Python script and then added to Neutrino as a dynamic expression script to provide the fluid height based on the particles' positions.

The first step was to develop the method to determine the fluid height based on the particles' positions. In other words, the surface profile of the fluid is wanted so that either the average fluid height can be calculated or the entire surface profile can be extracted. The surface profile is determined by separating the measurement field into bins. The maximum particle position in each bin is saved to determine the surface profile. Figure 43 shows a 2D example with the bins, selected by the user, shown in blue and the maximum particles for each bin shown with a red dot.

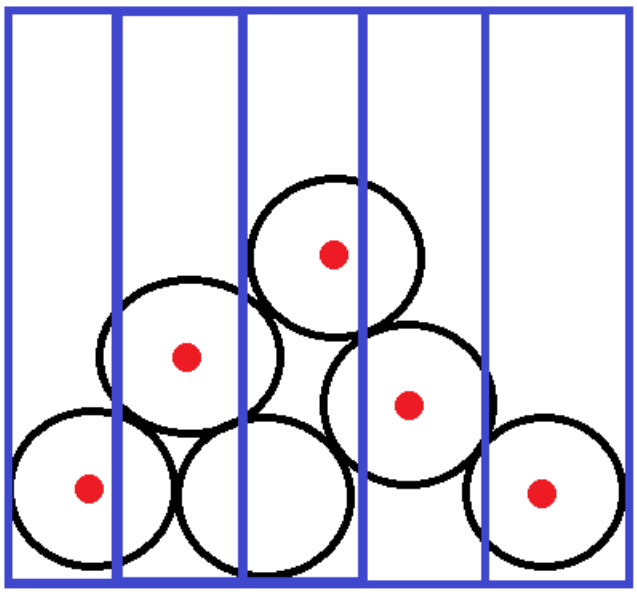


Figure 43: 2D improved fluid height analysis example.

While this provides a 2D example, the same methodology can be applied in three-dimensions (3D) to obtain a 3D surface profile. For the 2D methodology, two axes of interest are selected, i.e. x and y. The z axis would be ignored and not considered. Therefore, for 2D cases, the maximum particle position in a bin is determined by making sure the x position is within the bin, but not considering the z position. The average fluid height position can then be calculated by averaging the maximum particle positions together. One thing that needs to be understood though is that the maximum positions are exported and not the fluid height. To get the fluid height, the coordinate system of the simulation setup needs to be included.

The improved fluid height analysis method uses the particle position to calculate the surface profile or average fluid height. The particle position is the center position of the particle if the particle diameter is being rendered using the particle spacing. The particle position was used to calculate the surface profile or average fluid height since information is known at that location. However, the improved fluid height analysis method could easily be modified so that the particle radius is included in the height calculations. Investigation into the reason for including the particle radius is left for future research.

The benefit of Neutrino's dynamic expression scripts is that the improved fluid height analysis method script can be added as a dynamic expression script. The script also contains code to perform the analysis as the simulation is running. It reads the per particle output for a time step, deletes the per particle output file to save storage space, performs the analysis, and then exports the wanted results to a new output file. Figure 44 shows the flow chart for the 2D surface profile extraction script per time step, Figure 45 shows the flow chart for the 3D surface profile extraction script per time step, and Figure 46 shows

the flow chart for the average height script per time step. Each of these Python scripts can be found in Appendix F: Improved Fluid Height Analysis Method Files.

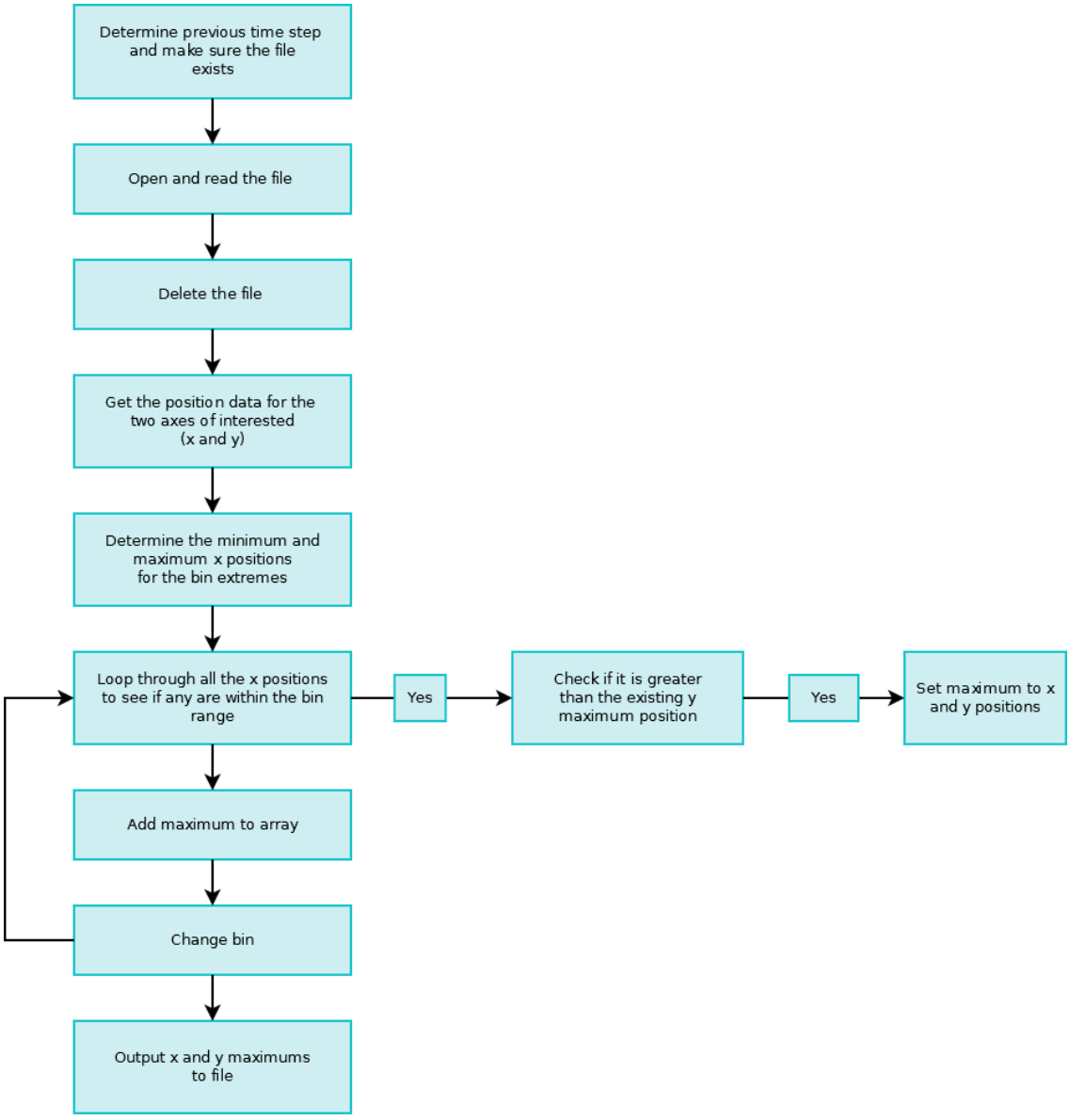


Figure 44: 2D surface profile flow chart.

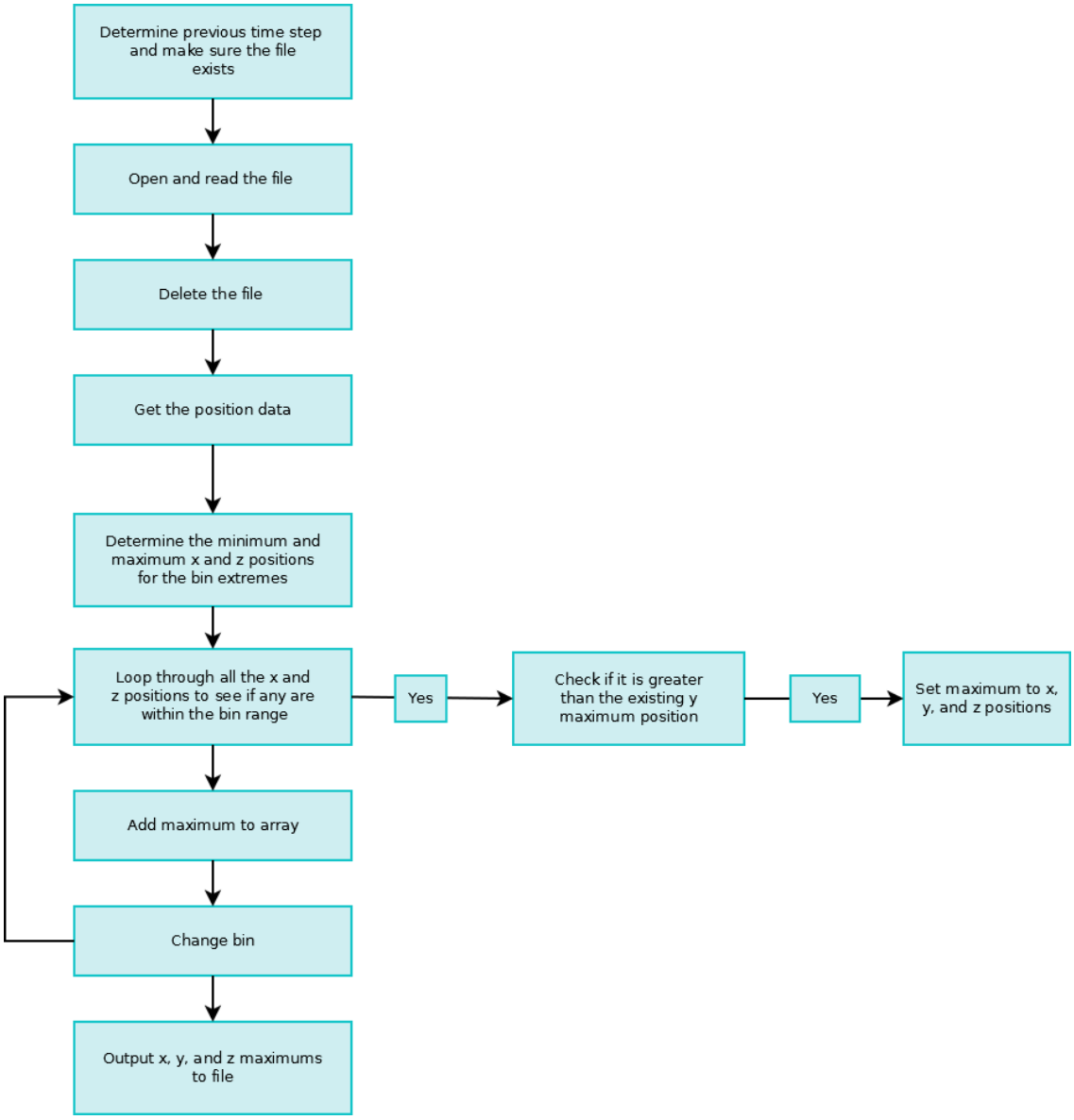


Figure 45: 3D surface profile flow chart.

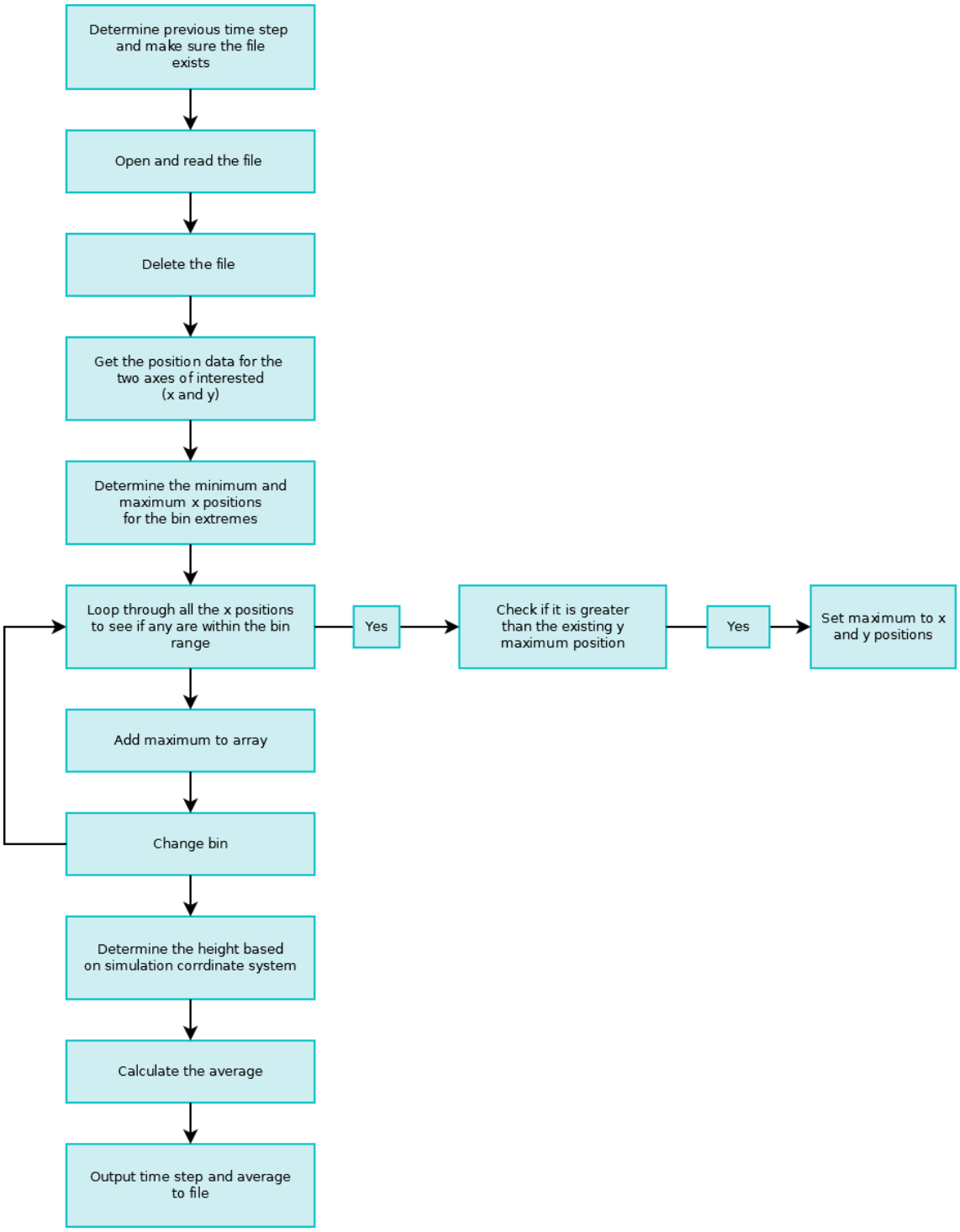


Figure 46: Improved average height flow chart.

Once the improved fluid height methodology was developed, the correct bin size needed to be determined. If the bin size is too small, the method will find particles that

are lower than the surface profile since it would be the maximum in that bin size. If the bin size is too large, the method would not provide enough detail of the surface profile. The 2D surface profile was analyzed first for different bin sizes. The improved average height method uses the same methodology as the 2D surface profile so the same bin size for the 2D surface profile is used for the improved average height script. The last time step of the small box model was analyzed. Figure 47 through Figure 50 shows the 2D surface profile result in red for four different bin sizes with all of the particle positions shown in black.

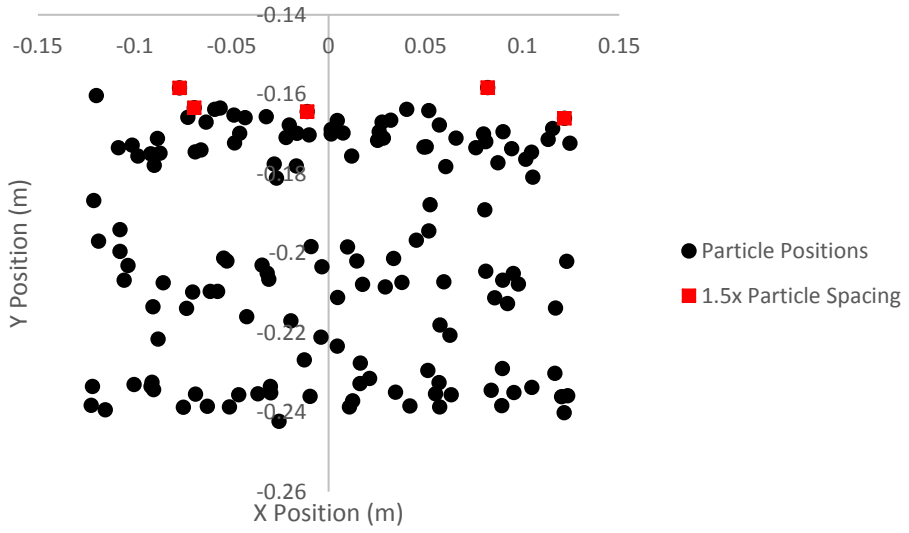


Figure 47: 2D surface profile using 1.5x particle spacing bin size.

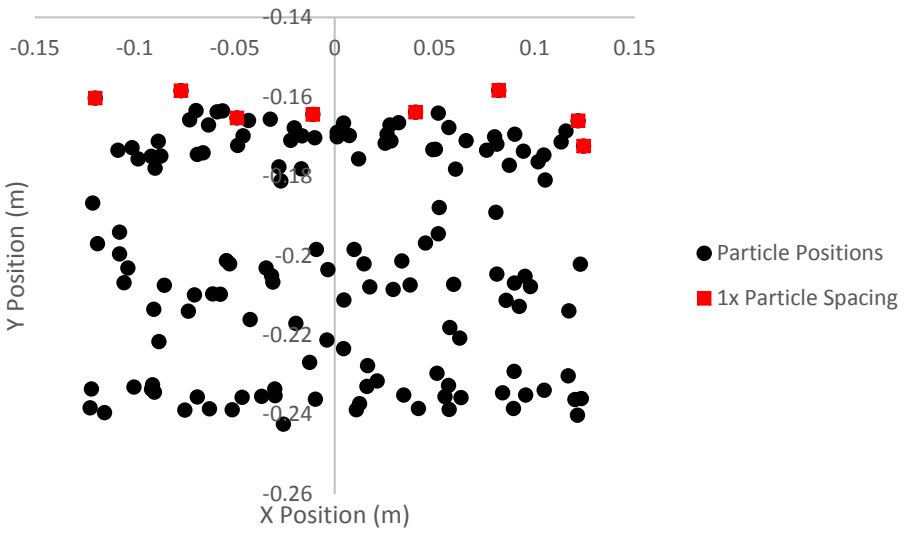


Figure 48: 2D surface profile using 1x particle spacing bin size.

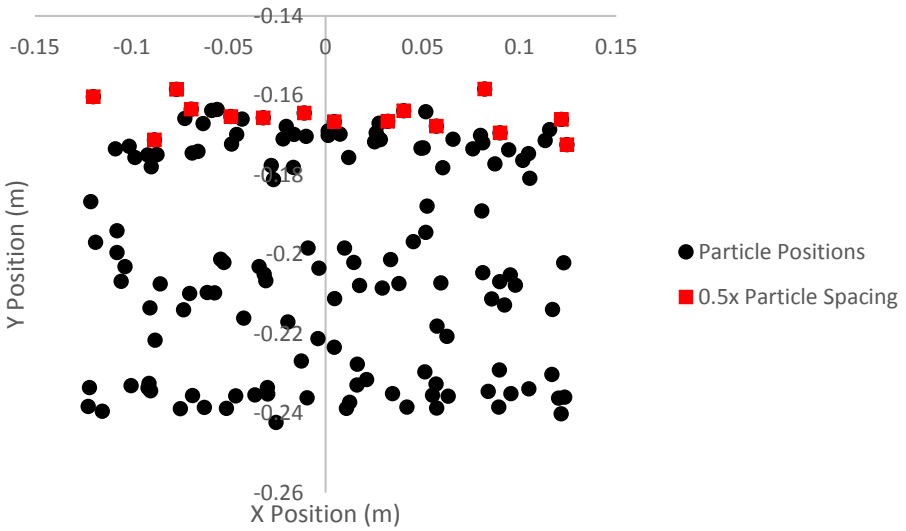


Figure 49: 2D surface profile using 0.5x particle spacing bin size.

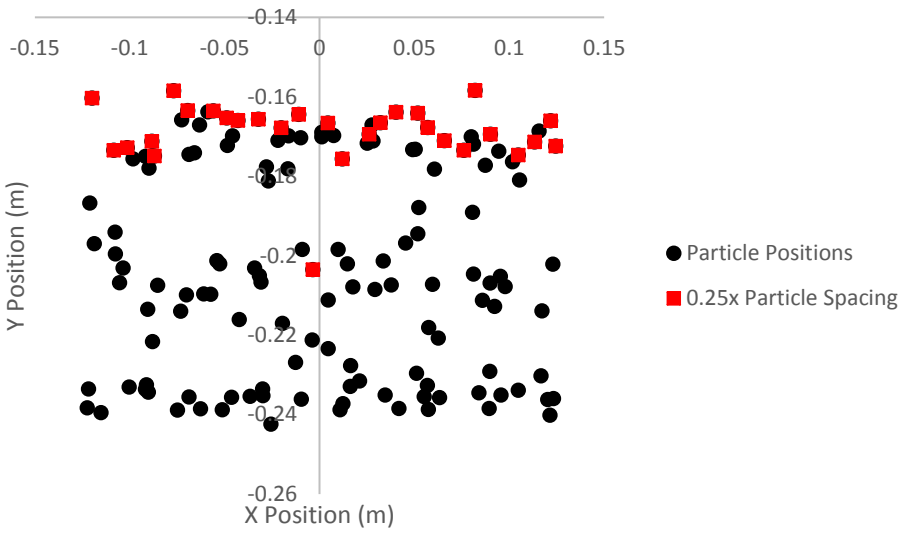


Figure 50: 2D surface profile using 0.25x particle spacing bin size.

The 1.5x and 1x particle spacing bin sizes are too large and do not provide enough resolution of the surface profile. The 0.25x particle spacing bin size is too small because it is finding positions that are below the surface profile. Therefore, the 0.5x particle spacing bin size provides the best result for the 2D surface profile, and improved average.

Next, the same analysis was performed for the 3D surface profile to determine the appropriate bin size. Figure 51 through Figure 54 shows the 3D surface profile in red for four different bin sizes with all of the particle positions shown in black. The 3D plotter script can be found in Appendix F: Improved Fluid Height Analysis Method Files.

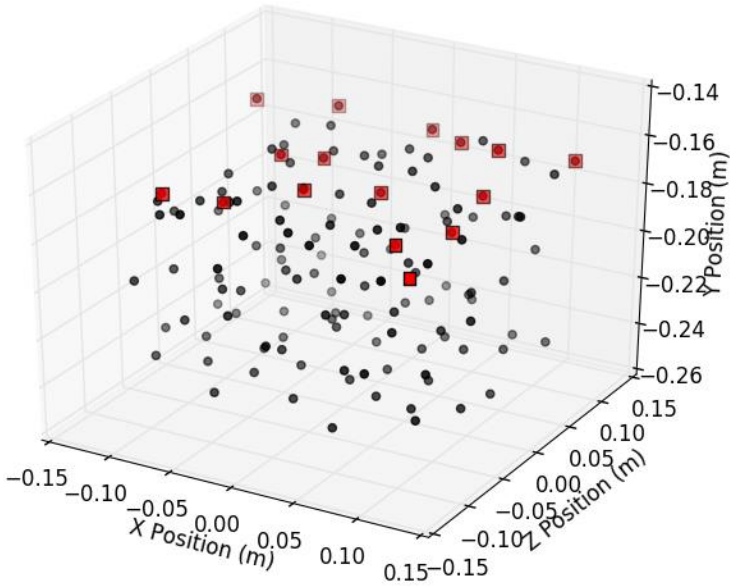


Figure 51: 3D surface profile using 2x particle spacing bin size.

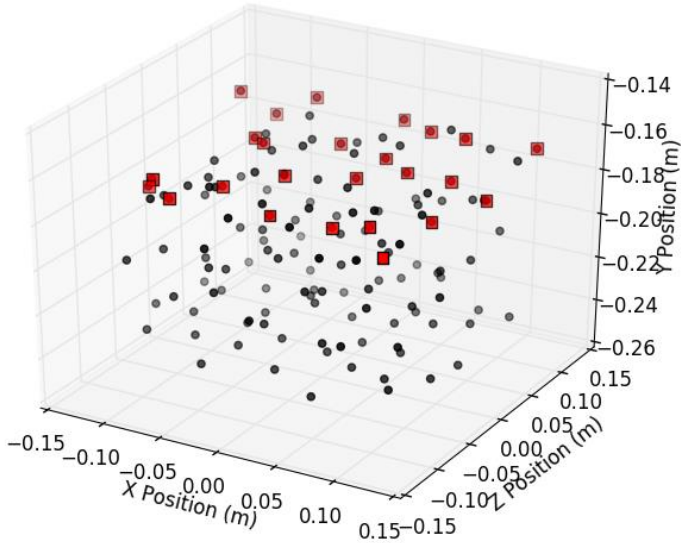


Figure 52: 3D surface profile using 1.5x particle spacing bin size.

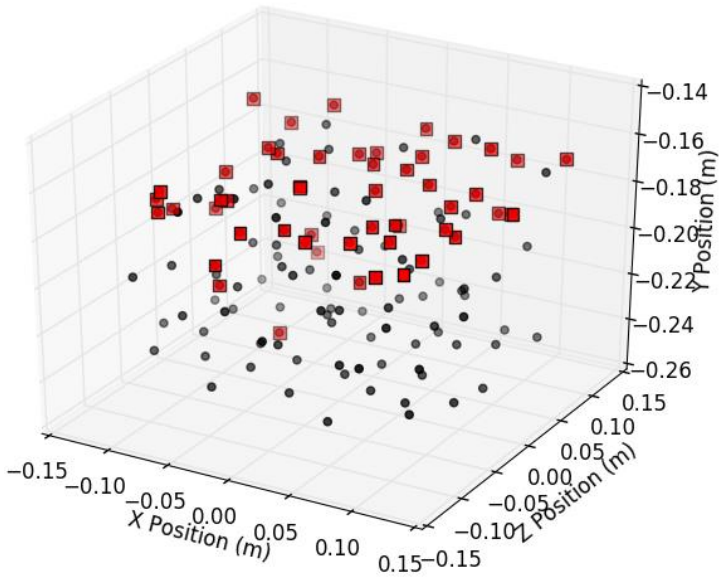


Figure 53: 3D surface profile using 1x particle spacing bin size.

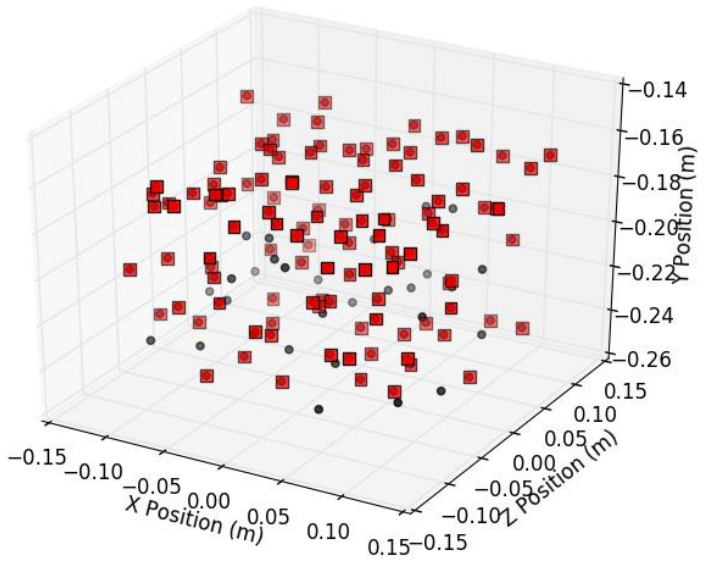


Figure 54: 3D surface profile using 0.5x particle spacing bin size.

The 1x and 0.5x particle spacing bin sizes are too small for the 3D surface profile because it is providing results that are below the surface. The 2x particle spacing is too large since it does not provide enough resolution of the particle spacing. Therefore, for 3D surface profiles, a bin size 1.5x particle spacing should be used.

The advantage of using the surface extraction scripts, besides not making an SC arrangement assumption, is that it allows the user to obtain a surface profile by using only one measurement field. If only the Neutrino measurement field capabilities were used, then multiple measurement fields calculating the average height would have to be placed to determine the surface profile. This could result in tens to hundreds of measurement fields depending on the size and resolution needed. However, the surface extraction scripts can obtain this information with the appropriate resolution using a single measurement field.

While the improved fluid height analysis method is advantageous, there are a couple of places where it might not work. One area is if the particle spacing is fairly large and the measurement field is small. The improved height methodology might have trouble finding the surface profile if there are not many particles inside the measurement field. The other area it would have an issue is if the surface profile contains splashing. The improved methodology would find the splashed particles as the maximum resulting in those being used as the surface profile. This problem could be resolved by adding a check to the surface extraction scripts to make sure that there are other particles near the maximum particle. However, for this work, splashing does not occur, so a check was not implemented in the surface extraction scripts.

11 Validation of the Improved Fluid Height Analysis Method

Once the improved fluid height analysis method was developed, it needed to be validated. Validation is the process of determining if the model represents the real world application [52]. To validate the improved fluid height analysis method, the ogee spillway experiment was simulated. This experiment was previously simulated using Neutrino with extensive details provided in "Comparison of free surface flow measurements and smoothed particle hydrodynamic simulation for potential nuclear power plant flooding simulation" by Emerald D. Ryan, Bruce M. Savage, Curtis L. Smith, and Chad L. Pope [25]. A brief description of the experiment and simulation setup is provided below.

The physical experiment used for this comparison was conducted using a scaled model of an ogee spillway with a horizontal apron [53] [54]. The scaled model experiment was conducted at the Utah Water Research Laboratory (UWRL) in Logan, Utah. The spillway was constructed of Plexiglas so smooth curves could be obtained and pressure taps could be easily installed along the spillway. The ogee spillway was approximately 0.80 m tall, 1.83 m wide, and 1.36 m long and was placed in a flume with approximate dimensions of 12 m long, 1.83 m wide, and 1.22 m high.

The experiment consisted of setting different values of the upstream head, measured approximately 2 m upstream, and then taking flow rate and pressure measurements. The flow rate measurements were taken using weight tanks, volumetric tanks, or an ultrasonic flowmeter. Different flow rate measurement tools were used to measure different flow rate quantities. The weight tanks were used at low flow rates, the volumetric tanks were used for the intermediate to high flow rates, and the ultrasonic flowmeter was used at the two highest flow rates.

Once the experiment for the scaled model was completed and the results recorded, the results were then scaled up 30 times to prototype scale. One difference between the scaled model and the prototype is that the prototype has a silt level behind the spillway. The silt level is essentially the ground level behind the spillway after fine sand and clay has built up. By adding the silt level to the SPH simulation, it provides a more realistic simulation since the silt level would be found behind existing spillways. The prototype scale was used for the SPH comparison since that scale represents the size of an actual spillway.

The Neutrino model was comprised of the following components: a rigid custom (spillway), a rigid box (flume), 1 rigid cuboid (silt level), the Neutrino IISPH fluid solver, 2 square particle emitters to fill the area behind the spillway, 1 flow particle emitter to set the flow rate, 1 measurement field to measure the total head 60 m upstream, and 2 particle killers to remove particles at the end of the spillway and any particles that might leak from the system. The difference between the square particle emitter and the flow particle emitter is that the velocity of the particles is set in the square particle emitter while the flow rate is set in the flow particle emitter. Figure 55 shows a wireframe representation of the Neutrino model. Fluid particles were introduced on the left side of Figure 55 to fill the reservoir and to set the flow rate. Once particles have passed over the spillway they were killed since they can no longer contribute to the total head at the measurement field.

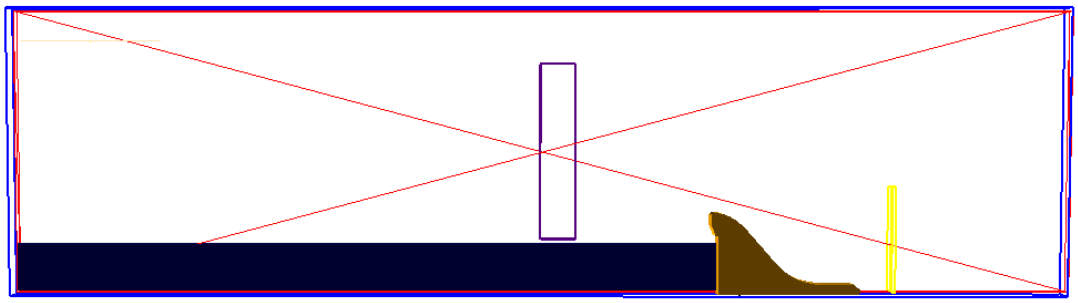


Figure 55: Neutrino spillway model.

Two parametric studies were performed to determine certain properties of the model. The two parametric studies were on the width of the model and the particle spacing of the fluid. The goal of these parametric studies was to determine the most computationally efficient model. The parametric studies resulted in using a model width of 4 m and a particle spacing of 0.35 m.

Next, the ten experimental runs were simulated using Neutrino. All the simulations were rerun since a different Neutrino version was used from the one in [25]. The total head needed to be calculated and Eq. (25) was used.

$$H_T = H_0 + \frac{u^2}{2g} \tag{25}$$

where H_T is the total head, H_0 is the water elevation, u is the x-direction velocity, and g is gravity.

To calculate the total head of the simulation, the average height and x-direction velocity was needed. The x-direction velocity was obtained from the Neutrino measurement field. However, the average height was calculated two ways. One way used the Neutrino measurement field result and the other way used the improved fluid height

analysis method. The reason for using both ways is to determine the difference between the two results and see what method provides an answer closer to the experimental results.

Table 12 shows the experimental results, Neutrino results, improved results, and the relative percent error for each method compared to the experimental result, Figure 56 shows the plot of total head vs. flow rate for the experimental, Neutrino, and improved height results, and Figure 57 shows the plot of total head relative percent errors between the SPH and experimental results for both the Neutrino and improved height results.

Table 12: Spillway Results Comparison

Run	Flow Rate ($m^3/s/m$)	Exp. Total Head (m)	Neutrino Total Head (m)	Improved Total Head (m)	Neutrino Relative Error (%)	Improved Relative Error (%)
1	1.90	24.27 ± 0.25 %	25.34	24.87	4.40	2.49
2	6.03	25.33 ± 0.25 %	26.43	26.06	4.34	2.87
3	12.29	26.48 ± 0.25 %	27.74	27.31	4.76	3.13
4	19.02	27.37 ± 0.25 %	28.90	28.44	5.58	3.92
5	27.92	28.47 ± 0.25 %	30.44	29.96	6.94	5.24
6	37.79	29.50 ± 0.25 %	31.92	31.38	8.19	6.37
7	48.24	30.44 ± 0.25 %	33.24	32.67	9.21	7.31
8	58.86	31.35 ± 0.25 %	34.42	33.83	9.81	7.92
9	73.77	32.42 ± 0.5 %	35.96	35.35	10.91	9.04
10	89.90	33.45 ± 0.5 %	37.43	36.82	11.91	10.07

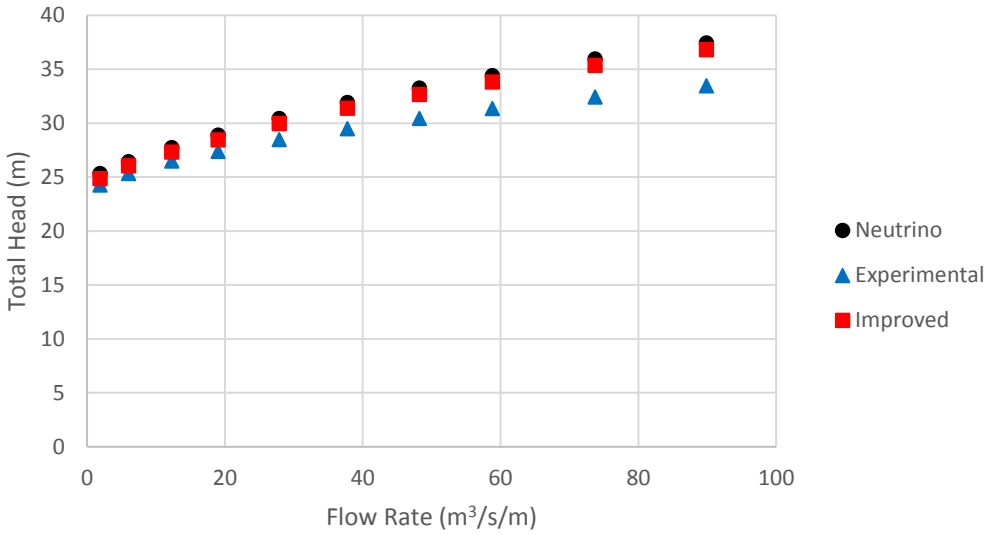


Figure 56: Spillway total head vs. flow rate comparison plot.

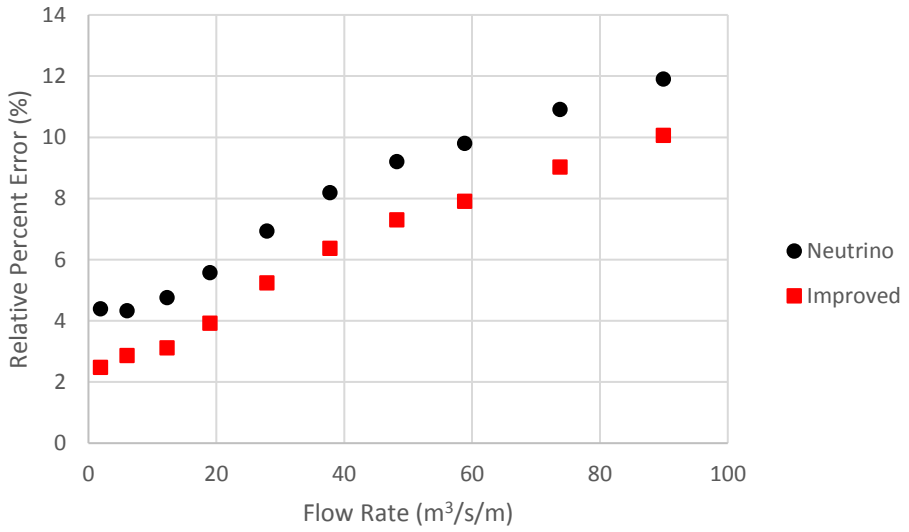


Figure 57: Spillway relative error vs. flow rate comparison plot.

The comparison results show that the improved fluid height analysis method always provided results closer to the experiment than the Neutrino results. Table 13 shows the relative percent improvement for each run by using the improved fluid height analysis method.

Table 13: Relative Percent Improvement Using Improved Fluid Height Analysis Method

Run	Relative Percent Improvement (%)
1	43.41
2	33.87
3	34.24
4	29.75
5	24.50
6	22.22
7	20.63
8	19.27
9	17.14
10	15.45

The improved method provides between a 15% to 43% improvement compared to the Neutrino results. Therefore, this validates the improved fluid height analysis method and shows that it does provide realistic results. For the output files for the spillway comparison, please refer to Appendix G: Spillway Files.

12 Optimum Particle Spacing Determination Using the Improved Fluid Height Analysis Method

12.1 Adjustments

Now that an improved fluid height analysis method has been developed and validated, the determination of the optimum particle spacing for the three models can be redone. To use the improved method, a few changes needed to be made to the convergence optimizer so that the dynamic expression script can be used. The first change was made to the Main file. Instead of specifying the maximum time in seconds, the maximum time is specified in time steps since the improved average height script outputs the information as time step and average height.

Next, the command prompt class was modified. The portion modified was the command to run Neutrino. Since a dynamic expression script is used, a Python file must be created to tell Neutrino what Neutrino input file to run and where to attach the dynamic expression script. Table 14 shows the Python file for the small box model.

Table 14: Python File for Small Box Model

```
Base.open("E:/Emerald/Optimizer/Test Code/Small_Box.nescene")
with open("E:/Emerald/Optimizer/Test Code/Average.py","r") as myfile:
    data=myfile.read()
MeasurementField_1.setPropertyUpdateExpression("Position", data);
```

The first line tells Neutrino where to find the Neutrino input file. The next lines tell Neutrino where to find the dynamic expression script and to open and read the file. The last line tells Neutrino to apply the dynamic expression script to the measurement field position property. The improved height analysis method script can be applied to any

property. It was applied to the position property because it is known to work and no other script needed to be added to the position.

Once the convergence optimizer was adjusted to handle the dynamic expression script, the determination of optimum particle spacing could be redone with the improved fluid height analysis method. The details of the convergence optimizer mostly remained the same as shown in section 6.1. A few exceptions are the starting particle spacing and simulation time. The starting particle spacing was changed to ensure the particle spacing was not so large that it created an issue with the improved height method. The simulation time was extended in the large box model due to the particles taking longer to settle. The total simulation time for the large box was changed to 90 seconds with results averaged for the last 10 seconds. Table 15 shows the starting particle spacing for the three models using the improved height method.

Table 15: Model Starting Particle Spacing for Improved Method Optimization

Model	**Starting Particle Spacing (m)**
Small	0.1
Medium	2
Large	5

12.2 Results

Each model was reanalyzed using the convergence optimizer and the improved fluid height analysis method. Table 16 shows the optimum particle spacing found for each model, Figure 58 through Figure 60 show plots of fluid height vs. particle spacing for each

model with the optimum particle spacing shown in red, and Figure 61 through Figure 63 show plots of computational runtime vs. particle spacing for each model with the optimum particle spacing shown in red. The complete details of each model can be found in Appendix H: Optimum Particle Spacing Details Using Improved Method.

Table 16: Optimum Particle Spacing for Models Using Improved Method

Model	**Optimum Particle Spacing (m)**
Small	0.008928
Medium	1.000000
Large	2.500000

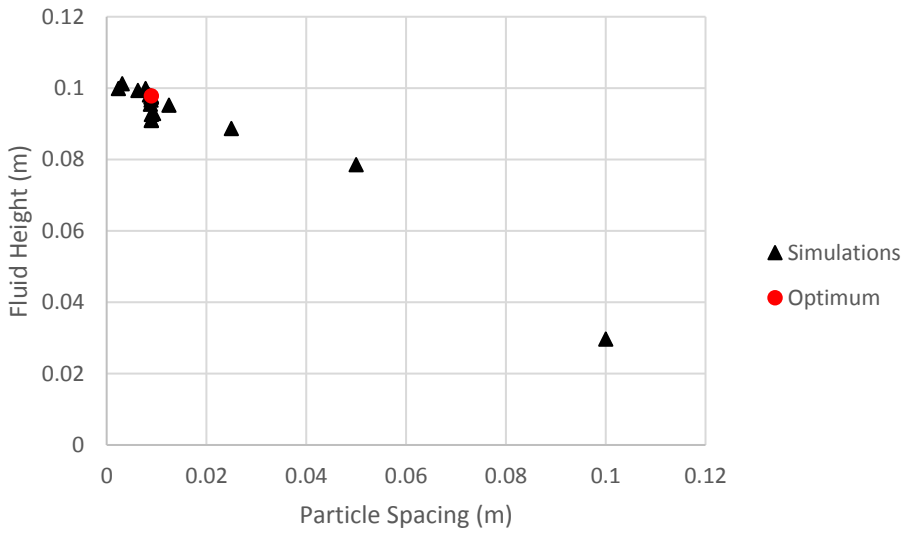


Figure 58: Small box fluid height vs. particle spacing plot using improved height result.

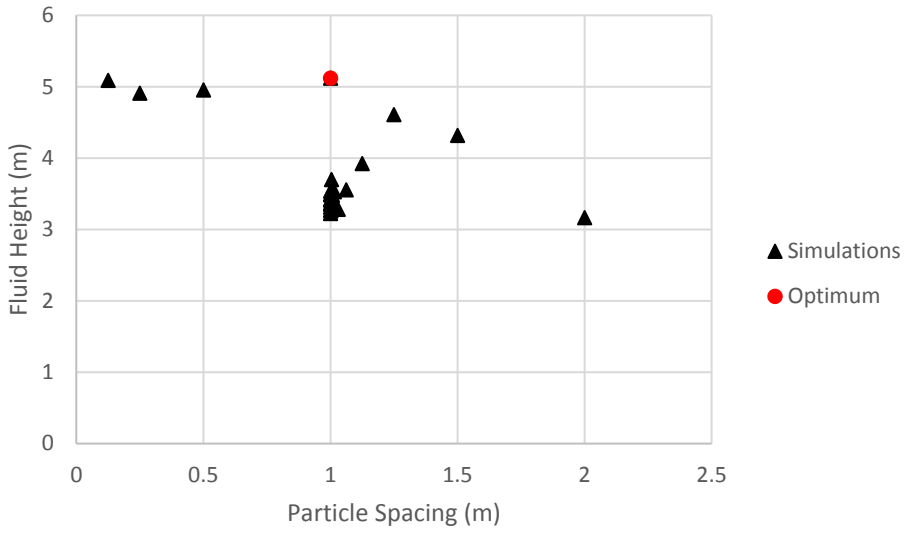


Figure 59: Medium box fluid height vs. particle spacing plot using improved height result.

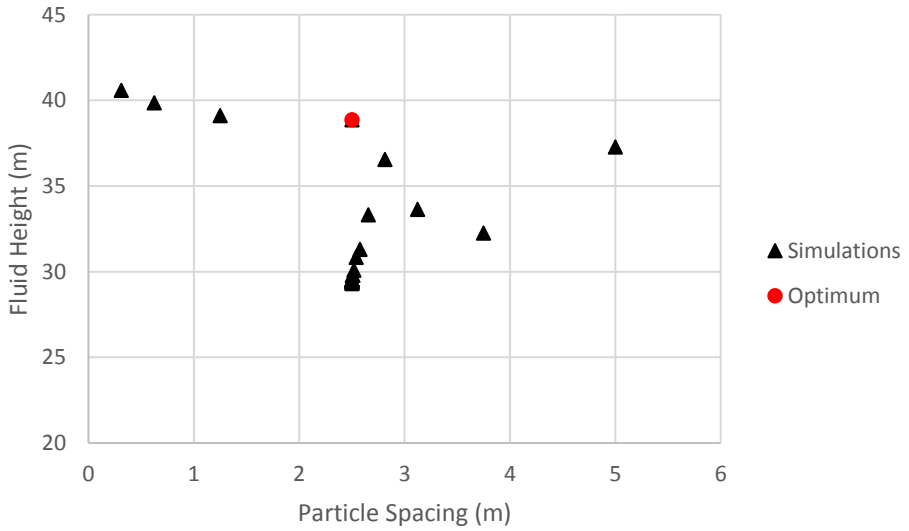


Figure 60: Large box fluid height vs. particle spacing plot using improved height result.

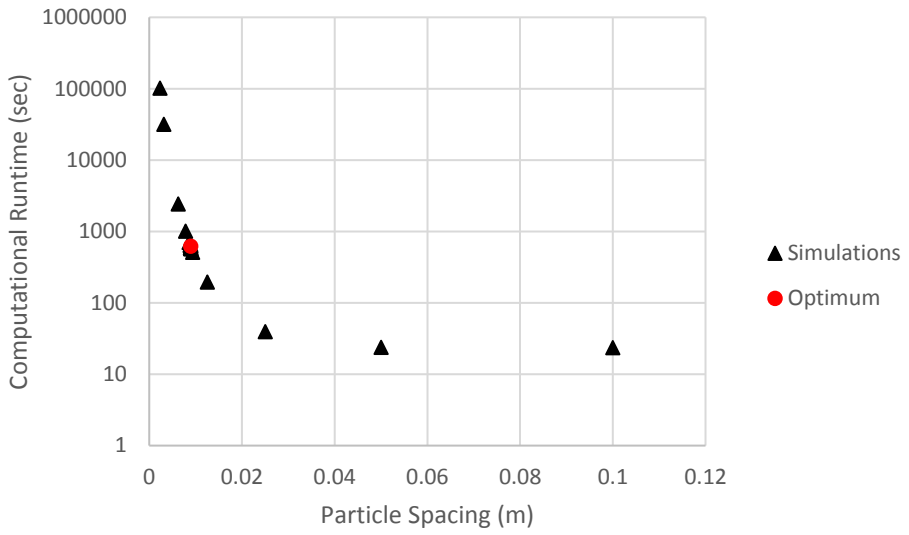


Figure 61: Small box computational runtime vs. particle spacing plot using improved height result.

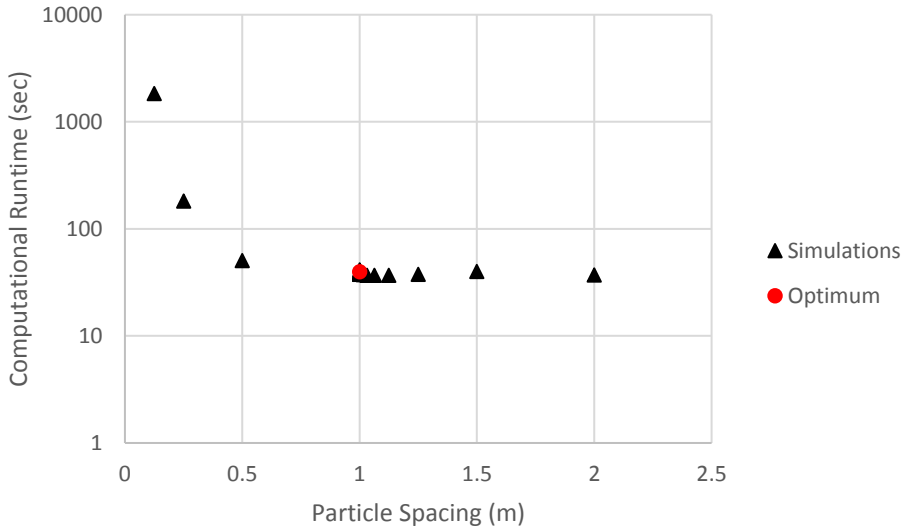


Figure 62: Medium box computational runtime vs. particle spacing plot using improved height result.

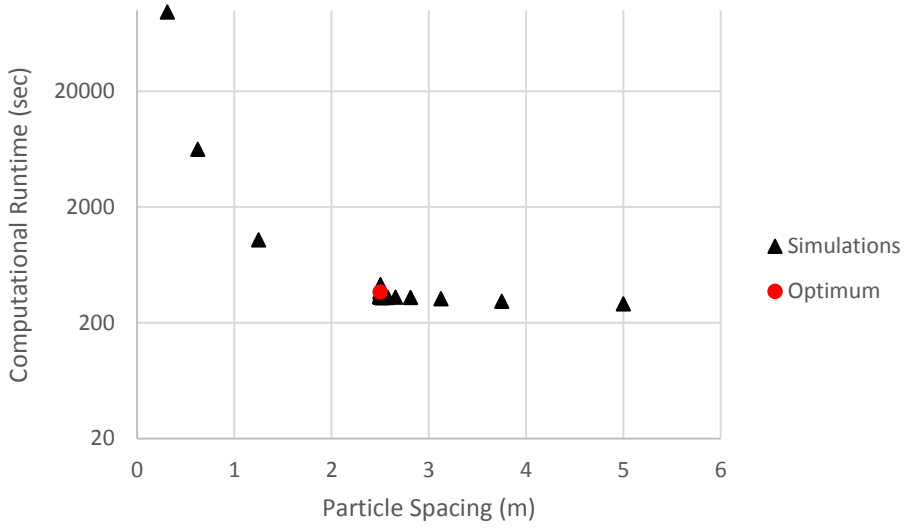


Figure 63: Large box computational runtime vs. particle spacing plot using improved height result.

12.3 Protocol Determination and Verification

Once the optimum particle spacing values for the three models were obtained, they were analyzed to determine a particle spacing protocol. Since the parameter being varied in each model was the scale, the smallest dimension was used for comparison with the optimum particle spacing. Table 17 shows the smallest dimension for each model.

Table 17: Model Smallest Dimension Details

Model	Smallest Dimension (m)	Smallest Dimension Location
Small	0.1	Fluid Height
Medium	5	Fluid Height
Large	30	Box Width and Length

The smallest dimension of the model was then divided by the optimum particle spacing to determine the number of particles associated with the smallest dimension. Table 18 shows the optimum particle spacing, smallest dimension, and number of particles per dimension for each of the models.

Table 18: Model Particles Per Dimension Details

Model	**Optimum Particle Spacing (m)**	**Smallest Dimension (m)**	**Number of Particles Per Dimension**
Small	0.008928	0.1	11.2
Medium	1.0	5	5
Large	2.5	30	12

Based on the results, the small and large model require ~12 particles per the smallest dimension. The medium model only required five particles per the smallest dimension. Therefore, to provide a converged result for other simulations, the particle spacing protocol requires 12 particles per the smallest dimension.

After the particle spacing protocol was determined, the protocol was verified to determine if the protocol provided a converged simulation result. Verification is the process of determining if the computational model fits the mathematical description [52]. For this work, the verification is to determine if the computational result fits the particle spacing protocol. Three additional simulation models, two outside the original model scale range and one in the middle, were analyzed using the particle spacing protocol and the convergence optimizer. The three simulation models are called the following:

- Small test

- Medium test
- Large test

Table 19 provides the details of each of the three test simulation models.

Table 19: Test Model Details

Model	Rigid Box Length and Width (m)	Analytical Fluid Height (m)	Flow Particle Emitter Size (m)	Flow Rate (m^3/sec)	Emitter Time On (sec)	Simulation Time (sec)
Small Test	0.03 x 0.03	0.06	0.02 x 0.02	0.000018	3	12
Medium Test	17.5 x 17.5	17.5	10 x 10	1,786	3	130
Large Test	55 x 65	52	50 x 50	61,967	3	30

For each test model, some of the convergence optimizer parameters were the same. The following is a list of the parameters and their values that were the same for each test model:

- Minimum number of iterations = 3
- Maximum number of iterations = 25
- Convergence criterion = 0.05
- Number of converged iterations = 3
- Reduction method = 0.5
- Minimum particle spacing change = 0.000001

The rest of the convergence optimizer details can be found in Table 20.

Table 20: Test Model Convergence Optimizer Details

Model	Starting Particle Spacing (m)	Time Step to Start Averaging
Small Test	0.01	500
Medium Test	5	6,400
Large Test	20	1,400

Next, each test model was analyzed using the convergence optimizer and particle spacing protocol. Table 21 shows the optimum particle spacing found for each test model and the suggested particle spacing from the protocol, Figure 64 through Figure 66 show plots of fluid height vs. particle spacing for each test model with the optimum particle spacing shown in red, and Figure 67 through Figure 69 show plots of computational runtime vs. particle spacing for each test model with the optimum particle spacing shown in red. The complete details of each test model can be found in Appendix I: Particle Spacing Protocol Verification Test Details.

Table 21: Optimum Particle Spacing and Protocol Particle Spacing for Test Model

Model	Optimum Particle Spacing (m)	Protocol Particle Spacing (m)
Small Test	0.005	0.0025
Medium Test	2.5	1.46
Large Test	10.0	4.33

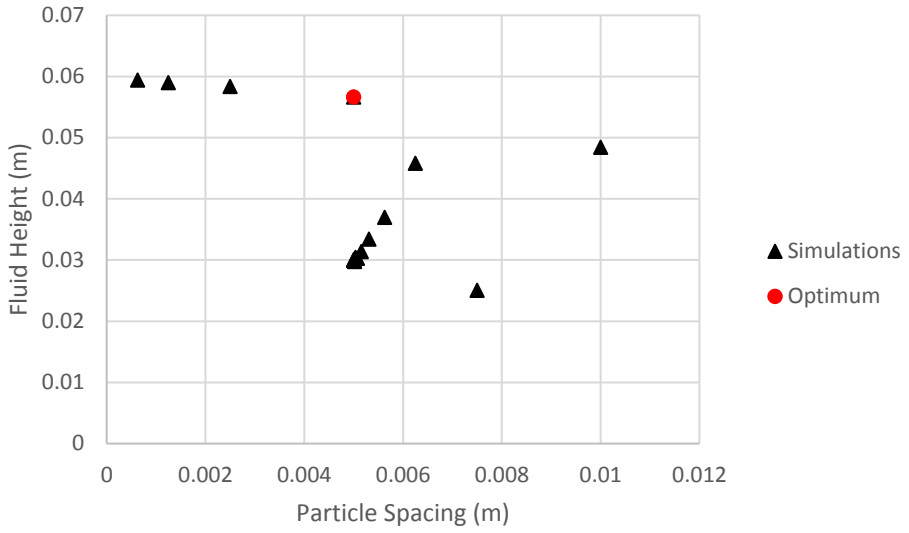


Figure 64: Small test box fluid height vs. particle spacing plot using improved height result.

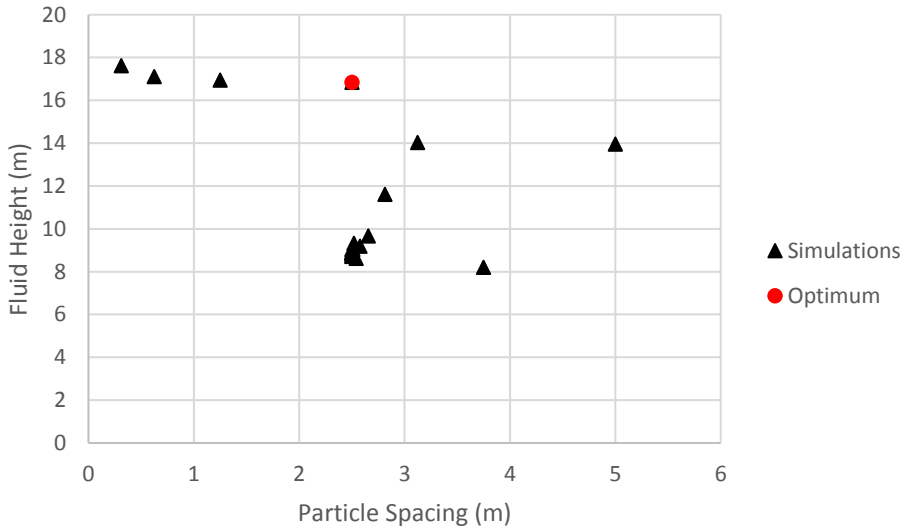


Figure 65: Medium test box fluid height vs. particle spacing plot using improved height result.

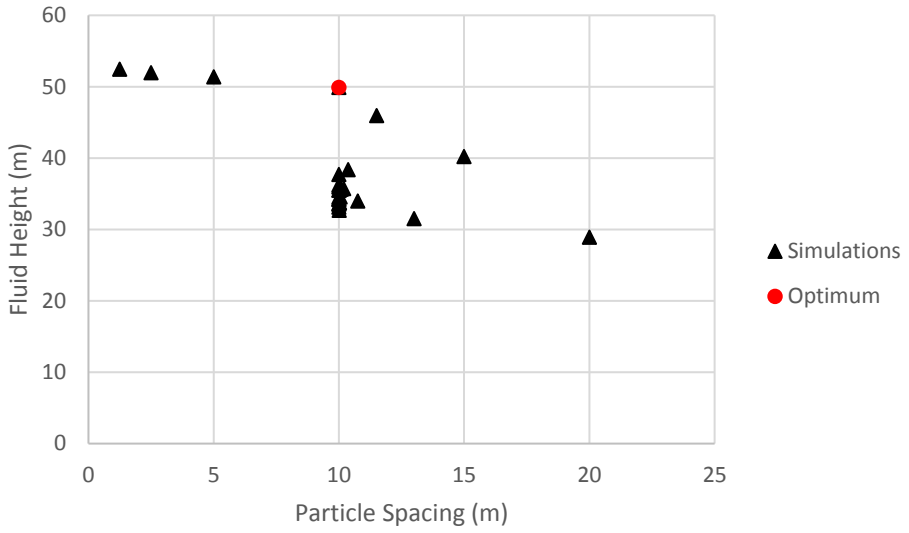


Figure 66: Large test box fluid height vs. particle spacing plot using improved height result.

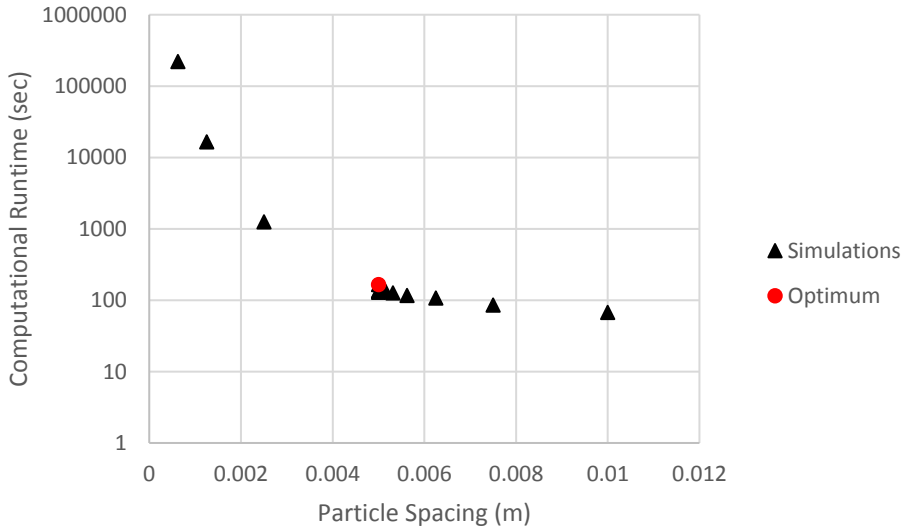


Figure 67: Small test box computational runtime vs. particle spacing plot using improved height result.

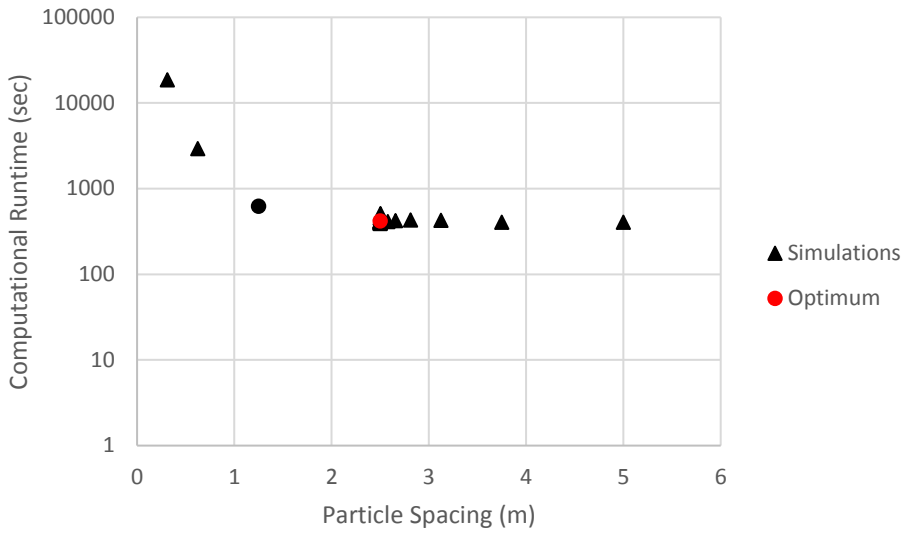


Figure 68: Medium test box computational runtime vs. particle spacing plot using improved height result.

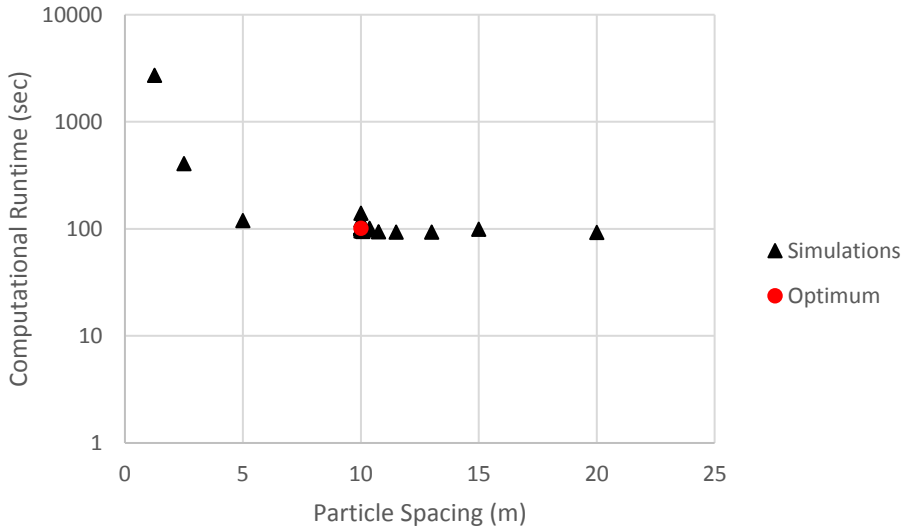


Figure 69: Large test box computational runtime vs. particle spacing plot using improved height result.

Based on the results shown in Table 21, all recommended particle spacings using the protocol are less than the optimum particle spacing found using the optimizer. This means that the particle spacing protocol should provide a converged result but is not

necessarily the optimum particle spacing. To assure that the particle spacing protocol provides converged results, each test model was ran using the protocol particle spacing. Table 22 shows the protocol particle spacing result and the convergence result compared to smallest particle spacing from the optimizer to make sure that it is within the convergence criterion and Figure 70 through Figure 72 shows the plots of fluid height vs. particle spacing for each test model with the optimum particle spacing shown in red and the protocol particle spacing show in blue.

Table 22: Particle Spacing Protocol Convergence Results for Test Models

Model	**Smallest Optimizer Particle Spacing Fluid Height (m)**	**Protocol Particle Spacing Fluid Height (m)**	**Convergence Criterion**
Small Test	0.059	0.058	0.018
Medium Test	17.62	12.84	0.27
Large Test	52.47	48.42	0.077

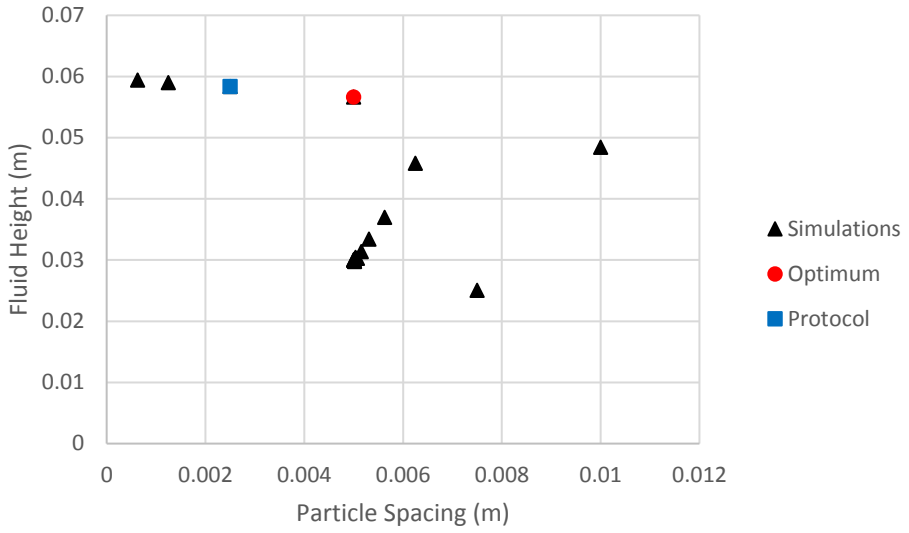


Figure 70: Small test box fluid height vs. particle spacing comparison with particle spacing protocol.

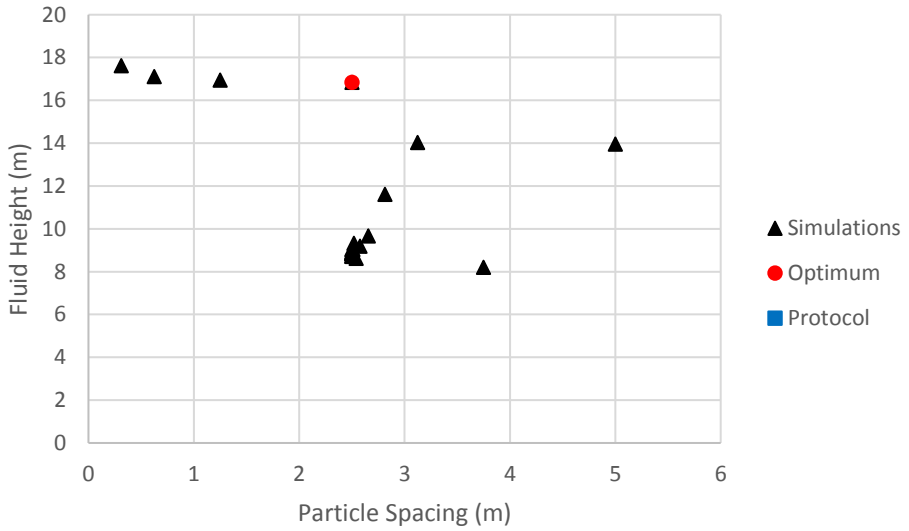


Figure 71: Medium test box fluid height vs. particle spacing comparison with particle spacing protocol.

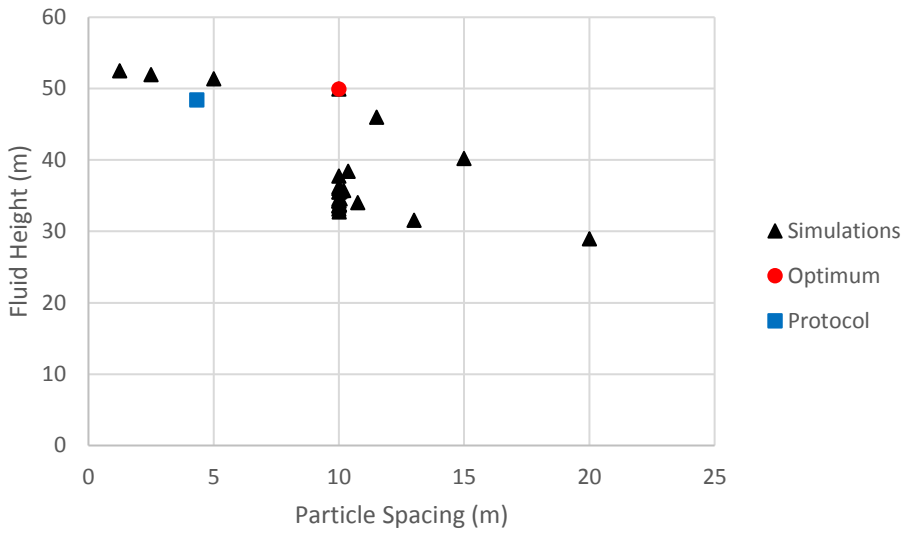


Figure 72: Large test box fluid height vs. particle spacing comparison with particle spacing protocol.

As seen in Table 22 and Figure 71 and Figure 72, the protocol particle spacing provides a result that is not within the convergence criterion for the medium and large test models. The reason for this being due to the rounding associated with the particle emitter in Neutrino. As detailed in the section 7.1, the particle emitter in Neutrino rounds the number of particles being emitted down. Therefore, to account for this rounding, the particle emitter size was adjusted to the protocol particle spacing to prevent any rounding from occurring. Table 23 shows the protocol particle spacing result and the convergence result compared to smallest particle spacing from the optimizer to make sure it is within the convergence criterion for the adjusted particle emitter size and Figure 73 and Figure 74 show the plots of fluid height vs. particle spacing for the medium and large test models

with the optimum particle spacing shown in red and the protocol particle spacing show in blue for the adjusted particle emitter size.

Table 23: Particle Spacing Protocol Convergence Results for Test Models with Adjusted Particle Emitter Size

Model	Smallest Optimizer Particle Spacing Fluid Height (m)	Protocol Particle Spacing Fluid Height (m)	Convergence Criterion
Medium Test	17.62	16.84	0.044
Large Test	52.47	51.32	0.022

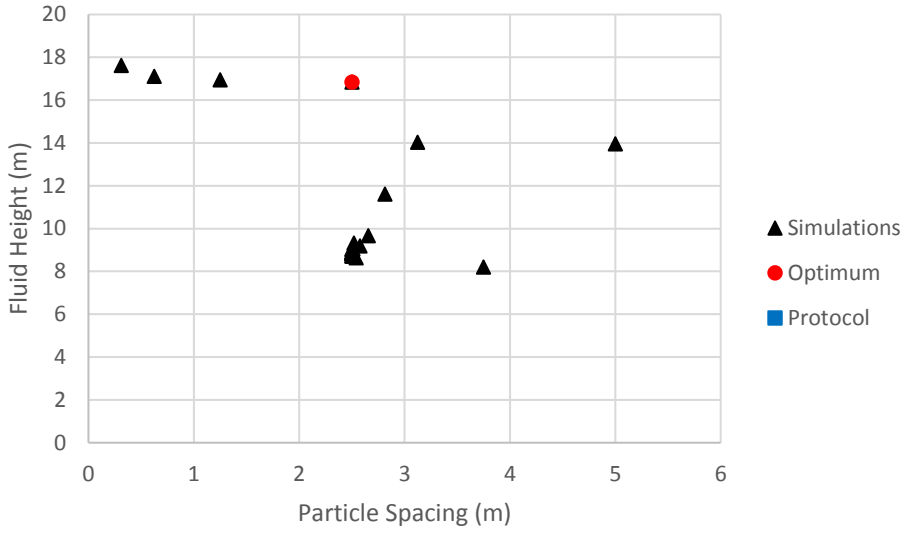


Figure 73: Medium test box fluid height vs. particle spacing comparison with particle spacing protocol with adjusted particle emitter size.

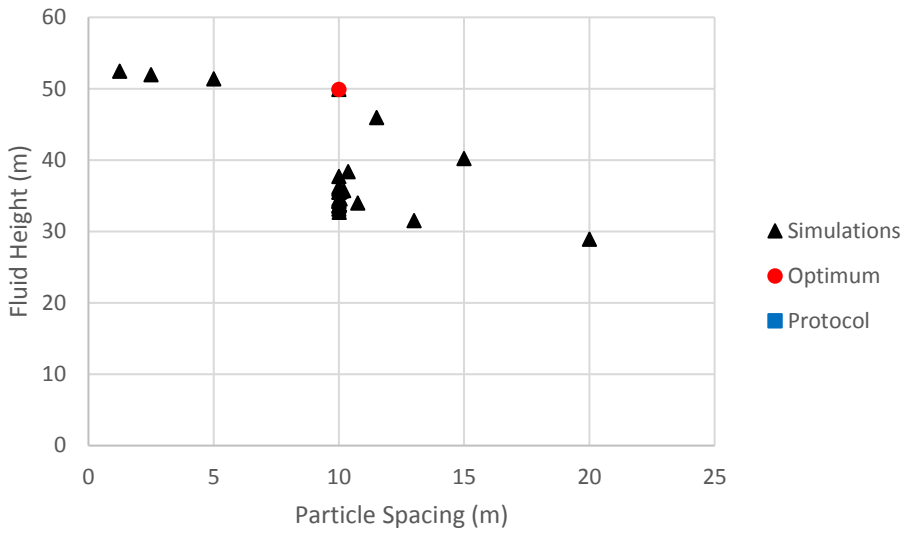


Figure 74: Large test box fluid height vs. particle spacing comparison with particle spacing protocol with adjusted particle emitter size.

Based on the results in Table 23, the medium and large test models with the particle spacing protocol provide converged results as long as the particle emitter is of a size that prevents rounding from occurring. Therefore, when using the particle spacing protocol, care should be taken in selecting the correct particle emitter size if the particle emitter is being used to produce a specific fluid height.

13 Conclusion

Two significant SPH advancements were developed through careful investigation and research. One advancement addresses a major shortcoming in the general application of SPH fluid flow: particle spacing selection. The other advancement addresses a fluid height bias specific to the SPH code Neutrino. However, the developed height methodology could be applied to other SPH codes.

The improved fluid height analysis method provides height results that are representative of the actual particle positions with no assumption about the particle arrangement. The Ogee spillway validation showed that this method provided improvements between 15% and 43% compared to the Neutrino height results. Lastly, the improved fluid height analysis method allows for the surface profile, 2D or 3D, to be analyzed with a single measurement field which is a feature that is not available in Neutrino.

The particle spacing protocol of 12 particles per the smallest dimension suggests a particle spacing that provides a converged result. The protocol does not provide the optimum particle spacing, but it does provide a particle spacing that is more efficient than just selecting a very small particle spacing. This protocol will allow users a starting particle spacing to try rather than just choosing a particle spacing with no other guidance. However, the particle emitter size is of importance and should be a size that does not provide rounding of the number of particles per emission. Neutrino users should be aware of the rounding protocol of the flow particle emitter since it can have a drastic effect on the results of the simulation.

Continued work on the particle spacing protocol would be to reanalyze the models by adjusting the particle emitter size based on the particle spacing to determine if there is a change in the optimum particle spacing. Additionally, other parameters, i.e. flow rate, and results, i.e. pressure, should be analyzed with the convergence optimizer so that the particle spacing protocol could account for additional simulation details.

While simulation results provide valuable information, users should be skeptical of code results especially when the code is proprietary. Understanding what the code is doing and how it is doing it is important in providing confidence in the code results. This is especially important if SPH was to be used to simulate flooding events at NPPs. Since the SPH results would be used to guide mitigation efforts, the results would need to be understood extremely well considering an incorrect result could result in more damage being done. Although more work needs to be done with SPH and specifically Neutrino, improvements were made to increase the ability of SPH to model flooding scenarios.

14 References

[1] "Backgrounder on Probablistic Risk Assessment," U.S. NRC, 8 Apr 2016. [Online]. Available: http://www.nrc.gov/reading-rm/doc-collections/fact-sheets/probabilistic-risk-asses.html. [Accessed 16 Oct 2016].

[2] A. Wells, E. D. Ryan, B. Savage, A. Tahhan, S. Suresh, C. Muchmore, C. L. Smith and C. L. Pope, "Non-Watertight Door Performance Experiments and Analysis Under Flooding Scenarios," *Results In Engineering*, vol. 3, 2019.

[3] "Fukushima Accident," World Nuclear Association , July 2016. [Online]. Available: http://www.world-nuclear.org/information-library/safety-and-security/safety-of-plants/fukushima-accident.aspx. [Accessed 16 Oct 2016].

[4] "Information Notice No. 80-37: Containment Cooler Leaks and Reactor Cavity Flooding at Indian Point Unit 2," U.S. NRC, 22 May 2015. [Online]. Available: https://www.nrc.gov/reading-rm/doc-collections/gen-comm/info-notices/1980/in80037.html. [Accessed 25 Sept 2019].

[5] "PRELIMINARY NOTIFICATION OF EVENT OR UNUSUAL OCCURRENCE," 27 Jan 2014. [Online]. Available: https://www.nrc.gov/docs/ML1402/ML14027A183.pdf. [Accessed 25 Sept 2019].

[6] A. Tahhan, C. Muchmore, L. Nichols, A. Wells, G. Roberts, E. Ryan, S. Suresh, B. Bhandari and C. Pope, "DEVELOPMENT OF EXPERIMENTAL AND COMPUTATIONAL PROCEDURES FOR NUCLEAR POWER PLANT COMPONENT TESTING UNDER FLOODING CONDITIONS," in *ICONE25-67895*, Shanghai, 2017.

[7] R. A. Gingold and J. J. Monaghan, "Smooth Particle Hydrodynamics: Theory and Applications to Non-Spherical Stars," *Monthly Notices of the Royal Astronomical Society,* vol. 181, pp. 375-389, 1977.

[8] L. B. Lucy, "A numerical approach to the testing of the fission hypothesis," *Astronomical Journal*, vol. 82, pp. 1013-1024, 1977.

[9] R. Sampath, N. Montanari, N. Akinci, S. Prescott and C. Smith, "Large-scale solitary wave simulation with implicit incompressible SPH," *Journal of Ocean Engineering and Marine Energy*, vol. 2, no. 3, pp. 313-329, 2016.

[10] L. Mihich, "Euler and Lagrange descriptions," University of Pavia, [Online]. Available: http://fisica.unipv.it/Mihich/FM/7-PDF_1_sect05.pdf. [Accessed 30 Nov 2017].

[11] J. J. Monaghan, "Smoothed Particle Hydrodynamics," *Annual Review of Astronomy and Astrophysics*, vol. 30, pp. 534-574, 1992.

[12] M. Kelager, "Lagrangian Fluid Dynamics Using Smoothed Particle Hydrodynamics," 9 Jan 2006. [Online]. Available: http://image.diku.dk/projects/media/kelager.06.pdf. [Accessed 7 Sept 2015].

[13] D. Violeau, Fluid Mechanics and the SPH Method, Oxford: Oxford University Press, 2012.

[14] V. Naik, "Taylor Polynomials and Taylor Series," [Online]. Available: https://files.vipulnaik.com/math-153-sequence/taylorseries.pdf. [Accessed 20 Sept 2019].

[15] "Neutrino Documentation," GitHub, [Online]. Available: http://neutrinodocs.readthedocs.io/en/master/. [Accessed 17 Sept 2016].

[16] M. Gómez-Gesteira, B. D. Rogers, A. Crespo, R. A. Dalrymple, M. Narayanaswamy and J. M. Dominguez , "SPHysics - development of a free-surface fluid solver- Part 1: Theory and Formulations," Computers & Geosciences, DOI:10.1016/j.cageo.2012.02.029, 2012. [Online]. Available: http://www.sphysics.org. [Accessed 2 Nov 2015].

[17] D. De Padova, R. A. Dalrymple, M. Mossa and A. F. Petrillo, "SPH SIMULATIONS OF REGULAR AND IRREGULAR WAVES AND THEIR COMPARISON WITH EXPERIMENTAL DATA," 10 Nov 2009. [Online]. Available: http://arxiv.org/ftp/arxiv/papers/0911/0911.1872.pdf. [Accessed 11 Nov 2015].

[18] Y. HuiXia, L. Ran, L. PengZhi, W. Hang and F. JingJie, "Two-phase smooth particle hydrodynamics modeling of air-water interface in aerated flows," *Science China Technological Sciences,* vol. 60, no. March 2017, pp. 479-490, 2017.

[19] Q. Liang, X. Xia and J. Hou, "Efficient urban flood simulation using a GPU-accelerated SPH model," *Enivornmental Earth Sciences,* vol. 74, no. Thematic Issue, pp. 7285-7294, 2015.

[20] A. C. Crespo, J. M. Dominguez, A. Barreiro, M. Gomez-Gesteira and B. D. Rogers, "GPUs, a New Tool of Acceleration in CFD: Efficiency and Reliability on Smoothed Particle Hydrodynamics Methods," *PLoS ONE,* vol. 6, no. 6, 2011.

[21] J. H. Pu, Y. Huang, S. Shao and K. Hussain, "Three-Gorges Dam Fine Sediment Pollutant Transport: Turbulence SPH Model Simulation of Multi-Fluid Flows," *Journal of Applied Fluid Mechanics,* vol. 9, no. 1, pp. 1-10, 2016.

[22] S. Ghafarizadeh, A. M. Tahvilian, J. F. Chatelain, Zhaoheng Liu, H. Champliaud and G. Lebrun, "Numerical simulation of ball-end milling with SPH method," *International Journal of Advanced Manufacturing Technology,* vol. 88, pp. 401-408, 2017.

[23] X. Li, T. Zhang and Y. O. Zhang, "Time domain simulation of sound waves using smoothed particle hydrodynamics algorithm with artificial viscosity," *Algorithms,* vol. 8, pp. 321-335, 2015.

[24] K. Guo, P.-n. Sun, X.-y. Cao and X. Huang, "A 3-D SPH model for simulating water flooding of a damaged floating structure," *Journal of Hydrodynamics,* vol. 29, no. 5, pp. 831-844, 2017.

[25] E. D. Ryan, B. M. Savage, C. L. Smith and C. L. Pope, "Comparison of free surface flow measurements and smoothed particle hydrodynamic simulation for potential nuclear power plant flooding simulation," *Annals of Nuclear Energy*, vol. 126, pp. 389-397, 2019.

[26] M. S. Shadloo and M. Yildiz, "Numerical modeling of Kelvin-Helmholtz instability using smoothed particle hydrodynamics," *International Journal For Numerical Methods in Engineering* , vol. 87, pp. 988-1006, 2011.

[27] P. Omidvar, P. K. Stansby and B. D. Rogers, "SPH for 3D floating bodies using variable mass particle distribution," *International Journal For Numerical Methods In Fluids,* vol. 72, pp. 427-452, 2013.

[28] A. Barreiro, A. Crespo, J. Domínguez and M. Gómez-Gesteira, "Smoothed Particle Hydrodynamics for coastal engineering problems," *Computers and Structures*, vol. 120, pp. 96-106, 2013.

[29] T.-J. Chang, H.-M. Kao, K.-H. Chang and M.-H. Hsu, "Numerical simulation of shallow-water dam break flows in open channels using smoothed particle hydrodynamics," *Journal of Hydrology*, vol. 408, pp. 78-90, 2011.

[30] H.-M. Kao and T.-J. Chang, "Numerical modeling of dambreak-induced flood and inundation using smoothed particle hydrodynamics," *Journal of Hydrology,* Vols. 448-449, pp. 232-244, 2012.

[31] R. Vignjevic, J. Campbell and L. Libersky, "A treatment of zero-energy modes in the smoothed particle hydrodynamics method," *Computer Methods in Applied Mechanics and Engineering,* vol. 184, pp. 67-85, 2000.

[32] C. Altomare, A. J. Crespo, J. M. Domínguez, M. Gómez-Gesteira, T. Suzuki and T. Verwaest, "Applicability of Smoothed Particle Hydrodynamics for estimation of sea wave impact on coastal structures," *Coastal Engineering,* vol. 96, pp. 1-12, 2015.

[33] "RAVEN," INL, [Online]. Available: https://raven.inl.gov/SitePages/Overview.aspx. [Accessed 8 Sept 2017].

[34] E. D. Ryan and C. L. Pope, "Coupling of the Smoothed Particle Hydrodynamic Code Neutrino and the Risk Analysis Virtual Environment for Particle Spacing Optimization," *Submitted for Publication,* 2019.

[35] X.-Y. Zhang, M. Trame, L. Lesko and S. Schmidt, "Sobol Sensitivity Analysis: A Tool to Guide the Development and Evaulation of Systems Pharmacology Models," *CPT: Pharmacometrics & Systems Pharmacology*, vol. 4(2), pp. 69-79, 2015.

[36] N. Bilal, "Implementation of Sobol's Method of Global Sensitivity Analysis to a Compressor Simulation Model," in *22nd International Compressor Engineering Conference* , Purdue, 2014.

[37] J. Fruth, "New methods for the sensitivity analysis of black-box functions with an application to sheet metal forming," TU Dortmund University and Mines Saint-Etienne, Dortmund, 2015.

[38] A. Alfonsi, C. Rabiti, D. MAndelli, J. Cogliati, C. Wang, P. W. Talbot, D. P. Maljovec and C. Smith, "RAVEN Theory Manual and User Guide," Mar 2017. [Online]. Available: https://raven.inl.gov/Shared%20Documents/Manuals/raven_theory_manual.pdf. [Accessed 8 July 2019].

[39] P. W. Talbot, "Advanced Stochastic Collocation Methods for Polynomial Chaos in RAVEN," Dec 2016. [Online]. Available: https://pdfs.semanticscholar.org/7399/40cb19662f9d54c579928c74941c528f82f5.pdf. [Accessed 8 July 2019].

[40] A. Alfonsi and C. Rabiti, "Uncertainty Quantification and Validation Using RAVEN," 28 June 2017. [Online]. Available: https://www.ne.ncsu.edu/wp-content/uploads/2017/07/08_Andrea_RAVEN_MultiPhysics_Validation_NCSU_2017-1.pdf. [Accessed 8 July 2019].

[41] J. Burkardt, "Sparse Grid Collocation for Uncertainty Quantification," 13 Feb 2014. [Online]. Available: https://people.sc.fsu.edu/~jburkardt/presentations/sparse_2014_fsu.pdf. [Accessed 8 July 2019].

[42] D. Ayres and M. D. Eaton, "Uncertainty quantification in nuclear criticality modelling using a high dimensionl model representation," *Annals of Nuclear Energy*, vol. 80, pp. 379-402, 2015.

[43] C. E. Rasmussen and C. K. I. Williams, Gaussian Processes for Machine Learning, Cambridge: The MIT Press, 2006.

[44] "Reduced Order Models (ROMs) and RAVEN," 26 Apr 2015. [Online]. Available: http://www.diegomandelli.com/raven_workshop/RAVEN_Reduced_order_models-DiegoMandelli.pdf. [Accessed 8 July 2019].

[45] M. André, P. Bardet, R. Sampath, N. Montanari, L. Lin, S. Prescott and E. Ryan, "Validation Of Risk-Informed Safety Margin Characterization For Flooding Of Nuclear Power Plants," in *18th International Topical Meeting on Nuclear Reactor Thermal Hydraulics*, Portland, 2019.

[46] E. Ryan, S. Prescott, M. Andre, P. Bardet, N. Montanari and R. Sampath, "Validation and Determination of Significant Simulation Parameters Using The Smoothed Particle Hydrodynamics Code Neutrino," in *25th International Conference on Structural Mechanics in Reactor Technology*, Charlotte, 2019.

[47] Y. Qin, J. Wu, Q. Hu, D. N. Ghista and K. K. Wong, "Computational evaluation of smoothed particle hydrodynamics for implementing blood flow modelling through CT reconstructed arteries," *Journal of X-Ray Science and Technology*, pp. 213-232, 2017.

[48] "Introduction to Petrophysics," AAPG, 28 Sept 2014. [Online]. Available: https://aapgscui.wordpress.com/2014/09/28/aapg-short-course-introduction-to-petrophysics/. [Accessed 23 Dec 2017].

[49] "Organization of Particles (Crystal Packing)," North Toronto Collegiate Institute, [Online]. Available: http://www.ntci.on.ca/chem/sch4u/crystalpacking.pdf. [Accessed 24 June 2019].

[50] "12.2: The Arrangement of Atoms in Crystalline Solids," Chemistry LibreTexts, 16 June 2019. [Online]. Available: https://chem.libretexts.org/Bookshelves/General_Chemistry/Map%3A_Chemistry_(Averill_and_Eldredge)/12%3A_Solids/12.2%3A_The_Arrangement_of_Atoms_in_Crystalline_Solids. [Accessed 24 June 2019].

[51] E. W. Weisstein, "Cubic Close Packing," WolframMathWorld, [Online]. Available: http://mathworld.wolfram.com/CubicClosePacking.html. [Accessed 24 June 2019].

[52] "Verification, Validation and Uncertainty Quantification (VVUQ)," The American Society of Mechanical Engineers, [Online]. Available: https://www.asme.org/codes-standards/publications-information/verification-validation-uncertainty. [Accessed 24 July 2019].

[53] B. M. Savage and M. C. Johnson, "Flow over Ogee Spillway: Physical and Numerical Model Case Study," *Journal of Hydraulic Engineering,* no. August, pp. 640-649, 2001.

[54] M. C. Johnson and B. M. Savage, "Physical and Numerical Comparison of Flow over Ogee Spillway in the Presence of Tailwater," *Journal of Hydraulic Engineering,* no. December, pp. 1353-1357, 2006.

Appendix A: RAVEN Files

Base Neutrino Coupling Interface

Table 24 shows the base Neutrino coupling interface used with RAVEN.

Table 24: Base Neutrino Coupling Interface

```
"""
Created on Jul 25, 2017

@author: Emerald Ryan

"""

from __future__ import division, print_function, unicode_literals,
absolute_import
import warnings
warnings.simplefilter('default',DeprecationWarning)

import os
import xml.etree.ElementTree as ET
import shutil
import tempfile
import csv
import time
import numpy as np
import math

# import CodeInterfaceBaseClass
from CodeInterfaceBaseClass import CodeInterfaceBase

class NeutrinoSimple(CodeInterfaceBase):
  """
    Provides code to interface RAVEN to Neutrino code
    The name of this class represents the type in the RAVEN input file
    e.g.
    <Models>
      <Code name="myName" subType="NeutrinoSimple">
      ...
      </Code>
      ...
    </Models>

  """

  def generateCommand(self, inputFiles, executable,
clargs=None,fargs=None,preExec=None):
    """
      See base class.  Collects all the clargs and the executable to produce
the command-line call.
      Returns tuple of commands and base file name for run.
      Commands are a list of tuples, indicating parallel/serial and the
execution command to use.
      @ In, inputFiles, list, List of input files (length of the list depends
on the number of inputs have been added in the Step is running this code)
      @ In, executable, string, executable name with absolute path (e.g.
/home/path_to_executable/code.exe)
```

```
      @ In, clargs, dict, optional, dictionary containing the command-line
flags the user can specify in the input (e.g. under the node < Code ><
clargstype =0 input0arg =0 i0extension =0 .inp0/ >< /Code >)
      @ In, fargs, dict, optional, a dictionary containing the axuiliary input
file variables the user can specify in the input (e.g. under the node < Code ><
clargstype =0 input0arg =0 aux0extension =0 .aux0/ >< /Code >)
      @ Out, returnCommand, tuple, tuple containing the generated command.
returnCommand[0] is the command to run the code (string), returnCommand[1] is
the name of the output root
    """
    found = False

    # Find the first file in the inputFiles that is an XML, which is what we
need to work with.
    for index, inputFile in enumerate(inputFiles):
      if self._isValidInput(inputFile):
        found = True
        break
    if not found: raise Exception('No correct input file has been found. Got:
'+' '.join(inputFiles))

    #Determines the path to the input file
    path = inputFiles[0].getAbsFile()

    #Creates the output file that saves information that is outputted to the
command prompt
       #The output file name is results + Neutrino input file name.
       #Example: resultsNeutrinoInput
    outputfile = 'results'

    #Creates run command tuple (['executionType','execution command'], output
file root)
       #The path to the Neutrino executable is specified in the RAVEN input file
as the executable
    returnCommand = [('serial','cd ' + executable + ' && Neutrino.exe --nogui -
-file ' + str(path) + ' --run')], outputfile

    return returnCommand

  def _isValidInput(self, inputFile):
    """
      Check if an input file is a Neutrino input file.
      @ In, inputFile, string, the file name to be checked
      @ Out, valid, bool, 'True' if an input file has an extension of
'.nescene', otherwise 'False'.
    """
    valid = False
    if inputFile.getExt() in ('nescene'): valid = True
    return valid

  def getInputExtension(self):
    """
      Return a tuple of possible file extensions for a simulation
initialization file (i.e., dsin.txt).
      @ In, None
      @ Out, validExtensions, tuple, tuple of valid extensions
    """
    validExtensions = ('nescene')
    return validExtensions

  def createNewInput(self, currentInputFiles, oriInputFiles, samplerType,
**Kwargs):
    """
```

```
      Generate a new OpenModelica input file (XML format) from the original,
changing parameters
      as specified in Kwargs['SampledVars']
      @ In , currentInputFiles, list,  list of current input files (input files
of this iteration)
      @ In , oriInputFiles, list, list of the original input files
      @ In , samplerType, string, Sampler type (e.g. MonteCarlo, Adaptive, etc.
see manual Samplers section)
      @ In , Kwargs, dictionary, kwarded dictionary of parameters. In this
dictionary there is another dictionary called "SampledVars"
            where RAVEN stores the variables that got sampled (e.g.
Kwargs['SampledVars'] => {'var1':10,'var2':40})
      @ Out, newInputFiles, list, list of newer input files, list of the new
input files (modified and not)
    """
    # Look for the correct input file
    found = False
    for index, inputFile in enumerate(currentInputFiles):
      if self._isValidInput(inputFile):
        found = True
        break
    if not found: raise Exception('No correct input file has been found. Got:
'+' '.join(oriInputFiles))

    originalPath = currentInputFiles[index].getAbsFile()

    # Since the input file is XML we can load and edit it directly using etree
    # Load the XML into a tree:
    tree = ET.parse(originalPath, ET.XMLParser(encoding='utf-8'))
    # get the root node
    root = tree.getroot()

    # grep the variables that got sampled
    varDict = Kwargs['SampledVars']

    # Go through sampled variables
    for var in varDict:
        #Search for the SPH solver properties
             #NIISphSolver_1 name may need to be changed based on Neutrino
input file
        for element in root.findall('./properties/Scene/NIISphSolver_1/'):
          #Search for the Radius property
          if element.get('name') == 'ParticleSize':
              #Set the radius value to the sampled value
              element.set('val',str(varDict[var]))


        #Change where the measurements and the output data is stored in the
input file
        #Search for the Base properties
        for elementBase in root.findall('./properties/Base/'):
            #Search for the SceneFilePath property
            if elementBase.get('name') == 'SceneFilePath':
                #Set the SceneFilePath
                elementBase.set('val', str(originalPath))

            #Search for the SaveDir property
            if elementBase.get('name') == 'SaveDir':
                #Create and set SaveDir
                            #NeutrinoInput.nescene needs to be changed to the
Neutrino input file name
                savePath = originalPath.replace("NeutrinoInput.nescene","",1)
                elementBase.set('val',str(savePath))
```

```
            if elementBase.get('name') == 'CacheDir':
                #Create and set CacheDir
                            #NeutrinoInput.nescene needs to be changed to the
Neutrino input file name
                cachePath = originalPath.replace("NeutrinoInput.nescene","",1)
                elementBase.set('val',str(cachePath))

        #Search for the Measurement field properties
             #MeasurementField_1 name may need to be changed based on Neutrino
input file
        for elementMeas in
root.findall('./properties/Scene/MeasurementField_1/'):
            #Search for the exportPath property
            if elementMeas.get('name') == 'exportPath':
                #Create and set the exportPath
                            #NeutrinoInput.nescene needs to be changed to the
Neutrino input file name
                exportPath = originalPath.replace("NeutrinoInput.nescene","",1)
                exportPath = exportPath + "\\Measurements\\results.csv"
                elementMeas.set('val',str(exportPath))


    # Now we can re-write the input file
    tree.write(currentInputFiles[index].getAbsFile())

    timerPath = originalPath.replace("NeutrinoInput.nescene","",1) +
"\\timer.csv"
    file = open(timerPath, 'w+')
    file.write(str(time.time()))
    file.close()

    return currentInputFiles


  def finalizeCodeOutput(self, command, output, workingDir):
    """
    Called by RAVEN to modify output files (if needed) so that they are in a
proper form.
    In this case, even if this simple code dumps the results into a CSV, we are
going to read
    the .out file that is in ASCI format, just to show how to use this method
    @ In, command, string, the command used to run the ended job
    @ In, output, string, the Output name root
    @ In, workingDir, string, current working dir
    @ Out, newOutputRoot, string, present in case the root of the output file
gets changed in this method.
    """
    # create full path to the outputfile
       # NeutrinoInput needs to be the name of the Neutrino Input file
       # Name of results file name needs to be the same as in the createNewInput
function
    endTime = time.time()
    outputPath = workingDir + '\\NeutrinoInput\\Measurements\\results.csv'
    timerPath = workingDir + '\\timer.csv'

    # open original output file (the working directory is provided)
    outputFile = open(outputPath,"r+")
       lines = outputFile.readlines()

    #Change the output path so RAVEN can read the output
    newOutputPath = workingDir + "\\" + output
    newOutputPath1 = workingDir + "\\" + output + ".csv"
```

```
    #Open the new output file so the results can be written to it and put in
the form for RAVEN to read
    resultsFile = open(newOutputPath1, 'w')


       #Needed for RAVEN to read output
       #These need to match RAVEN input file output names
    resultsFile.write('time,height\n')

       #Write Neutrino results to a new file for RAVEN
    for line in lines:
       resultsFile.write(line)

    resultsFile.close()

    #outputFile.close()

    return newOutputPath
```

Neutrino/RAVEN Oscillating Tank Example Files

Table 25 shows the modified Neutrino interface for the oscillating tank example.

Table 25: Neutrino Interface for Oscillating Tank Example

```
"""
Created on Jul 25, 2017

@author: Emerald Ryan

"""

from __future__ import division, print_function, unicode_literals,
absolute_import
import warnings
warnings.simplefilter('default',DeprecationWarning)

import os
import xml.etree.ElementTree as ET
import shutil
import tempfile
import csv
import time
import numpy as np

# import CodeInterfaceBaseClass
from CodeInterfaceBaseClass import CodeInterfaceBase

class NeutrinoTank(CodeInterfaceBase):
  """
    Provides code to interface RAVEN to Neutrino code
    The name of this class represents the type in the RAVEN input file
    e.g.
```

```
    <Models>
      <Code name="myName" subType="NeutrinoSimple">
      ...
      </Code>
      ...
    </Models>

  """

  def generateCommand(self, inputFiles, executable,
clargs=None,fargs=None,preExec=None):
    """
      See base class.  Collects all the clargs and the executable to
produce the command-line call.
      Returns tuple of commands and base file name for run.
      Commands are a list of tuples, indicating parallel/serial and the
execution command to use.
      @ In, inputFiles, list, List of input files (length of the list
depends on the number of inputs have been added in the Step is running
this code)
      @ In, executable, string, executable name with absolute path
(e.g. /home/path_to_executable/code.exe)
      @ In, clargs, dict, optional, dictionary containing the command-
line flags the user can specify in the input (e.g. under the node <
Code >< clargstype =0 input0arg =0 i0extension =0 .inp0/ >< /Code >)
      @ In, fargs, dict, optional, a dictionary containing the
axuiliary input file variables the user can specify in the input (e.g.
under the node < Code >< clargstype =0 input0arg =0 aux0extension =0
.aux0/ >< /Code >)
      @ Out, returnCommand, tuple, tuple containing the generated
command. returnCommand[0] is the command to run the code (string),
returnCommand[1] is the name of the output root
    """
    found = False

    # Find the first file in the inputFiles that is an XML, which is
what we need to work with.
    for index, inputFile in enumerate(inputFiles):
      if self._isValidInput(inputFile):
        found = True
        break
    if not found: raise Exception('No correct input file has been
found. Got: '+' '.join(inputFiles))

    #Determines the path to the input file
    path = inputFiles[0].getAbsFile()
    path = path.replace("NeutrinoInput.nescene","NeutrinoScene.py", 1)

    #Creates the output file that saves information that is outputted
to the command prompt
      #The output file name is results + Neutrino input file name.
      #Example: resultsNeutrinoInput
    outputfile = 'Pressure'

    #Creates run command tuple (['executionType','execution command'],
output file root)
```

```
      #The path to the Neutrino executable is specified in the RAVEN
input file as the executable
    returnCommand = [('serial','cd ' + executable + ' && Neutrino.exe -
-nogui --source ' + str(path) + ' --run')], outputfile

    return returnCommand

  def _isValidInput(self, inputFile):
    """
      Check if an input file is a Neutrino input file.
      @ In, inputFile, string, the file name to be checked
      @ Out, valid, bool, 'True' if an input file has an extension of
'.nescene', otherwise 'False'.
    """
    valid = False
    if inputFile.getExt() in ('nescene'): valid = True
    return valid

  def getInputExtension(self):
    """
      Return a tuple of possible file extensions for a simulation
initialization file (i.e., dsin.txt).
      @ In, None
      @ Out, validExtensions, tuple, tuple of valid extensions
    """
    validExtensions = ('nescene')
    return validExtensions

  def createNewInput(self, currentInputFiles, oriInputFiles,
samplerType, **Kwargs):
    """
      Generate a new OpenModelica input file (XML format) from the
original, changing parameters
      as specified in Kwargs['SampledVars']
      @ In , currentInputFiles, list,  list of current input files
(input files of this iteration)
      @ In , oriInputFiles, list, list of the original input files
      @ In , samplerType, string, Sampler type (e.g. MonteCarlo,
Adaptive, etc. see manual Samplers section)
      @ In , Kwargs, dictionary, kwarded dictionary of parameters. In
this dictionary there is another dictionary called "SampledVars"
            where RAVEN stores the variables that got sampled (e.g.
Kwargs['SampledVars'] => {'var1':10,'var2':40})
      @ Out, newInputFiles, list, list of newer input files, list of
the new input files (modified and not)
    """
    # Look for the correct input file
    found = False
    for index, inputFile in enumerate(currentInputFiles):
      if self._isValidInput(inputFile):
        found = True
        break
    if not found: raise Exception('No correct input file has been
found. Got: '+' '.join(oriInputFiles))

    originalPath = currentInputFiles[index].getAbsFile()
```

```
    # Since the input file is XML we can load and edit it directly
using etree
    # Load the XML into a tree:
    tree = ET.parse(originalPath, ET.XMLParser(encoding='utf-8'))
    # get the root node
    root = tree.getroot()

    # grep the variables that got sampled
    varDict = Kwargs['SampledVars']

      #Boolean to check if movement scripts have been created or not
    scripts = False

      #Determine other properties that depend on the particle size
    size = varDict['ParticleSize']

      #Set the measurement field length and width to 3x the particle
size
      #Set the measurement field height to the same as the particle
size
    measLW = size * 3
    measH =  size

      #Determine the z-location for the measurement field based on the
particle size
    tankL = 5.951
    tankW = 0.2
    tankH = 1.2
    depth = 0.1524
    measZ2 = tankL / 2 - measLW / 2
    measZ3 = -measZ2

      #Determine required number of particles for 6in of fluid based on
the particle size
    partV =  size * size * size
    numberOfParticles = (depth * tankL * tankW) / (partV)

      #Determine the number of particles in block emitter 1
    block1W = 0.2
    block1H = 0.14
    block1L = 5.95
    block1V = block1W * block1H * block1L
    block1N = block1V / partV

      #Determine the depth uncertainty value and number of particles
from the particle settling
    settlingFactor = 0.4
    settlingDepth = settlingFactor * size * (varDict['Settling'] + 1) -
settlingFactor * size
    settlingParticles = (tankL * tankW * settlingDepth) / partV

      #Determine the number of additional particles needed with the
settling uncertainty accounted for
    neededParts = numberOfParticles - block1N + settlingParticles

      #Determine the required size of the second block emitter
    setLength = neededParts * partV / (size * size)
```

```
      #Checks if the block emitter 2 can be the entire length of the
tank or if it must be shorter
    if setLength <= 5.95:
        block2L = setLength
        block2H = size
        block2W = size
    else:
        block2L = 5.95
        heightCounter = 1
        block2W = neededParts * partV / (block2L * heightCounter *
size)

          #Determine how many particle heights the block emitter needs
to be as well as the required block emitter width
        while block2W >= 0.2:
            heightCounter = heightCounter + 1
            block2W = neededParts * partV / (block2L * heightCounter *
size)

        block2H = heightCounter * size

    #Determine the y-position of the second block emitter
    block2Y =  -(tankH / 2) + block1H + block2H / 2

      #Set the first block emitter properties
    for element in root.findall('./properties/Scene/BlockSource_1/'):
        if element.get('name') == 'Position':
            element.set('x', str(0))
            element.set('y',str(-0.53))
            element.set('z',str(0))
        elif element.get('name') == 'Scale':
            element.set('x', str(block1W))
            element.set('y', str(block1H))
            element.set('z', str(block1L))

      #Set the second block emitter properties
    for element in root.findall('./properties/Scene/BlockSource_2/'):
        if element.get('name') == 'Scale':
            element.set('x',str(block2W))
            element.set('y',str(block2H))
            element.set('z',str(block2L))
        if element.get('name') == 'Position':
            element.set('y',str(block2Y))


    # Go through sampled variables
    for var in varDict:
        if var != 'Frequency' or var != 'Amplitude' or var !=
'Transducer' or var != 'Settling': # or var != 'Friction
                #Search for the SPH solver properties
                #NIISphSolver_1 name may need to be changed based on
Neutrino input file
            for element in
root.findall('./properties/Scene/NIISphSolver_1/'):
                        #Search for the variable name
                if element.get('name') == var:
```

```
                            #Set the variable value to the sampled
value
                    element.set('val',str(varDict[var]))

                #Change where the measurements and the output data is
stored in the input file
                #Search for the Base properties
            for elementBase in root.findall('./properties/Base/'):
                        #Search for the SceneFilePath property
                if elementBase.get('name') == 'SceneFilePath':
                            #Set the SceneFilePath
                    elementBase.set('val', str(originalPath))

                        #Search for the SaveDir property
                if elementBase.get('name') == 'SaveDir':
                            #Create and set SaveDir
                            #NeutrinoInput.nescene needs to be
changed to the Neutrino input file name
                    savePath =
originalPath.replace("NeutrinoInput.nescene","",1)
                    elementBase.set('val',str(savePath))

                if elementBase.get('name') == 'CacheDir':
                            #Create and set CacheDir
                            #NeutrinoInput.nescene needs to be
changed to the Neutrino input file name
                    cachePath =
originalPath.replace("NeutrinoInput.nescene","",1)
                    elementBase.set('val',str(cachePath))

        if var == 'Transducer':
                #Search for the Measurement field properties
                  #MeasurementField_2/3 name may need to be changed
based on Neutrino input file
            for elementMeas in
root.findall('./properties/Scene/MeasurementField_2/'):
                        #Search for the exportPath property
                if elementMeas.get('name') == 'exportPath':
                            #Create and set the exportPath
                            #NeutrinoInput.nescene needs to be
changed to the Neutrino input file name
                    exportPath =
originalPath.replace("NeutrinoInput.nescene","",1)
                    exportPath = exportPath +
"Measurements\\RightPressure.csv"
                    elementMeas.set('val',str(exportPath))
                        #Adjust the position based on sampled height
and particle size for z-location
                elif elementMeas.get('name') == 'Position':
                    elementMeas.set('y',str(-0.6+varDict[var]))
                    elementMeas.set('z',str(measZ2))
                        #Adjust the size of the measurement field based
on the particle size
                elif elementMeas.get('name') == 'Scale':
                    elementMeas.set('x',str(measLW))
                    elementMeas.set('y',str(measH))
                    elementMeas.set('z',str(measLW))
```

```
            for elementMeas in
root.findall('./properties/Scene/MeasurementField_3/'):
                            #Search for the exportPath property
                    if elementMeas.get('name') == 'exportPath':
                                #Create and set the exportPath
                                #NeutrinoInput.nescene needs to be
changed to the Neutrino input file name
                        exportPath =
originalPath.replace("NeutrinoInput.nescene","",1)
                        exportPath = exportPath +
"Measurements\\LeftPressure.csv"
                        elementMeas.set('val',str(exportPath))
                            #Adjust the position based on the sampled
height and particle size for z-location
                    elif elementMeas.get('name') == 'Position':
                        elementMeas.set('y',str(-0.6+varDict[var]))
                        elementMeas.set('z',str(measZ3))
                            #Adjust the size of the measurement field based
on the particle size
                    elif elementMeas.get('name') == 'Scale':
                        elementMeas.set('x',str(measLW))
                        elementMeas.set('y',str(measH))
                        elementMeas.set('z',str(measLW))

        #if var == 'Friction':
            #for elementBox in
root.findall('./properties/Scene/RigidBox_1/'):
                          #Search for fricition factor property
                #if elementBox.get('name') == 'Viscosity':
                    #elementBox.set('val',str(varDict[var]))

        if scripts == False:
                    #Set the paths to the python scripts
            tankPath =
originalPath.replace("NeutrinoInput.nescene","Tank.py",1)
            meas2Path =
originalPath.replace("NeutrinoInput.nescene","Meas2.py",1)
            meas3Path =
originalPath.replace("NeutrinoInput.nescene","Meas3.py",1)

                    #Get the data needed for the python scripts
            freq = varDict['Frequency']
            amp = varDict['Amplitude']
            yPos = -0.6 + varDict['Transducer']

                    #Create python script for tank and measurement field
            createTankScript(self, tankPath, freq, amp)
            createMeasurementScript(self, meas2Path, freq, amp, measZ2,
yPos)
            createMeasurementScript(self, meas3Path, freq, amp, measZ3,
yPos)

                    #Create scene builder script for Neutrino
            inpPath =
originalPath.replace("NeutrinoInput.nescene","NeutrinoScene.py",1)
            createSceneBuilderScript(self, originalPath, inpPath,
tankPath, meas2Path, meas3Path)
```

```
            scripts = True


    # Now we can re-write the input file
    tree.write(currentInputFiles[index].getAbsFile())

    timerPath = originalPath.replace("NeutrinoInput.nescene","",1) +
"\\timer.csv"
    file = open(timerPath, 'w+')
    file.write(str(time.time()))
    file.close()

    valuesPath = originalPath.replace("NeutrinoInput.nescene","",1) +
"\\values.csv"
    fileV = open(valuesPath, 'w+')
    for var in varDict:
        fileV.write(str(var) + ' = ' + str(varDict[var]) + '\n')
    fileV.close()

      #Delete any previous data
    removePath =
originalPath.replace("NeutrinoInput.nescene","NeutrinoInput",1)
    command = 'rmdir /S /Q ' + str(removePath)
    os.system(command)


    return currentInputFiles


  def finalizeCodeOutput(self, command, output, workingDir):
    """
    Called by RAVEN to modify output files (if needed) so that they are
in a proper form.
    In this case, even if this simple code dumps the results into a
CSV, we are going to read
    the .out file that is in ASCI format, just to show how to use this
method
    @ In, command, string, the command used to run the ended job
    @ In, output, string, the Output name root
    @ In, workingDir, string, current working dir
    @ Out, newOutputRoot, string, present in case the root of the
output file gets changed in this method.
    """
      #Remove the simulation files
    removePath = workingDir + '\\NeutrinoInput\\'
    command = 'rmdir /S /Q ' + str(removePath) + 'Log'
    os.system(command)
    command = 'rmdir /S /Q ' + str(removePath) + 'RenderData'
    os.system(command)
    command = 'rmdir /S /Q ' + str(removePath) + 'SceneGeometry'
    os.system(command)
    command = 'rmdir /S /Q ' + str(removePath) + 'SimData'
    os.system(command)
    command = 'rmdir /S /Q ' + str(removePath) + 'Visualization'
    os.system(command)
```

```
    # create full path to the outputfile
      # NeutrinoInput needs to be the name of the Neutrino Input file
      # Name of results file name needs to be the same as in the
createNewInput function
    endTime = time.time()
    outputPath2 = workingDir +
'\\NeutrinoInput\\Measurements\\RightPressure.csv'
    outputPath3 = workingDir +
'\\NeutrinoInput\\Measurements\\LeftPressure.csv'
    timerPath = workingDir + '\\timer.csv'

    # open original output file (the working directory is provided)
    outputFile2 = open(outputPath2,"r+")
    outputFile3 = open(outputPath3,"r+")

    #Change the output path so RAVEN can read the output
    newOutputPath2 = workingDir + "\\Right" + output #+ ".csv"
    newOutputPath3 = workingDir + "\\Left" + output #+ ".csv"

    #Open the new output file so the results can be written to it and
put in the form for RAVEN to read
    resultsFile2 = open(newOutputPath2 + ".csv", 'w')
    resultsFile3 = open(newOutputPath3 + ".csv", 'w')
    timerFile = open(timerPath, 'r')
    lineT = timerFile.readline()
    runtime = endTime - float(lineT)
    lines2 = outputFile2.readlines()
    lines3 = outputFile3.readlines()

      #Needed for RAVEN to read output
      #These need to match RAVEN input file output names
    resultsFile2.write('time,pressure\n')
#,runtime='+str(runtime)+'\n')
    resultsFile3.write('time,pressure\n')
#,runtime='+str(runtime)+'\n')

      #Write Neutrino results to a new file for RAVEN
    for line in lines2:
       resultsFile2.write(line)
    for line in lines3:
        resultsFile3.write(line)

    resultsFile2.close()
    resultsFile3.close()

    return newOutputPath2

#Method that creates the scene builder script to run the Neutrino scene
file with dynamic expressions
def createSceneBuilderScript(self, scenePath, inputPath, tankPath,
meas2Path, meas3Path):
    scenePath = scenePath.replace('\\','/')
    inputPath = inputPath.replace('\\','/')
    tankPath = tankPath.replace('\\', '/')
    meas2Path = meas2Path.replace('\\','/')
    meas3Path = meas3Path.replace('\\','/')
    file = open(inputPath, 'w+')
```

```
    file.write('Base.open("'+scenePath+'")\n')
    file.write('with open("'+tankPath+'","r") as myfile:\n')
    file.write('    data=myfile.read()\n')

file.write('RigidBox_1.setPropertyUpdateExpression("Position",data);\n'
)
    file.write('with open("'+meas2Path+'","r") as myfile2:\n')
    file.write('    data2=myfile2.read()\n')

file.write('MeasurementField_2.setPropertyUpdateExpression("Position",d
ata2);\n')
    file.write('with open("'+meas3Path+'","r") as myfile3:\n')
    file.write('    data3=myfile3.read()\n')

file.write('MeasurementField_3.setPropertyUpdateExpression("Position",d
ata3);\n')

#Method that creates the tank oscillation script
def createTankScript(self, path, frequency, amplitude):
   file = open(path, 'w+')
   file.write('a='+str(amplitude)+'\n')
   file.write('f='+str(frequency)+'\n')
   file.write('time = Scene.ElapsedTime\n')
   file.write('if time >= 2:\n')
   file.write('  time = time - 2\n')
   file.write('  zpos = a * math.sin(2 * math.pi * f * time)\n')
   file.write('  prop = vec3f(0.0, 0.0, zpos)\n')
   file.close()

#Method that creates the measurement field oscillation script
def createMeasurementScript(self, path, frequency, amplitude, z,
yPosition):
  file = open(path, 'w+')
  file.write('a='+str(amplitude)+'\n')
  file.write('f='+str(frequency)+'\n')
  file.write('z='+str(z)+'\n')
  file.write('time = Scene.ElapsedTime\n')
  file.write('if time >= 2:\n')
  file.write('  time = time - 2\n')
  file.write('  zpos = a * math.sin(2 * math.pi * f * time) + z\n')
  file.write('  prop = vec3f(0.0, '+str(yPosition)+', zpos)\n')
  file.close()
```

Table 26 shows the RAVEN input file for the oscillating tank example.

Table 26: RAVEN Input File for Oscillating Tank Example

```
<?xml version="1.0" ?>
<Simulation verbosity="debug">
  <RunInfo>
    <!-- States where the working directory is, the sequence of steps
to take, and the number of batches to run-->
    <WorkingDir>E:\Emerald\RAVEN\Sparse_Grid_Tank\</WorkingDir>
```

```
    <Sequence>sample, train, meta, validateModel, validateROM,
output_print</Sequence>
    <batchSize>1</batchSize>
  </RunInfo>

   <!-- Gives the name of the neutrino input file and provides the name
of the file to be used in this input-->
  <Files>
    <Input name="neutrinoInput" type="">NeutrinoInput.nescene</Input>
  </Files>

  <!-- States the Neutrino interface and the executable -->
  <Models>
    <Code name="neutrinoCode" subType="NeutrinoTank">
      <executable>"E:\Neutrino_02_22_19\"</executable>
    </Code>
      <ROM name = "rom" subType = "GaussPolynomialRom">
        <Target>pressure</Target>
        <Features>ParticleSize, InterRadiusToPartSizeRatio,
StopThreshold, Frequency, Amplitude, Transducer, Settling</Features>
        <IndexSet>TotalDegree</IndexSet>
        <PolynomialOrder>1</PolynomialOrder>
      </ROM>
  </Models>

  <!-- Gives different distributions for raven to use in order to
modify variables-->
  <Distributions>
    <Uniform name="size">
      <lowerBound>0.007</lowerBound>
      <upperBound>0.02</upperBound>
    </Uniform>
      <Uniform name="ratio">
      <lowerBound>2.0</lowerBound>
      <upperBound>2.4</upperBound>
    </Uniform>
      <Uniform name="stop">
        <lowerBound>0.0001</lowerBound>
        <upperBound>0.01</upperBound>
      </Uniform>
      <Uniform name="freq">
        <lowerBound>0.1099868</lowerBound>
        <upperBound>0.110011</upperBound>
      </Uniform>
      <Uniform name="amp">
        <lowerBound>0.1012</lowerBound>
        <upperBound>0.102</upperBound>
      </Uniform>
      <Uniform name="tran">
        <lowerBound>0.1006</lowerBound>
        <upperBound>0.1026</upperBound>
      </Uniform>
      <Uniform name="set">
        <lowerBound>-1</lowerBound>
        <upperBound>1</upperBound>
      </Uniform>
  </Distributions>
```

```
<!--Gives how the distributions should be sampled -->
<Samplers>
  <MonteCarlo name="myMC">
    <samplerInit>
      <limit>5</limit>
          <initialSeed>42</initialSeed>
          <reseedEachIteration>True</reseedEachIteration>
    </samplerInit>
    <variable name="ParticleSize">
      <distribution>size</distribution>
    </variable>
      <variable name="InterRadiusToPartSizeRatio">
      <distribution>ratio</distribution>
    </variable>
      <variable name="StopThreshold">
        <distribution>stop</distribution>
      </variable>
      <variable name="Frequency">
        <distribution>freq</distribution>
      </variable>
      <variable name="Amplitude">
        <distribution>amp</distribution>
      </variable>
      <variable name="Transducer">
        <distribution>tran</distribution>
      </variable>
      <variable name="Settling">
        <distribution>set</distribution>
      </variable>
   </MonteCarlo>
     <SparseGridCollocation name = "SG">
       <variable name = "ParticleSize">
         <distribution>size</distribution>
       </variable>
       <variable name = "InterRadiusToPartSizeRatio">
         <distribution>ratio</distribution>
       </variable>
       <variable name = "StopThreshold">
         <distribution>stop</distribution>
       </variable>
       <variable name = "Frequency">
         <distribution>freq</distribution>
       </variable>
       <variable name = "Amplitude">
         <distribution>amp</distribution>
       </variable>
       <variable name = "Transducer">
         <distribution>tran</distribution>
       </variable>
       <variable name = "Settling">
         <distribution>set</distribution>
       </variable>
       <ROM class = "Models" type = "ROM">rom</ROM>
     </SparseGridCollocation>
</Samplers>
```

```
  <!-- Provides variable to store information to-->
  <DataObjects>
    <PointSet name="inputPlaceHolder">
        <Input> ParticleSize, InterRadiusToPartSizeRatio,
StopThreshold, Amplitude, Frequency, Transducer, Settling</Input>
      <Output>OutputPlaceHolder</Output>
    </PointSet>
      <PointSet name = "samples">
        <Input> ParticleSize, InterRadiusToPartSizeRatio,
StopThreshold, Amplitude, Frequency, Transducer, Settling</Input>
        <Output>pressure</Output>
      </PointSet>
      <PointSet name = "samplesModel">
        <Input> ParticleSize, InterRadiusToPartSizeRatio,
StopThreshold, Amplitude, Frequency, Transducer, Settling</Input>
        <Output>pressure</Output>
      </PointSet>
      <PointSet name = "samplesROM">
        <Input> ParticleSize, InterRadiusToPartSizeRatio,
StopThreshold, Amplitude, Frequency, Transducer, Settling</Input>
        <Output>pressure</Output>
      </PointSet>
      <PointSet name = "rom_meta"/>
  </DataObjects>

  <!--Provides output files -->
  <OutStreams>
  <Print name = "samples">
    <type>csv</type>
      <source>samples</source>
  </Print>
    <Print name = "samplesModel">
        <type>csv</type>
        <source>samplesModel</source>
      </Print>
      <Print name = "samplesROM">
        <type>csv</type>
        <source>samplesROM</source>
      </Print>
      <Print name = "rom_output">
        <type>csv</type>
        <source>rom_meta</source>
      </Print>
  </OutStreams>

  <!--Provides the steps to run RAVEN and what to do  -->
  <Steps>
    <MultiRun name = "sample">
        <Input class = "Files" type = "">neutrinoInput</Input>
        <Model class = "Models" type = "Code"> neutrinoCode</Model>
        <Sampler class = "Samplers" type =
"SparseGridCollocation">SG</Sampler>
        <Output class = "DataObjects" type =
"PointSet">samples</Output>
      </MultiRun>
      <RomTrainer name = "train">
        <Input class = "DataObjects" type = "PointSet">samples</Input>
```

```
        <Output class = "Models" type = "ROM">rom</Output>
      </RomTrainer>
      <IOStep name = "meta">
        <Input class = "Models" type = "ROM">rom</Input>
        <Output class = "DataObjects" type =
"PointSet">rom_meta</Output>
      </IOStep>
      <MultiRun name = "validateModel">
        <Input class = "Files" type = "">neutrinoInput</Input>
        <Model class = "Models" type = "Code">neutrinoCode</Model>
        <Sampler class = "Samplers" type = "MonteCarlo">myMC</Sampler>
        <Output class = "DataObjects" type =
"PointSet">samplesModel</Output>
      </MultiRun>
      <MultiRun name = "validateROM">
        <Input class = "DataObjects" type =
"PointSet">inputPlaceHolder</Input>
        <Model class = "Models" type = "ROM">rom</Model>
        <Sampler class = "Samplers" type = "MonteCarlo">myMC</Sampler>
        <Output class = "DataObjects" type =
"PointSet">samplesROM</Output>
      </MultiRun>
      <IOStep name = "output_print">
        <Input class = "DataObjects" type = "PointSet">samples</Input>
        <Input class = "DataObjects" type =
"PointSet">samplesModel</Input>
        <Input class = "DataObjects" type =
"PointSet">samplesROM</Input>
        <Input class = "DataObjects" type = "PointSet">rom_meta</Input>
        <Input class = "Models" type = "ROM">rom</Input>
        <Output class = "OutStreams" type =
"Print">samplesModel</Output>
        <Output class = "OutStreams" type = "Print">samplesROM</Output>
        <Output class = "OutStreams" type = "Print">samples</Output>
        <Output class = "OutStreams" type = "Print">rom_output</Output>
      </IOStep>
  </Steps>

</Simulation>
```

Table 27 shows the RAVEN output for the oscillating tank simulation.

Table 27: RAVEN Output File for Oscillating Tank Simulation

```
<DataObjectMetadata name="rom_meta">
  <rom type="Static">
    <ROM>
      <type>GAUSSgpcROM(pressure)</type>
    </ROM>
    <pressure>
      <mean>-184.24675</mean>
      <expectedValue>-184.24675</expectedValue>
      <variance>18705.8767374</variance>
```

```
      <samples>15</samples>
      <polyCoeffs>
        <_0_0_0_0_0_0_0_>-184.24675</_0_0_0_0_0_0_0_>
        <_0_0_0_0_0_0_1_>-17.8905</_0_0_0_0_0_0_1_>
        <_0_0_0_0_0_1_0_>-18.2515</_0_0_0_0_0_1_0_>
        <_0_0_0_0_1_0_0_>-3.2365</_0_0_0_0_1_0_0_>
        <_0_0_0_1_0_0_0_>-42.399</_0_0_0_1_0_0_0_>
        <_0_0_1_0_0_0_0_>23.056</_0_0_1_0_0_0_0_>
        <_0_1_0_0_0_0_0_>-26.10065</_0_1_0_0_0_0_0_>
        <_1_0_0_0_0_0_0_>122.6039</_1_0_0_0_0_0_0_>

<inputVariables>ParticleSize,InterRadiusToPartSizeRatio,StopThreshold,F
requency,Amplitude,Transducer,Settling</inputVariables>
      </polyCoeffs>
      <partialVariance>
        <Amplitude>10.47493225</Amplitude>
        <Frequency>1797.675201</Frequency>

<InterRadiusToPartSizeRatio>681.243930422</InterRadiusToPartSizeRatio>
        <ParticleSize>15031.7162952</ParticleSize>
        <Settling>320.06999025</Settling>
        <StopThreshold>531.579136</StopThreshold>
        <Transducer>333.11725225</Transducer>
      </partialVariance>
      <sobolIndices>
        <Amplitude>0.000559980822982</Amplitude>
        <Frequency>0.0961021622369</Frequency>

<InterRadiusToPartSizeRatio>0.036418711616</InterRadiusToPartSizeRatio>
        <ParticleSize>0.803582558906</ParticleSize>
        <Settling>0.0171106649928</Settling>
        <StopThreshold>0.0284177610846</StopThreshold>
        <Transducer>0.0178081603406</Transducer>
      </sobolIndices>
      <sobolTotalIndices>
        <Amplitude>0.000559980822982</Amplitude>
        <Frequency>0.0961021622369</Frequency>

<InterRadiusToPartSizeRatio>0.036418711616</InterRadiusToPartSizeRatio>
        <ParticleSize>0.803582558906</ParticleSize>
        <Settling>0.0171106649928</Settling>
        <StopThreshold>0.0284177610846</StopThreshold>
        <Transducer>0.0178081603406</Transducer>
      </sobolTotalIndices>
    </pressure>
  </rom>

</DataObjectMetadata>
```

Appendix B: Convergence Optimizer Files

XML Editor Class

Table 28 shows the XML editor class for the convergence optimizer.

Table 28: XML Editor Class

```
# -*- coding: utf-8 -*-
"""
Created on Wed Jun  6 12:55:27 2018

@author: Emerald Ryan
"""
import xml.etree.ElementTree as ET

class XMLEditor():
    """
    This class edits XML files.
    It has a method to change the particle spacing in a Neutrino input file.
    """

    """This method changes the particle spacing in a Neutrino input file."""
    def changeParticleSpacing(self, path, newSpacing):
        """
        Input:
            path is a string.
            newSpacing is a double.
        Output:
            None
        """

        #Need to change the particle value to spacing to smoothing length
        spacing = newSpacing * 2.0

        #Need to adjust the path format
        path = path.replace("\"","")

        # Since the input file is XML we can load and edit it directly using etree
        # Load the XML into a tree:
        tree = ET.parse(path, ET.XMLParser(encoding='utf-8'))

        # Get the root node
        root = tree.getroot()

        #Search for the SPH solver properties
               #NIISphSolver_1 name may need to be changed based on Neutrino input file
        for element in root.findall('./properties/Scene/NIISphSolver_1/'):
            #Search for the Radius property
            if element.get('name') == 'Radius':
                #Set the radius value to the sampled value
                element.set('val',str(spacing))
            #Search for the Radius property
            if element.get('name') == 'ParticleSize':
                #Set the radius value to the sampled value
                element.set('val',str(newSpacing))

        #Rewrite the input file
        tree.write(path)
```

Command Prompt Class

Table 29 shows the command prompt class for the convergence optimizer.

Table 29: Command Prompt Class

```
# -*- coding: utf-8 -*-
"""
Created on Wed Jun  6 12:04:07 2018

@author: Emerald Ryan
"""

import os

class CommandPrompt():
    """
    This class contains methods that run commands from the command prompt.
    It includes a method to run Neutrino and a method to copy the results
    file to a new location and rename it.
    """

    """This method runs Neutrino."""
    def runNeutrinoCommand(self, path):
        """
        Input:
            path is a string.
        Output:
            None
        """

        #This is the command to run Neutrino.
        #It depends on the installation path of the Neutrino.
        command = 'C: && cd \"C:\\Program Files\\Neutrino_11_05_18\\" && Neutrino.exe --
nogui --file ' \
        + str(path) + ' --run'

        #Runs the command via command prompt.
        os.system(command)

    """This method copies the results file to a different location and renames it."""
    def copyResultsAndRename(self, resultsPath, outputPath):
        """
        Input:
            resultsPath is a string.
            outputPath is a string.
            newOutputName is a string.
        Output:
            None
        """

        #This is the command to copy the results file to a new location under a new name.
        command = 'copy ' + str(resultsPath) + ' ' + str(outputPath)

        #Runs the command via command prompt.
        os.system(command)
```

Adjust Value Class

Table 30 shows the adjust value class for the convergence optimizer.

Table 30: Adjust Value Class

```
# -*- coding: utf-8 -*-
"""
Created on Thu Jun  7 12:46:22 2018

@author: Emerald Ryan
"""
largestSpacing = 0

class AdjustValue():
    """
    This class calculates the next value to try in the convergence process.
    It has a method that reduces the value by the user input and a method that
    calculates the next value to try for the optimization process.
    """

    """This method returns the next value to try based on the reduction method
       entered by the user. It will return None if the value change is smaller
       than the minimum change allowed."""
    def reduceByUserInput(self, reductionMethod, previousValue, minimumChangeAllowed):
        """
        Input:
            reductionMethod is a double.
            previousValue is a double.
            minimumChangeAllowed is a double.
        Output:
            Returns a double.
        """
        #Calculate the new value by the users reduction method input
        newValue = reductionMethod * previousValue

        #Checks if the change is less than the minimum change allowed
        if newValue < minimumChangeAllowed:
            return None
        else:
            return newValue

    """This method returns the next value to try for the optimization process.
       It uses the middle value between the largest converged value and the smallest
       non converged value. It will return None if the value change is smaller
       than the minimum change allowed."""
    def adjustForOptimization(self, largestConvergedValue, smallestNonConvergedValue,
minimumChangeAllowed):
        """
        Input:
            largestConvergedValue is a double.
            smallestNonConvergedValue is a double.
            minimumChangeAllowed is a double.
        Output:
            Returns a double.
        """
        #Calculate the change in value
        global largestSpacing

               #Checks if the initial particle spacing provided convergence
        if smallestNonConvergedValue == largestConvergedValue:
                   #Checks if a larger spacing has been tried and determines the change
to add
            if largestConvergedValue < largestSpacing:
                change = (largestSpacing - largestConvergedValue) / 2
            else:
                change = largestConvergedValue / 2
        else:
            change = (smallestNonConvergedValue - largestConvergedValue) / 2

        #Check if change is less than the minimum change allowed
        #Otherwise add to largest converged value and return new value
        if change < minimumChangeAllowed:
            return None
```

```
        else:
            newValue = largestConvergedValue + change
            if newValue > largestSpacing:
                largestSpacing = newValue
            return newValue
```

Output File Analysis Class

Table 31 shows the output file analysis class for the convergence optimizer.

Table 31: Output File Analysis Class

```
# -*- coding: utf-8 -*-
"""
Created on Thu Jun  7 13:49:32 2018

@author: Emerald Ryan
"""

import numpy as np
from numpy import genfromtxt

class OutputFileAnalysis():
    """
    This class gathers information from the output file.
    """

    """This method reads the entire CSV output file."""
    def readCSVFile(self, outputPath):
        """
        Input:
            outputPath is a string.
        Output:
            Returns an array.
        """

        #Reads and stores the CSV output file.
        try:
          entireOutputFile = genfromtxt(outputPath, delimiter = ',')
        except IOError:
            entireOutputFile = np.empty

        return entireOutputFile

    """This method returns a specific column from the output file."""
    def getColumnOfDataFromOutputFile(self, entireOutputFile, column):
        """
        Input:
            entireOutputFile is an array.
            column is an integer.
        Output:
            Returns an array.
        """

        #Obtains the specific column from the output file
        outputColumn = entireOutputFile[:,column]

        return outputColumn

    """This method returns the average result."""
    def getAverage(self, resultsToAverage, startingTimestep):
        """
        Input:
            resultsToAverage is an array.
            startingTimeStep is an integer.
```

```
        Output:
            Returns a double.
        """

        #Makes sure that the starting timestep is less than the length of
        #the results array.
        if len(resultsToAverage) - 1 < startingTimestep:
            print("Starting timestep is greater than the length of the array. Fix
input.")
            return None

        #Calculates the average
        average = np.mean(resultsToAverage[startingTimestep : len(resultsToAverage)-1])

        return average
```

Analysis Class

Table 32 shows the analysis class for the convergence optimizer.

Table 32: Analysis Class

```
# -*- coding: utf-8 -*-
"""
Created on Thu Jun  7 13:22:49 2018

@author: Emerald Ryan
"""

class Analysis():
    """
    This class ***
    """

    """This method checks if the average results have converged. It returns True
       if the results have converged and False otherwise."""
    def checkConvergedAverage(self, previousAverage, newAverage, convergenceCriteria):
        """
        Input:
            previousAverage is a double.
            newAverage is a double.
            convergenceCriteria is a double.
        Output:
            Returns a boolean.
        """

        #Calculates the difference between the new and previous averages.
        differenceValue = abs((newAverage - previousAverage) / previousAverage)
        print("Difference Value = " + str(differenceValue))

        #Checks if the difference is less than the convergence criteria.
        #If yes, than returns True. Otherwise returns False.
        if differenceValue < convergenceCriteria:
            return True
        else:
            return False
```

Main File

Table 33 shows the main file for the convergence optimizer.

Table 33: Convergence Optimizer Main File

```
# -*- coding: utf-8 -*-
"""
Created on Wed Jun  6 11:38:48 2018

@author: Emerald Ryan
"""

from CommandPrompt import *
from XMLEditor import *
from AdjustValue import *
from Analysis import *
from OutputFileAnalysis import *

import numpy as np
import time
import csv

"""This method determines the optimum particle spacing for
   a simulation. It will continuously get a new particle spacing
   value, run the simulation, analyze the results, and repeat
   until one of the exit methods is reached."""
def findOptimumParticleSpacingAverage():
    startTime = time.time()
    #Input parameters
    pathToFile = '\"E:\\Emerald\\Optimizer\\Test Code\\Small_Box.nescene"'
    resultsPath = '\"E:\\Emerald\\Optimizer\\Test
Code\\Small_Box\\Measurements\\height.csv"'
    outputPath = '\"E:\\Emerald\\Optimizer\\Test Code\\Output\\Small_Box"'
    startingPoint = 1
    minimumIterations = 3
    maximumIterations = 25
    convergenceCriteria = 0.05
    timestepToStartAnalysis = 500
    reductionMethod = 0.5
    requiredConvergedIterations = 3
    minimumChangeAllowed = 0.000001
    maxTime = 12
    runTime = 0

    #Initialize classes
    xmlEditor = XMLEditor()
    commandPrompt = CommandPrompt()
    adjustValue = AdjustValue()
    analysis = Analysis()
    outputFileAnalysis = OutputFileAnalysis()

    #Variables
    exit = False #Exit the optimizer when True
    currentSpacing = startingPoint #Current particle spacing set as the starting particle
spacing input
    iteration = 1 #Counter for the number of iterations occuring. The starting point is
the first iteration.
    maximumConvergedSpacing = 0 #The largest particle spacing that has converged.
    minimumConvergedSpacing = 0 #The smallest particle spacing that has converged.
    minimumConvergedAverage = 0 #The average for the smallest converged particle spacing.
    maximumConvergedAverage = 0 #The average for the largest converged particle spacing.
    minimumNonConvergedSpacing = 5 #The smallest particle spacing that has not converged
    minimumNonConvergedAverage = 0
    convergedIterations = 0 #Counter for the number of iterations that has occured.
    crashed = False #Boolean for when Neutrino has crashed.
    optimizing = False #Boolean for when optimizing the particle spacing has started.
    converged = True
    previous = False

    #Change particle spacing to starting value
    runTime = time.time()
    xmlEditor.changeParticleSpacing(pathToFile, startingPoint)
```

```
    #Run starting particle spacing, move results, and get output
    commandPrompt.runNeutrinoCommand(pathToFile)
    currentOutputPath = str(outputPath) + str(currentSpacing) + str(".csv")
    commandPrompt.copyResultsAndRename(resultsPath, currentOutputPath)
    currentOutputPath = outputPath.replace("\"","") + str(currentSpacing) + str(".csv")
    currentEntireOutput = outputFileAnalysis.readCSVFile(currentOutputPath)
    currentOutputResults =
outputFileAnalysis.getColumnOfDataFromOutputFile(currentEntireOutput, 1)
    previousOutputResults = [i for i in range(len(currentOutputResults))]
    minimumResults = [i for i in range(len(currentOutputResults))]
    length = len(currentOutputResults)

    for i in range(0, len(currentOutputResults)):
        minimumResults[i] = currentOutputResults[i]

    #Get average and print out
    currentAverage = outputFileAnalysis.getAverage(currentOutputResults,
timestepToStartAnalysis)
    runTime = time.time() - runTime
    minimumConvergedAverage = currentAverage
    print("The starting average is " + str(currentAverage))

    #Store current spacing and its average
    results = [[currentSpacing, currentAverage, runTime]]

    while exit == False:
        #Store previous spacing and results
        if crashed == False:
            previousSpacing = currentSpacing
            length = min(len(currentOutputResults), len(previousOutputResults))
            for i in range(0, length):
                previousOutputResults[i] = currentOutputResults[i]

        #Adjust the particle spacing based on user input, for optimization, or if
Neutrino crashed
        if crashed == True:
            #Increase spacing to try and prevent Neutrino crashing
            currentSpacing = currentSpacing * 1.5
            crashed = False
        elif iteration < minimumIterations or convergedIterations <
requiredConvergedIterations:
            #Calculate new particle spacing based on user input
            currentSpacing = adjustValue.reduceByUserInput(reductionMethod,
previousSpacing, minimumChangeAllowed)
        else:
            #Calculate new particle spacing for optimization
            currentSpacing = adjustValue.adjustForOptimization(maximumConvergedSpacing,
minimumNonConvergedSpacing, minimumChangeAllowed)
            optimizing = True
            print("Optimizing particle spacing.")

        #Check if minimum change in particle spacing was reached
        if currentSpacing == None:
            if optimizing == True:
                print("Minimum change in particle spacing was reached. Exit Optimizer.")
                print("The maximum converged particle spacing found was: " +
str(maximumConvergedSpacing))
                print("Total runtime: " + str((time.time() - startTime)))
                #Write the results list to a csv file
                with open("E:\\Emerald\\Optimizer\\Test Code\\OptimizerDetails.csv", "w")
as output:
                    writer = csv.writer(output, lineterminator='\n')
                    writer.writerows(results)
                return
            else:
                print("Minimum change in particle spacing was reached before optimizing.
Exit Optimizer.")
                print("The minimum converged particle spacing found was: " +
str(minimumConvergedSpacing))
                print("Total runtime: " + str((time.time() - startTime)))
                #Write the results list to a csv file
```

```
                with open("E:\\Emerald\\Optimizer\\Test Code\\OptimizerDetails.csv", "w")
as output:
                    writer = csv.writer(output, lineterminator='\n')
                    writer.writerows(results)
                return

        #Prints out the current spacing
        print("Current Spacing = " + str(currentSpacing))

        #Change particle spacing, run, move results, and get output
        runTime = time.time()
        xmlEditor.changeParticleSpacing(pathToFile, currentSpacing)
        commandPrompt.runNeutrinoCommand(pathToFile)
        currentOutputPath = str(outputPath) + str(currentSpacing) + str(".csv")
        commandPrompt.copyResultsAndRename(resultsPath, currentOutputPath)
        currentOutputPath = outputPath.replace("\"","") + str(currentSpacing) +
str(".csv")
        currentEntireOutput = outputFileAnalysis.readCSVFile(currentOutputPath)

        #Check if Neutrino crashed******Might need to add length check other than
0*********
        if len(currentEntireOutput) < length and
currentEntireOutput[len(currentEntireOutput)-1][0] < maxTime:
            print("Neutrino crashed with particle spacing. Need to try larger spacing.")
            crashed = True
        else:
            currentOutputResults =
outputFileAnalysis.getColumnOfDataFromOutputFile(currentEntireOutput, 1)

            #Check if results are converged
            previousAverage = currentAverage
            currentAverage = outputFileAnalysis.getAverage(currentOutputResults,
timestepToStartAnalysis)
            runTime = time.time() - runTime
            print("Average = " + str(currentAverage))
            results.append([currentSpacing, currentAverage, runTime])
            previous = converged
            converged = analysis.checkConvergedAverage(minimumConvergedAverage,
currentAverage, convergenceCriteria)

            #If results are converged, adds 1 to the number of converged iterations
            #Otherwise, resets
            if converged == True:
                convergedIterations = convergedIterations + 1

                #Updates the largest particle spacing and average that has converged
                if previous == True or optimizing == False:
                    maximumConvergedSpacing = max(previousSpacing, currentSpacing,
maximumConvergedSpacing)
                else:
                    maximumConvergedSpacing = max(currentSpacing,
maximumConvergedSpacing)

                if maximumConvergedSpacing == previousSpacing:
                    maximumConvergedAverage = previousAverage
                elif maximumConvergedSpacing == currentSpacing:
                    maximumConvergedAverage = currentAverage

                #Updates the smallest converged particle spacing and average
                if optimizing == False:
                    minimumConvergedSpacing = min(previousSpacing, currentSpacing)
                    print("Converged iteration = " + str(convergedIterations))
                    if minimumConvergedSpacing == previousSpacing:
                        minimumConvergedAverage = previousAverage
                    else:
                        minimumConvergedAverage = currentAverage

                #Makes sure that the largest converged particle spacing average is still
within
                #the converged criteria of the smallest converged particle spacing
average
```

```
                maxCheck =  analysis.checkConvergedAverage(minimumConvergedAverage,
maximumConvergedAverage, convergenceCriteria)
                if maxCheck == False and optimizing == False:
                    #Updates to next smallest particle spacing and average
                    for i in range(0,len(results)):
                        if results[i][0] == maximumConvergedSpacing:
                            maximumConvergedSpacing = results[i + 1][0]
                            maximumConvergedAverage = results[i + 1][1]
                            minimumNonConvergedSpacing = results[i][0]
                            minimumNonConvergedAverage = results[i][1]
                            break;
                    print("Updated the largest converged particle spacing since it was
not longer within convergence criteria")
                    convergedIterations = convergedIterations - 1

            else:
                #Resets the number of converged iterations if it has not been met
                if convergedIterations < requiredConvergedIterations:
                  convergedIterations = 0
                  minimumConvergedAverage = currentAverage

                if optimizing == False:
                    #Sets the smallest non converged spacing and average
                    #Chooses the largest since the smallest could converge in the next
iterations
                    #Only works for the initial decreasing particle spacing, not the
optimizing
                    minimumNonConvergedSpacing = max(previousSpacing, currentSpacing)
                    if minimumNonConvergedSpacing == previousSpacing:
                        minimumNonConvergedAverage = previousAverage
                    else:
                        minimumNonConvergedAverage = currentAverage
                else:
                    #Sets the smallest non converged spacing and average
                    #Works for the optimization of particle spacing
                    minimumNonConvergedSpacing =
min(minimumNonConvergedSpacing,currentSpacing)
                    if minimumNonConvergedSpacing == currentSpacing:
                        minimumNonConvergedAverage = currentAverage

                print("Did not converge")

            print("Largest converged particle spacing: " + str(maximumConvergedSpacing))
            print("Largest converged particle spacing average: " +
str(maximumConvergedAverage))
            print("Smallest converged particle spacing: " + str(minimumConvergedSpacing))
            print("Smallest converged particle spacing average: " +
str(minimumConvergedAverage))
            print("Smallest non-converged particle spacing: " +
str(minimumNonConvergedSpacing))
            print("Smallest non-converged particle spacing average: " +
str(minimumNonConvergedAverage))

        if iteration > maximumIterations:
            print("Maximum iterations was reached. Exit Optimizer.")
            print("The maximum converged particle spacing found was: " +
str(maximumConvergedSpacing))
            print("Total runtime: " + str((time.time() - startTime)))
            #Write the results list to a csv file
            with open("E:\\Emerald\\Optimizer\\Test Code\\OptimizerDetails.csv", "w") as
output:
                writer = csv.writer(output, lineterminator='\n')
                writer.writerows(results)
            return

        iteration = iteration + 1



findOptimumParticleSpacingAverage()
```

Appendix C: Optimum Particle Spacing Details Using Neutrino Result

Table 34 through Table 36 show the output file details from the convergence optimizer for the three models with the optimum particle spacing highlighted in yellow and Table 37 through Table 39 show the printed information from the convergence optimizer for the three models.

Table 34: Small Box Convergence Optimizer Output File Details Using Neutrino Results

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
0.2	0.09472	17.02797
0.1	0.076929	17.37599
0.05	0.10604	19.56412
0.025	0.100702	32.16084
0.0125	0.099842	928.7781
0.00625	0.099884	1819.179
0.003125	0.099763	25380.03
0.0375	0.085594	23.13632
0.03125	0.099003	57.80331
0.034375	0.100282	22.7373
0.035938	0.083608	22.04326
0.035156	0.10248	22.44028
0.035547	0.10764	22.05726
0.035352	0.105225	22.31628
0.035254	0.105114	22.24227
0.035205	0.105474	22.25727
0.035181	0.104699	23.58435
0.035193	0.105717	22.82131
0.035187	0.10381	22.58829
0.03519	0.10502	22.15627
0.035188	0.104302	22.97531

Table 35: Medium Box Convergence Optimizer Output File Details Using Neutrino Results

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
5	15.15625	27.67758
2.5	6.269532	26.2515
1.25	5.112302	27.70558
0.625	5.051727	33.33091
0.3125	5.00719	64.49069
0.15625	4.992625	411.7926
1.875	2.875672	27.87159
1.5625	4.971027	26.73453
1.71875	2.67831	25.39645
1.640625	4.59815	28.08761
1.601563	4.749886	25.75147
1.621094	4.909815	25.71047
1.630859	4.787626	25.73847
1.635742	4.890936	25.96549
1.638184	4.68481	26.81253
1.636963	4.726434	26.54652
1.636353	4.677337	24.86742
1.636047	4.554292	25.43646
1.635895	4.3779	25.76147
1.635818	5.105013	26.58352
1.635857	5.151888	26.72953
1.635876	4.591166	27.01454
1.635866	4.374937	26.62952
1.635861	4.924839	25.50346
1.635864	4.845515	25.76247
1.635865	4.547298	25.32545

Table 36: Large Box Convergence Optimizer Output File Details Using Neutrino Results

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
5	41.11386	57.6843
2.5	39.85784	70.69504
1.25	40.08262	180.6563
0.625	39.56169	4587.108
7.5	24.39359	52.94203
6.25	34.59063	53.02703
5.625	31.22552	54.50912
5.3125	25.04324	54.72413

5.15625	25.6261	53.01503
5.078125	27.22752	54.47012
5.039063	27.32612	53.39905
5.019531	24.91845	55.48917
5.009766	26.22982	56.50023
5.004883	25.91735	56.12421
5.002441	20.59944	54.42911
5.001221	28.54813	54.79413
5.00061	25.52915	53.11804
5.000305	26.61378	54.02409
5.000153	24.7795	53.31005
5.000076	24.46894	53.84008
5.000038	27.9338	54.73013
5.000019	25.42109	53.93409
5.00001	24.3937	54.72813
5.000005	24.50505	55.57218
5.000002	24.36881	53.73107
5.000001	39.20792	53.96309

Table 37: Small Box Printed Convergence Optimizer Information Using Neutrino Results

```
The starting average is 0.09472000000000001
Current Spacing = 0.1
Average = 0.07692929292929294
Difference Value = 0.18782418782418778
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 0.07692929292929294
Smallest non-converged particle spacing: 0.2
Smallest non-converged particle spacing average: 0.09472000000000001
Current Spacing = 0.05
Average = 0.10604040404040407
Difference Value = 0.37841386554621864
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 0.10604040404040407
Smallest non-converged particle spacing: 0.1
Smallest non-converged particle spacing average: 0.07692929292929294
Current Spacing = 0.025
Average = 0.10070202020202022
Difference Value = 0.05034292246142132
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
```

```
Smallest converged particle spacing average: 0.10070202020202022
Smallest non-converged particle spacing: 0.05
Smallest non-converged particle spacing average: 0.10604040404040407
Current Spacing = 0.0125
Average = 0.09984164343434343
Difference Value = 0.008543788555093286
Converged iteration = 1
Difference Value = 0.00861741391749609
Largest converged particle spacing: 0.025
Largest converged particle spacing average: 0.10070202020202022
Smallest converged particle spacing: 0.0125
Smallest converged particle spacing average: 0.09984164343434343
Smallest non-converged particle spacing: 0.05
Smallest non-converged particle spacing average: 0.10604040404040407
Current Spacing = 0.00625
Average = 0.0998843888888889
Difference Value = 0.0004281325214120422
Converged iteration = 2
Difference Value = 0.00818577679882345
Largest converged particle spacing: 0.025
Largest converged particle spacing average: 0.10070202020202022
Smallest converged particle spacing: 0.00625
Smallest converged particle spacing average: 0.0998843888888889
Smallest non-converged particle spacing: 0.05
Smallest non-converged particle spacing average: 0.10604040404040407
Current Spacing = 0.003125
Average = 0.09976346565656564
Difference Value = 0.0012106319482794003
Converged iteration = 3
Difference Value = 0.009407798128079669
Largest converged particle spacing: 0.025
Largest converged particle spacing average: 0.10070202020202022
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.05
Smallest non-converged particle spacing average: 0.10604040404040407
Optimizing particle spacing.
Current Spacing = 0.037500000000000006
Average = 0.08559378181818185
Difference Value = 0.1420327947222958
Did not converge
Largest converged particle spacing: 0.025
Largest converged particle spacing average: 0.10070202020202022
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.037500000000000006
Smallest non-converged particle spacing average: 0.08559378181818185
Optimizing particle spacing.
Current Spacing = 0.03125
Average = 0.09900276464646464
Difference Value = 0.0076250459533924834
Difference Value = 0.0076250459533924834
Largest converged particle spacing: 0.03125
Largest converged particle spacing average: 0.09900276464646464
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.037500000000000006
```

```
Smallest non-converged particle spacing average: 0.08559378181818185
Optimizing particle spacing.
Current Spacing = 0.034375
Average = 0.10028202626262626
Difference Value = 0.0051979008813282145
Difference Value = 0.0051979008813282145
Largest converged particle spacing: 0.034375
Largest converged particle spacing average: 0.10028202626262626
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.037500000000000006
Smallest non-converged particle spacing average: 0.08559378181818185
Optimizing particle spacing.
Current Spacing = 0.035937500000000004
Average = 0.0836079313131313
Difference Value = 0.16193838332611213
Did not converge
Largest converged particle spacing: 0.034375
Largest converged particle spacing average: 0.10028202626262626
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.035937500000000004
Smallest non-converged particle spacing average: 0.0836079313131313
Optimizing particle spacing.
Current Spacing = 0.03515625
Average = 0.10247958585858584
Difference Value = 0.027225599914205076
Difference Value = 0.027225599914205076
Largest converged particle spacing: 0.03515625
Largest converged particle spacing average: 0.10247958585858584
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.035937500000000004
Smallest non-converged particle spacing average: 0.0836079313131313
Optimizing particle spacing.
Current Spacing = 0.035546875000000006
Average = 0.10763961616161616
Difference Value = 0.07894824476290803
Did not converge
Largest converged particle spacing: 0.03515625
Largest converged particle spacing average: 0.10247958585858584
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.035546875000000006
Smallest non-converged particle spacing average: 0.10763961616161616
Optimizing particle spacing.
Current Spacing = 0.0353515625
Average = 0.10522512121212123
Difference Value = 0.05474604876255259
Did not converge
Largest converged particle spacing: 0.03515625
Largest converged particle spacing average: 0.10247958585858584
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.0353515625
Smallest non-converged particle spacing average: 0.10522512121212123
Optimizing particle spacing.
```

```
Current Spacing = 0.03525390625
Average = 0.10511351515151517
Difference Value = 0.053627342030869225
Did not converge
Largest converged particle spacing: 0.03515625
Largest converged particle spacing average: 0.10247958585858584
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.03525390625
Smallest non-converged particle spacing average: 0.10511351515151517
Optimizing particle spacing.
Current Spacing = 0.035205078125
Average = 0.1054740101010101
Difference Value = 0.057240838686407725
Did not converge
Largest converged particle spacing: 0.03515625
Largest converged particle spacing average: 0.10247958585858584
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.035205078125
Smallest non-converged particle spacing average: 0.1054740101010101
Optimizing particle spacing.
Current Spacing = 0.0351806640625
Average = 0.10469874747474746
Difference Value = 0.04946983132253498
Difference Value = 0.04946983132253498
Largest converged particle spacing: 0.0351806640625
Largest converged particle spacing average: 0.10469874747474746
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.035205078125
Smallest non-converged particle spacing average: 0.1054740101010101
Optimizing particle spacing.
Current Spacing = 0.035192871093749994
Average = 0.10571650505050505
Difference Value = 0.059671537619118586
Did not converge
Largest converged particle spacing: 0.0351806640625
Largest converged particle spacing average: 0.10469874747474746
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.035192871093749994
Smallest non-converged particle spacing average: 0.10571650505050505
Optimizing particle spacing.
Current Spacing = 0.035186767578125
Average = 0.10380971717171719
Difference Value = 0.04055844981449133
Difference Value = 0.04055844981449133
Largest converged particle spacing: 0.035186767578125
Largest converged particle spacing average: 0.10380971717171719
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.035192871093749994
Smallest non-converged particle spacing average: 0.10571650505050505
Optimizing particle spacing.
Current Spacing = 0.0351898193359375
Average = 0.10502007070707069
```

```
Difference Value = 0.0526906820639214
Did not converge
Largest converged particle spacing: 0.035186767578125
Largest converged particle spacing average: 0.10380971717171719
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.0351898193359375
Smallest non-converged particle spacing average: 0.10502007070707069
Optimizing particle spacing.
Current Spacing = 0.03518829345703125
Average = 0.10430229292929293
Difference Value = 0.04549588612280302
Difference Value = 0.04549588612280302
Largest converged particle spacing: 0.03518829345703125
Largest converged particle spacing average: 0.10430229292929293
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.09976346565656564
Smallest non-converged particle spacing: 0.0351898193359375
Smallest non-converged particle spacing average: 0.10502007070707069
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 0.03518829345703125
Total runtime: 28565.2968416214
```

Table 38: Medium Box Printed Convergence Optimizer Information Using Neutrino Results

```
The starting average is 15.15625
Current Spacing = 2.5
Average = 6.2695324
Difference Value = 0.5863401303092785
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 6.2695324
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 15.15625
Current Spacing = 1.25
Average = 5.1123015999999994
Difference Value = 0.18458008128325498
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 5.1123015999999994
Smallest non-converged particle spacing: 2.5
Smallest non-converged particle spacing average: 6.2695324
Current Spacing = 0.625
Average = 5.051727400000001
Difference Value = 0.011848714089950932
Converged iteration = 1
```

```
Difference Value = 0.011990789526766365
Largest converged particle spacing: 1.25
Largest converged particle spacing average: 5.1123015999999994
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 5.051727400000001
Smallest non-converged particle spacing: 2.5
Smallest non-converged particle spacing average: 6.2695324
Current Spacing = 0.3125
Average = 5.0071902999999995
Difference Value = 0.008816212054514503
Converged iteration = 2
Difference Value = 0.020992072140737277
Largest converged particle spacing: 1.25
Largest converged particle spacing average: 5.1123015999999994
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 5.0071902999999995
Smallest non-converged particle spacing: 2.5
Smallest non-converged particle spacing average: 6.2695324
Current Spacing = 0.15625
Average = 4.992624848484849
Difference Value = 0.0029089071200570833
Converged iteration = 3
Difference Value = 0.023970707823454816
Largest converged particle spacing: 1.25
Largest converged particle spacing average: 5.1123015999999994
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 2.5
Smallest non-converged particle spacing average: 6.2695324
Optimizing particle spacing.
Current Spacing = 1.875
Average = 2.8756716000000013
Difference Value = 0.42401608627319476
Did not converge
Largest converged particle spacing: 1.25
Largest converged particle spacing average: 5.1123015999999994
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.875
Smallest non-converged particle spacing average: 2.8756716000000013
Optimizing particle spacing.
Current Spacing = 1.5625
Average = 4.971027099999999
Difference Value = 0.004325930575657893
Difference Value = 0.004325930575657893
Largest converged particle spacing: 1.5625
Largest converged particle spacing average: 4.971027099999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.875
Smallest non-converged particle spacing average: 2.8756716000000013
Optimizing particle spacing.
Current Spacing = 1.71875
Average = 2.678310100000001
Difference Value = 0.4635466951191398
Did not converge
Largest converged particle spacing: 1.5625
```

```
Largest converged particle spacing average: 4.971027099999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.71875
Smallest non-converged particle spacing average: 2.678310100000001
Optimizing particle spacing.
Current Spacing = 1.640625
Average = 4.5981499999999995
Difference Value = 0.07901151407451806
Did not converge
Largest converged particle spacing: 1.5625
Largest converged particle spacing average: 4.971027099999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.640625
Smallest non-converged particle spacing average: 4.5981499999999995
Optimizing particle spacing.
Current Spacing = 1.6015625
Average = 4.7498858
Difference Value = 0.0486195249696188
Difference Value = 0.0486195249696188
Largest converged particle spacing: 1.6015625
Largest converged particle spacing average: 4.7498858
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.640625
Smallest non-converged particle spacing average: 4.5981499999999995
Optimizing particle spacing.
Current Spacing = 1.62109375
Average = 4.909815099999999
Difference Value = 0.01658641516195239
Difference Value = 0.01658641516195239
Largest converged particle spacing: 1.62109375
Largest converged particle spacing average: 4.909815099999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.640625
Smallest non-converged particle spacing average: 4.5981499999999995
Optimizing particle spacing.
Current Spacing = 1.630859375
Average = 4.787626
Difference Value = 0.04106033493525175
Difference Value = 0.04106033493525175
Largest converged particle spacing: 1.630859375
Largest converged particle spacing average: 4.787626
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.640625
Smallest non-converged particle spacing average: 4.5981499999999995
Optimizing particle spacing.
Current Spacing = 1.6357421875
Average = 4.890935799999999
Difference Value = 0.02036785289720091
Difference Value = 0.02036785289720091
Largest converged particle spacing: 1.6357421875
Largest converged particle spacing average: 4.890935799999999
Smallest converged particle spacing: 0.15625
```

```
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.640625
Smallest non-converged particle spacing average: 4.5981499999999995
Optimizing particle spacing.
Current Spacing = 1.63818359375
Average = 4.6848104
Difference Value = 0.06165383096594244
Did not converge
Largest converged particle spacing: 1.6357421875
Largest converged particle spacing average: 4.890935799999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.63818359375
Smallest non-converged particle spacing average: 4.6848104
Optimizing particle spacing.
Current Spacing = 1.636962890625
Average = 4.726434200000001
Difference Value = 0.05331677355362099
Did not converge
Largest converged particle spacing: 1.6357421875
Largest converged particle spacing average: 4.890935799999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.636962890625
Smallest non-converged particle spacing average: 4.726434200000001
Optimizing particle spacing.
Current Spacing = 1.6363525390625
Average = 4.677336700000001
Difference Value = 0.06315077900966887
Did not converge
Largest converged particle spacing: 1.6357421875
Largest converged particle spacing average: 4.890935799999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.6363525390625
Smallest non-converged particle spacing average: 4.677336700000001
Optimizing particle spacing.
Current Spacing = 1.63604736328125
Average = 4.554291899999999
Difference Value = 0.08779609159255658
Did not converge
Largest converged particle spacing: 1.6357421875
Largest converged particle spacing average: 4.890935799999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.63604736328125
Smallest non-converged particle spacing average: 4.554291899999999
Optimizing particle spacing.
Current Spacing = 1.635894775390625
Average = 4.3779002
Difference Value = 0.12312654508207323
Did not converge
Largest converged particle spacing: 1.6357421875
Largest converged particle spacing average: 4.890935799999999
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.635894775390625
```

```
Smallest non-converged particle spacing average: 4.3779002
Optimizing particle spacing.
Current Spacing = 1.6358184814453125
Average = 5.1050128
Difference Value = 0.022510794406925726
Difference Value = 0.022510794406925726
Largest converged particle spacing: 1.6358184814453125
Largest converged particle spacing average: 5.1050128
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.635894775390625
Smallest non-converged particle spacing average: 4.3779002
Optimizing particle spacing.
Current Spacing = 1.6358566284179688
Average = 5.1518879
Difference Value = 0.03189966327301444
Difference Value = 0.03189966327301444
Largest converged particle spacing: 1.6358566284179688
Largest converged particle spacing average: 5.1518879
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.635894775390625
Smallest non-converged particle spacing average: 4.3779002
Optimizing particle spacing.
Current Spacing = 1.6358757019042969
Average = 4.591165800000001
Difference Value = 0.08041041749946065
Did not converge
Largest converged particle spacing: 1.6358566284179688
Largest converged particle spacing average: 5.1518879
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.6358757019042969
Smallest non-converged particle spacing average: 4.591165800000001
Optimizing particle spacing.
Current Spacing = 1.6358661651611328
Average = 4.3749367
Difference Value = 0.12372012062398466
Did not converge
Largest converged particle spacing: 1.6358566284179688
Largest converged particle spacing average: 5.1518879
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.6358661651611328
Smallest non-converged particle spacing average: 4.3749367
Optimizing particle spacing.
Current Spacing = 1.6358613967895508
Average = 4.924839100000001
Difference Value = 0.013577176443653595
Difference Value = 0.013577176443653595
Largest converged particle spacing: 1.6358613967895508
Largest converged particle spacing average: 4.924839100000001
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.6358661651611328
Smallest non-converged particle spacing average: 4.3749367
Optimizing particle spacing.
```

```
Current Spacing = 1.6358637809753418
Average = 4.8455153
Difference Value = 0.029465372013579894
Difference Value = 0.029465372013579894
Largest converged particle spacing: 1.6358637809753418
Largest converged particle spacing average: 4.8455153
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.6358661651611328
Smallest non-converged particle spacing average: 4.3749367
Optimizing particle spacing.
Current Spacing = 1.6358649730682373
Average = 4.5472983
Difference Value = 0.08919687779465264
Did not converge
Largest converged particle spacing: 1.6358637809753418
Largest converged particle spacing average: 4.8455153
Smallest converged particle spacing: 0.15625
Smallest converged particle spacing average: 4.992624848484849
Smallest non-converged particle spacing: 1.6358649730682373
Smallest non-converged particle spacing average: 4.5472983
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 1.6358637809753418
Total runtime: 1115.509803533554
```

Table 39: Large Box Printed Convergence Optimizer Information Using Neutrino Results

```
The starting average is 41.113861386138616
Current Spacing = 2.5
Average = 39.857842999999995
Difference Value = 0.03054975484647819
Converged iteration = 1
Difference Value = 0.03151245254638142
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 2.5
Smallest converged particle spacing average: 39.857842999999995
Smallest non-converged particle spacing: 10
Smallest non-converged particle spacing average: 0
Current Spacing = 1.25
Average = 40.082620999999996
Difference Value = 0.005639492332788822
Converged iteration = 2
Difference Value = 0.025727868098710915
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 40.082620999999996
Smallest non-converged particle spacing: 10
Smallest non-converged particle spacing average: 0
Current Spacing = 0.625
```

```
Average = 39.561685858585854
Difference Value = 0.01299653386973228
Converged iteration = 3
Difference Value = 0.039234312033644095
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 10
Smallest non-converged particle spacing average: 0
Optimizing particle spacing.
Current Spacing = 7.5
Average = 24.39359405940594
Difference Value = 0.38340357520148666
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 7.5
Smallest non-converged particle spacing average: 24.39359405940594
Optimizing particle spacing.
Current Spacing = 6.25
Average = 34.59062871287129
Difference Value = 0.1256533193121172
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 6.25
Smallest non-converged particle spacing average: 34.59062871287129
Optimizing particle spacing.
Current Spacing = 5.625
Average = 31.22552475247525
Difference Value = 0.21071298973224759
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.625
Smallest non-converged particle spacing average: 31.22552475247525
Optimizing particle spacing.
Current Spacing = 5.3125
Average = 25.043238613861394
Difference Value = 0.3669825218424963
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.3125
Smallest non-converged particle spacing average: 25.043238613861394
Optimizing particle spacing.
Current Spacing = 5.15625
Average = 25.626104950495048
```

```
Difference Value = 0.35224942025736355
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.15625
Smallest non-converged particle spacing average: 25.626104950495048
Optimizing particle spacing.
Current Spacing = 5.078125
Average = 27.227519801980208
Difference Value = 0.31177048674554475
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.078125
Smallest non-converged particle spacing average: 27.227519801980208
Optimizing particle spacing.
Current Spacing = 5.0390625
Average = 27.326115841584155
Difference Value = 0.3092782764803811
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.0390625
Smallest non-converged particle spacing average: 27.326115841584155
Optimizing particle spacing.
Current Spacing = 5.01953125
Average = 24.918446534653466
Difference Value = 0.3701368889150725
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.01953125
Smallest non-converged particle spacing average: 24.918446534653466
Optimizing particle spacing.
Current Spacing = 5.009765625
Average = 26.229815841584152
Difference Value = 0.3369894312557047
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.009765625
Smallest non-converged particle spacing average: 26.229815841584152
Optimizing particle spacing.
Current Spacing = 5.0048828125
Average = 25.91734851485148
Difference Value = 0.344887662080589
Did not converge
```

```
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.0048828125
Smallest non-converged particle spacing average: 25.91734851485148
Optimizing particle spacing.
Current Spacing = 5.00244140625
Average = 20.59944455445544
Difference Value = 0.4793082218970995
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.00244140625
Smallest non-converged particle spacing average: 20.59944455445544
Optimizing particle spacing.
Current Spacing = 5.001220703125
Average = 28.548125742574264
Difference Value = 0.27838955486830896
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.001220703125
Smallest non-converged particle spacing average: 28.548125742574264
Optimizing particle spacing.
Current Spacing = 5.0006103515625
Average = 25.529147524752478
Difference Value = 0.354700211310332
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.0006103515625
Smallest non-converged particle spacing average: 25.529147524752478
Optimizing particle spacing.
Current Spacing = 5.00030517578125
Average = 26.61378118811882
Difference Value = 0.3272839463098113
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.00030517578125
Smallest non-converged particle spacing average: 26.61378118811882
Optimizing particle spacing.
Current Spacing = 5.000152587890625
Average = 24.779504950495046
Difference Value = 0.3736489127619599
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
```

```
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.000152587890625
Smallest non-converged particle spacing average: 24.779504950495046
Optimizing particle spacing.
Current Spacing = 5.0000762939453125
Average = 24.46893564356436
Difference Value = 0.3814991673754974
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.0000762939453125
Smallest non-converged particle spacing average: 24.46893564356436
Optimizing particle spacing.
Current Spacing = 5.000038146972656
Average = 27.933801980198023
Difference Value = 0.2939178052207372
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.000038146972656
Smallest non-converged particle spacing average: 27.933801980198023
Optimizing particle spacing.
Current Spacing = 5.000019073486328
Average = 25.421091089108916
Difference Value = 0.3574315518307995
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.000019073486328
Smallest non-converged particle spacing average: 25.421091089108916
Optimizing particle spacing.
Current Spacing = 5.000009536743164
Average = 24.39370198019802
Difference Value = 0.38340084728962603
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.000009536743164
Smallest non-converged particle spacing average: 24.39370198019802
Optimizing particle spacing.
Current Spacing = 5.000004768371582
Average = 24.50505049504951
Difference Value = 0.3805862929440527
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
```

```
Smallest non-converged particle spacing: 5.000004768371582
Smallest non-converged particle spacing average: 24.50505049504951
Optimizing particle spacing.
Current Spacing = 5.000002384185791
Average = 24.36881188118812
Difference Value = 0.38402999386085335
Did not converge
Largest converged particle spacing: 5
Largest converged particle spacing average: 41.113861386138616
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.000002384185791
Smallest non-converged particle spacing average: 24.36881188118812
Optimizing particle spacing.
Current Spacing = 5.0000011920928955
Average = 39.20792079207921
Difference Value = 0.008942113027518286
Difference Value = 0.008942113027518286
Largest converged particle spacing: 5.0000011920928955
Largest converged particle spacing average: 39.20792079207921
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.561685858585854
Smallest non-converged particle spacing: 5.000002384185791
Smallest non-converged particle spacing average: 24.36881188118812
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 5.0000011920928955
Total runtime: 6090.549359560013
```

Appendix D: Sphere Packing Arrangement Analysis Script

Table 40 shows the sphere packing arrangement analysis script.

Table 40: Sphere Packing Arrangement Analysis Script

```
# -*- coding: utf-8 -*-
"""
Created on Thu May 23 10:54:55 2019

@author: Emerald Ryan
"""

import math
import matplotlib.pyplot as plt
import numpy as np

#Rigid box size and wanted fluid depth
W = 0.5
L = 0.5
D = 0.1
Vf = W * L * D

#Particle emitter size
El = 0.25
Ew = 0.25

#Number of particles per unit for the SC and FCC arrangements
Nsc =1
Nsf = 4

#Arrays to store results
allD = []
allDr = []
allSD = []

for i in range(1,200,2):
    #Calculate particle spacing
    dr = i / 1000

    #Calculate number of particles per emitter length and width
    Nel = math.floor(El / dr)
    New = math.floor(Ew / dr)

    #Calculate total number of particles per emission
    Ne = Nel * New

    #Calculate volume per emission
    Ve = Ne * dr * dr * dr

    #Calculate number of emissions
    En = int(round(Vf / Ve))

    #Calculate total number of particles
    Np = Ne * En

    #Calculate the number of unit cells for both arrangements
    Nucc = Np / Nsc
    Nucf = Np / Nsf

    #Calculate the volume of one unit cell for both arrangements
    Vucc = 8 * math.pow((dr / 2), 3)
    Vucf = 16 * math.sqrt(2) * math.pow((dr / 2), 3)

    #Calculate the fluid volume for both arrangements
    Vsc = Nucc * Vucc
```

```
    Vsf = Nucf * Vucf

    #Calculate the fluid depth for both arrangements
    Dsc = Vsc / (L * W)
    Dsf = Vsf / (L * W)
    Ds = [Dsc, Dsf]
    allD.append(Dsc)
    allDr.append(dr)

    #Plot the results
    plt.scatter(dr, Dsc, c='b', label='SC')
    plt.scatter(dr, Dsf, c='r', label='FCC')
    plt.xlabel('Particle Spacing (m)')
    plt.ylabel('Fluid Depth (m)')
    plt.pause(0.05)
    if i == 1:
        plt.legend()

plt.show()
```

Appendix E: Optimum Particle Spacing and Sphere Packing Arrangement Comparison Scripts

Table 41 shows the Python script used for the small box comparison.

Table 41: Small Box Optimum Particle Spacing and Sphere Packing Arrangement Comparison Script

```
# -*- coding: utf-8 -*-
"""
Created on Thu May 23 10:54:55 2019

@author: RYANED
"""

import math
import matplotlib.pyplot as plt
import numpy as np

########## Small Box ##############################
#Rigid box size and wanted fluid depth
W = 0.5
L = 0.5
D = 0.1
Vf = W * L * D

#Particle emitter size
El = 0.25
Ew = 0.25

#Number of particles per unit for the SC and FCC arrangements
Nsc =1
Nsf = 4

#Arrays to store results
allD = []
allDr = []
allSD = []

#Convergence optimizes sizes
size = [0.2,
0.1,
0.05,
0.025,
0.0125,
0.00625,
0.003125,
0.037500000000000006,
0.03125,
0.034375,
0.035937500000000004,
0.03515625,
```

```
0.035546875000000006,
0.0353515625,
0.03525390625,
0.035205078125,
0.0351806640625,
0.035192871093749994,
0.035186767578125,
0.0351898193359375,
0.03518829345703125]

#Convergence optimizer heights
height = [0.09472000000000001,
0.07692929292929294,
0.10604040404040407,
0.10070202020202022,
0.09984164343434343,
0.0998843888888889,
0.09976346565656564,
0.08559378181818185,
0.09900276464646464,
0.10028202626262626,
0.0836079313131313,
0.10247958585858584,
0.10763961616161616,
0.10522512121212123,
0.10511351515151517,
0.1054740101010101,
0.10469874747474746,
0.10571650505050505,
0.10380971717171719,
0.10502007070707069,
0.10430229292929293]

#Plot convergence results
plt.scatter(size, height, s=30, c='g', marker='x', label='Optimum
Particle Spacing Results')

#Optimum convergence point
optimumSize = 0.03518829345703125
optimumHeight = 0.10430229292929293

#Plot optimum convergence point
plt.scatter(optimumSize, optimumHeight, s=30, c='k', marker="s",
label='Simulation Optimum Particle Spacing')

for i in range(1,200,2):
    #Calculate particle spacing
    dr = i / 1000

    #Calculate number of particles per emitter length and width
    Nel = math.floor(El / dr)
    New = math.floor(Ew / dr)

    #Calculate total number of particles per emission
    Ne = Nel * New

    #Calculate volume per emission
```

```
    Ve = Ne * dr * dr * dr

    #Calculate number of emissions
    En = int(round(Vf / Ve))

    #Calculate total number of particles
    Np = Ne * En

    #Calculate the number of unit cells for both arrangements
    Nucc = Np / Nsc
    Nucf = Np / Nsf

    #Calculate the volume of one unit cell for both arrangements
    Vucc = 8 * math.pow((dr / 2), 3)
    Vucf = 16 * math.sqrt(2) * math.pow((dr / 2), 3)

    #Calculate the fluid volume for both arrangements
    Vsc = Nucc * Vucc
    Vsf = Nucf * Vucf

    #Calculate the fluid depth for both arrangements
    Dsc = Vsc / (L * W)
    Dsf = Vsf / (L * W)
    Ds = [Dsc, Dsf]
    allD.append(Dsc)
    allDr.append(dr)

    #Plot the results
    plt.scatter(dr, Dsc, c='b', label='SC Results')
    plt.scatter(dr, Dsf, c='r', label='FCC Results')
    plt.xlabel('Particle Spacing (m)')
    plt.ylabel('Fluid Depth (m)')
    plt.pause(0.05)
    if i == 1:
        plt.legend(loc='upper left')

plt.show()
```

Table 42 shows the Python script used for the medium box comparison.

Table 42: Medium Box Optimum Particle Spacing and Sphere Packing Arrangement Comparison Script

```
# -*- coding: utf-8 -*-
"""
Created on Thu May 23 10:54:55 2019

@author: RYANED
"""

import math
```

```
import matplotlib.pyplot as plt
import numpy as np


################### Medium Box ################################
#Rigid box size and wanted fluid depth
W = 7.5
L = 7.5
D = 5
Vf = W * L * D

#Particle emitter size
El = 5
Ew = 5

#Number of particles per unit for the SC and FCC arrangements
Nsc =1
Nsf = 4

#Arrays to store results
allD = []
allDr = []
allSD = []

#Convergence optimizes sizes
size = [5,
2.5,
1.25,
0.625,
0.3125,
0.15625,
1.875,
1.5625,
1.71875,
1.640625,
1.6015625,
1.62109375,
1.630859375,
1.635742188,
1.638183594,
1.636962891,
1.636352539,
1.636047363,
1.635894775,
1.635818481,
1.635856628,
1.635875702,
1.635866165,
1.635861397,
1.635863781,
1.635864973
]

#Convergence optimizer heights
height = [15.15625,
6.2695324,
5.1123016,
```

```
5.0517274,
5.0071903,
4.992624848,
2.8756716,
4.9710271,
2.6783101,
4.59815,
4.7498858,
4.9098151,
4.787626,
4.8909358,
4.6848104,
4.7264342,
4.6773367,
4.5542919,
4.3779002,
5.1050128,
5.1518879,
4.5911658,
4.3749367,
4.9248391,
4.8455153,
4.5472983
]

#Plot convergence results
plt.scatter(size, height, s=30, c='g', marker='x', label='Optimum
Particle Spacing Results')

#Optimum convergence point
optimumSize = 1.6358637809753418
optimumHeight = 4.8455153

#Plot optimum convergence point
plt.scatter(optimumSize, optimumHeight, s=30, c='k', marker="s",
label='Simulation Optimum Particle Spacing')

for i in range(5,5000,5):
    #Calculate particle spacing
    dr = i / 1000


    #Calculate number of particles per emitter length and width
    Nel = math.floor(El / dr)
    New = math.floor(Ew / dr)

    #Calculate total number of particles per emission
    Ne = Nel * New

    #Calculate volume per emission
    Ve = Ne * dr * dr * dr

    #Calculate number of emissions
    En = int(round(Vf / Ve))

    #Calculate total number of particles
    Np = Ne * En
```

```
    #Calculate the number of unit cells for both arrangements
    Nucc = Np / Nsc
    Nucf = Np / Nsf

    #Calculate the volume of one unit cell for both arrangements
    Vucc = 8 * math.pow((dr / 2), 3)
    Vucf = 16 * math.sqrt(2) * math.pow((dr / 2), 3)

    #Calculate the fluid volume for both arrangements
    Vsc = Nucc * Vucc
    Vsf = Nucf * Vucf

    #Calculate the fluid depth for both arrangements
    Dsc = Vsc / (L * W)
    Dsf = Vsf / (L * W)
    Ds = [Dsc, Dsf]
    allD.append(Dsc)
    allDr.append(dr)

    #Plot the results
    plt.scatter(dr, Dsc, c='b', label='SC Results')
    plt.scatter(dr, Dsf, c='r', label='FCC Results')
    plt.xlabel('Particle Spacing (m)')
    plt.ylabel('Fluid Depth (m)')
    plt.pause(0.05)
    if i == 1:
        plt.legend(loc='upper left')

plt.show()
```

Table 43 shows the Python script used for the large box comparison.

Table 43: Large Box Optimum Particle Spacing and Sphere Packing Arrangement Comparison Script

```
# -*- coding: utf-8 -*-
"""
Created on Thu May 23 10:54:55 2019

@author: RYANED
"""

import math
import matplotlib.pyplot as plt
import numpy as np


################### Large Box ###############################
#Rigid box size and wanted fluid depth
W = 30
```

```
L = 30
D = 40
Vf = W * L * D

#Particle emitter size
El = 20
Ew = 20

#Number of particles per unit for the SC and FCC arrangements
Nsc =1
Nsf = 4

#Arrays to store results
allD = []
allDr = []
allSD = []

#Convergence optimizes sizes
size = [5,
2.5,
1.25,
0.625,
7.5,
6.25,
5.625,
5.3125,
5.15625,
5.078125,
5.0390625,
5.01953125,
5.009765625,
5.004882813,
5.002441406,
5.001220703,
5.000610352,
5.000305176,
5.000152588,
5.000076294,
5.000038147,
5.000019073,
5.000009537,
5.000004768,
5.000002384,
5.000001192
]

#Convergence optimizer heights
height = [41.11386139,
39.857843,
40.082621,
39.56168586,
24.39359406,
34.59062871,
31.22552475,
25.04323861,
25.62610495,
27.2275198,
```

```
27.32611584,
24.91844653,
26.22981584,
25.91734851,
20.59944455,
28.54812574,
25.52914752,
26.61378119,
24.77950495,
24.46893564,
27.93380198,
25.42109109,
24.39370198,
24.5050505,
24.36881188,
39.20792079
]

#Plot convergence results
plt.scatter(size, height, s=30, c='g', marker='x', label='Optimum
Particle Spacing Results')

#Optimum convergence point
optimumSize = 5.0000011920928955
optimumHeight = 39.20792079207921

#Plot optimum convergence point
plt.scatter(optimumSize, optimumHeight, s=30, c='k', marker="s",
label='Simulation Optimum Particle Spacing')

for i in range(100,7000,5):
    #Calculate particle spacing
    dr = i / 1000


    #Calculate number of particles per emitter length and width
    Nel = math.floor(El / dr)
    New = math.floor(Ew / dr)

    #Calculate total number of particles per emission
    Ne = Nel * New

    #Calculate volume per emission
    Ve = Ne * dr * dr * dr

    #Calculate number of emissions
    En = int(round(Vf / Ve))

    #Calculate total number of particles
    Np = Ne * En

    #Calculate the number of unit cells for both arrangements
    Nucc = Np / Nsc
    Nucf = Np / Nsf

    #Calculate the volume of one unit cell for both arrangements
    Vucc = 8 * math.pow((dr / 2), 3)
```

```
    Vucf = 16 * math.sqrt(2) * math.pow((dr / 2), 3)

    #Calculate the fluid volume for both arrangements
    Vsc = Nucc * Vucc
    Vsf = Nucf * Vucf

    #Calculate the fluid depth for both arrangements
    Dsc = Vsc / (L * W)
    Dsf = Vsf / (L * W)
    Ds = [Dsc, Dsf]
    allD.append(Dsc)
    allDr.append(dr)

    #Plot the results
    plt.scatter(dr, Dsc, c='b', label='SC Results')
    plt.scatter(dr, Dsf, c='r', label='FCC Results')
    plt.xlabel('Particle Spacing (m)')
    plt.ylabel('Fluid Depth (m)')
    plt.pause(0.05)
    if i == 1:
        plt.legend(loc='upper left')

plt.show()
```

Appendix F: Improved Fluid Height Analysis Method Files

Per Particle Output File Example

Table 44 shows an example of a per particle output file for the small box model for the last time step.

Table 44: Per Particle Output Example

ID	x	y	z	u	v	w	Dens.	Press.	Temp.
216	-0.108	-0.173	-0.070	0.0556	0.0099	0.0040	997.266	0	293
516	-0.122	-0.234	-0.044	-0.2306	0.0207	-0.0436	998.004	1241.61	293
69	-0.103	-0.203	-0.059	0.0520	-0.0869	-0.1233	998.005	897.592	293
36	0.025	-0.172	-0.006	-0.1371	0.0813	0.2412	997.995	0	293
545	0.117	-0.230	-0.105	-0.0277	0.1116	-0.0482	998.004	1165.5	293
478	-0.017	-0.178	0.023	-0.0207	-0.0014	0.0288	998.004	585.615	293
12	0.057	-0.168	0.027	-0.0899	0.0013	-0.0876	997.284	0	293
397	0.057	-0.233	0.011	0.1313	-0.0023	0.0304	998.006	1393.94	293
54	0.090	-0.207	0.022	-0.1533	0.0607	0.0019	997.992	0	293
326	0.123	-0.236	-0.053	0.0756	0.0727	-0.0114	998.001	618.355	293
124	0.117	-0.214	-0.093	-0.1045	-0.0123	-0.0370	997.997	62.8213	293
114	0.123	-0.202	-0.040	-0.0273	0.0853	-0.0312	998.004	790.531	293
524	0.105	-0.234	0.044	-0.0879	0.0068	-0.0346	998.007	1596.64	293
580	0.122	-0.240	-0.006	-0.0828	-0.0729	0.0378	998.002	827.012	293
404	-0.108	-0.200	-0.104	-0.0489	0.0195	-0.1187	997.924	0	293
543	-0.100	-0.233	-0.091	-0.0835	0.0650	0.0730	998.006	1152.61	293
575	-0.073	-0.214	-0.094	-0.0332	-0.0491	0.1331	997.996	1216.86	293
304	0.124	-0.172	0.026	-0.0969	-0.0748	-0.0516	998.003	594.657	293
289	0.094	-0.174	-0.017	0.0047	-0.0408	-0.0403	998.003	414.306	293
20	0.090	-0.229	0.012	0.1565	-0.0651	0.1679	998	398.44	293
403	-0.057	-0.210	-0.045	-0.0541	0.0720	-0.0419	998.003	583.307	293
480	0.105	-0.175	-0.021	0.0260	-0.0067	0.0212	997.981	0	293
370	-0.090	-0.234	-0.040	0.0125	0.1332	0.1040	998.002	621.245	293
259	-0.030	-0.235	-0.074	0.1643	-0.0206	-0.0517	998.006	1356.08	293
214	-0.053	-0.202	-0.077	-0.0423	-0.0311	0.0048	998.007	1014.01	293
104	-0.087	-0.175	-0.049	-0.0796	0.0874	0.0094	998	80.2395	293
509	-0.066	-0.174	-0.044	-0.0406	0.1718	0.1039	998	171.075	293
246	-0.046	-0.236	-0.033	0.0102	-0.0231	-0.0033	998.003	1079.44	293
577	-0.069	-0.236	-0.073	-0.0881	0.1607	0.1381	998.003	727.408	293
479	0.081	-0.172	0.016	-0.0246	0.0159	-0.0414	998.001	281.659	293
120	0.116	-0.169	0.092	-0.0683	0.0091	-0.0253	962.102	0	293

225	0.120	-0.236	0.079	0.1423	0.0357	0.0570	998.001	274.089	293
52	0.057	-0.239	-0.115	-0.1203	0.0613	-0.1952	998.004	1659.47	293
197	0.122	-0.166	0.112	0.0387	0.0116	-0.0292	974.588	0	293
338	0.005	-0.211	-0.115	0.0200	0.1300	-0.0330	998.008	1354.73	293
331	0.090	-0.169	0.035	-0.0443	0.0392	-0.0874	997.099	0	293
170	-0.016	-0.170	-0.108	0.0253	-0.0896	0.0844	963.733	0	293
542	0.052	-0.194	0.000	-0.1348	-0.0851	0.0342	998.006	899.567	293
146	-0.026	-0.242	0.079	-0.0402	0.0593	-0.0333	998.004	1565.73	293
203	-0.091	-0.233	0.124	0.1395	-0.0084	-0.1037	997.999	245.276	293
365	0.089	-0.238	0.087	0.0527	0.0468	-0.0833	998	917.828	293
531	0.001	-0.170	0.090	0.0869	-0.0058	0.1547	998.005	492.686	293
198	0.016	-0.228	0.059	-0.0631	-0.0357	0.0472	998.004	976.146	293
169	0.021	-0.232	0.086	0.0621	0.0936	0.0336	998.003	974.392	293
318	-0.022	-0.171	0.067	0.0259	0.0331	0.0421	997.804	0	293
16	0.010	-0.198	0.069	-0.0483	-0.0837	0.0021	998.003	526.506	293
215	-0.019	-0.217	0.101	-0.0579	0.0922	-0.0061	997.966	0	293
7	0.016	-0.233	0.121	-0.0413	0.0042	-0.0458	998.001	624.521	293
329	0.015	-0.202	0.120	-0.0721	-0.0081	0.0864	997.999	247.275	293
229	0.061	-0.178	-0.109	0.0018	-0.0338	-0.0526	997.998	0	293
284	0.087	-0.177	-0.102	-0.0632	-0.0061	-0.0492	997.998	250.343	293
323	0.004	-0.167	0.118	-0.1209	0.0245	0.0456	968.822	0	293
438	0.001	-0.169	0.046	-0.1431	-0.0624	-0.0217	997.996	54.4953	293
89	0.028	-0.167	0.070	0.1481	-0.0003	0.0478	997.789	0	293
428	0.052	-0.188	0.054	-0.0749	0.1126	-0.1059	998.005	704.107	293
472	0.105	-0.181	0.060	-0.0494	-0.0121	0.0122	998.001	411.381	293
4	0.086	-0.211	-0.121	-0.0701	0.0516	0.0494	998.002	564.61	293
454	0.063	-0.221	0.069	0.0900	-0.1343	-0.0663	998.003	905.485	293
144	0.081	-0.189	0.089	-0.0964	-0.0162	0.0768	997.293	0	293
81	0.045	-0.197	0.106	-0.0494	-0.1672	-0.0624	998.009	1126.59	293
476	0.098	-0.208	0.055	-0.0284	0.1167	0.0910	998.005	910.429	293
385	0.095	-0.205	0.107	-0.1981	-0.0334	-0.0898	998.006	944.772	293
576	0.076	-0.173	0.124	-0.0666	-0.0622	0.0116	996.047	0	293
441	0.058	-0.218	-0.099	0.0086	0.0538	-0.0330	998.004	823.158	293
101	0.055	-0.235	0.048	-0.0928	0.0241	-0.0683	997.908	0	293
264	0.113	-0.171	-0.110	0.0043	-0.0305	0.0288	998	138.307	293
154	0.051	-0.230	0.096	0.0724	0.0280	-0.0345	998.002	435.562	293
448	0.012	-0.237	-0.092	0.0200	-0.0847	-0.1670	998.001	710.954	293
317	-0.010	-0.236	-0.044	-0.0952	0.0563	0.0149	998	706.948	293
62	-0.077	-0.158	-0.113	0.0741	0.0636	0.0832	988.334	0	293
430	-0.059	-0.164	-0.119	0.0258	0.0352	0.0252	997.135	0	293
520	-0.043	-0.166	-0.071	-0.1951	0.0615	-0.0401	988.158	0	293
469	-0.049	-0.165	-0.080	-0.0294	0.1155	0.0948	979.688	0	293
382	-0.011	-0.164	-0.036	-0.0316	0.0519	-0.1012	998	90.6107	293

354	-0.073	-0.166	0.025	0.0882	0.0585	-0.1285	998	0	293
296	0.035	-0.235	-0.052	0.0311	0.1126	0.0265	998.005	1492.39	293
498	0.012	-0.176	-0.040	0.0784	0.0197	-0.1125	998.005	467.709	293
162	-0.056	-0.164	0.009	-0.0682	0.0084	-0.0264	997.996	199.401	293
437	-0.032	-0.166	0.027	-0.0277	0.0374	0.0142	960.007	0	293
451	-0.120	-0.160	0.083	0.0476	-0.0352	-0.0618	998.039	2002.01	293
50	-0.069	-0.163	0.113	0.1310	0.0145	0.0911	997.913	0	293
378	0.040	-0.164	0.094	-0.1518	-0.0062	-0.0636	997.99	0	293
100	0.082	-0.158	0.072	0.0417	0.0921	-0.1018	707.05	0	293
334	0.028	-0.171	-0.096	-0.0482	0.0705	-0.0632	997.985	0	293
505	-0.009	-0.198	-0.049	0.0472	0.1333	0.0130	997.99	0	293
464	-0.004	-0.221	-0.079	-0.0200	-0.1077	0.0750	997.999	398.276	293
57	0.029	-0.209	-0.050	0.0327	-0.0701	-0.2098	998	376.481	293
526	0.034	-0.201	-0.083	0.0497	-0.0948	-0.0504	998.006	911.046	293
450	-0.010	-0.170	-0.070	0.0077	0.0884	0.0647	998.005	633.082	293
517	0.007	-0.170	-0.088	0.0695	0.0284	0.1199	998.002	202.699	293
79	0.102	-0.176	-0.064	-0.0760	-0.0214	-0.0938	997.999	0	293
324	0.081	-0.205	-0.075	0.0960	-0.0061	-0.0606	998.008	1186.55	293
139	0.052	-0.164	0.106	-0.0534	-0.1211	-0.0062	982.027	0	293
156	0.084	-0.235	-0.028	-0.0519	0.0921	-0.0508	998.005	1415.75	293
6	0.096	-0.235	-0.073	-0.0102	-0.1186	0.0510	998.001	791.449	293
233	0.066	-0.171	-0.037	-0.0956	-0.0592	0.0079	997.999	0	293
196	0.050	-0.173	-0.073	-0.1444	0.0328	0.0939	997.999	30.1831	293
119	0.049	-0.173	-0.040	-0.0429	0.0090	-0.1573	998.005	480.925	293
235	0.080	-0.170	-0.070	-0.0672	-0.0751	0.0973	994.074	0	293
486	0.063	-0.236	-0.065	-0.0616	-0.0839	-0.1134	997.998	311.563	293
381	0.060	-0.207	-0.034	0.0418	-0.0750	0.1897	998.001	446.821	293
182	0.092	-0.213	-0.026	-0.1109	-0.0285	-0.1991	998.003	776.092	293
442	-0.122	-0.238	0.028	-0.0668	-0.0067	-0.0310	998.003	1110.1	293
58	-0.051	-0.239	0.002	-0.0844	0.0481	-0.0479	998.003	1249.01	293
254	-0.075	-0.239	0.043	-0.0056	-0.0191	-0.0764	998	614.435	293
349	-0.098	-0.176	0.008	0.0313	0.0140	0.0282	997.625	0	293
125	-0.119	-0.197	0.026	0.0133	-0.1091	0.0363	998.01	1490.17	293
556	-0.069	-0.174	0.043	0.0591	-0.1425	0.0087	998.004	647.066	293
72	-0.030	-0.234	0.040	0.0064	-0.0017	0.0419	998.004	1257.41	293
201	-0.034	-0.203	-0.022	-0.0321	-0.0035	-0.0039	998.001	523.089	293
319	-0.046	-0.170	-0.020	-0.0218	0.0020	-0.1531	998	45.4491	293
501	-0.091	-0.213	-0.020	-0.0940	-0.0569	-0.0813	998.003	704.816	293
537	-0.054	-0.201	0.004	0.0128	-0.0403	-0.0352	998.008	1131.84	293
160	-0.085	-0.207	0.022	0.0128	0.1067	-0.0857	997.998	82.2058	293
585	-0.042	-0.216	0.031	-0.0016	-0.0418	-0.1244	998.002	750.418	293
294	-0.092	-0.234	0.004	-0.1394	-0.0977	0.0050	998.002	936.048	293
1	-0.090	-0.178	-0.015	-0.0347	0.0417	-0.0545	998.003	482.151	293

364	-0.115	-0.239	0.084	0.0308	0.0679	0.0585	998.003	1254	293
525	-0.063	-0.239	0.090	-0.0077	-0.0405	-0.0621	997.998	141.121	293
232	-0.108	-0.194	0.062	-0.0641	-0.0427	-0.0240	997.973	0	293
582	-0.101	-0.173	0.106	-0.0543	0.0975	-0.0090	997.922	0	293
549	-0.105	-0.207	0.097	0.0362	-0.0191	0.1468	997.998	0	293
85	-0.028	-0.178	0.103	-0.0082	0.1057	-0.0746	981.232	0	293
209	-0.088	-0.222	0.062	-0.0751	0.0022	-0.1180	998.013	2224.54	293
502	-0.092	-0.175	0.073	0.0542	-0.0195	0.0708	997.972	0	293
256	-0.031	-0.207	0.076	0.0392	-0.0421	-0.1254	998.004	1173.51	293
420	-0.061	-0.210	0.065	-0.1004	-0.0609	-0.0377	997.977	0	293
507	-0.049	-0.172	0.069	0.1011	-0.1254	-0.1109	998.006	513.804	293
310	-0.070	-0.210	0.104	-0.0123	0.0266	-0.0368	998.012	1963.14	293
167	-0.063	-0.167	0.091	-0.1616	-0.0754	0.0194	998.002	268.376	293
239	0.042	-0.238	-0.011	-0.0502	-0.0741	-0.0045	997.995	0	293
163	0.011	-0.239	0.028	-0.0388	-0.0843	-0.0018	998.002	1172.75	293
181	0.005	-0.223	-0.018	-0.0655	-0.0267	0.0243	998.009	1657.61	293
579	-0.003	-0.203	0.034	0.0896	-0.0485	0.0976	998.003	661.295	293
212	0.032	-0.167	0.040	-0.0782	-0.0230	0.0511	997.709	0	293
544	-0.020	-0.168	-0.020	0.0500	-0.0852	-0.0017	998.003	319.453	293
111	0.038	-0.207	0.035	-0.0186	0.0908	-0.1221	998.001	507.736	293
494	0.018	-0.208	-0.001	0.0385	-0.0534	-0.0822	998.001	471.817	293
455	0.026	-0.169	0.005	-0.1672	0.0694	-0.2071	995.383	0	293
532	-0.013	-0.227	0.005	0.0236	0.0276	-0.0369	997.992	0	293
165	-0.032	-0.205	-0.105	0.0475	0.0188	0.1948	985.8	0	293
118	-0.088	-0.171	-0.099	0.1464	0.1371	-0.0454	998.004	569.905	293
40	-0.027	-0.181	-0.124	-0.1029	-0.0917	-0.1092	998.001	415.655	293
536	-0.037	-0.235	-0.117	-0.0062	0.0482	-0.1418	998.001	814.781	293
391	-0.121	-0.187	-0.029	-0.0646	-0.0463	0.0520	997.246	0	293

2D Surface Profile Script

Table 45 shows the 2D surface profile extraction script.

Table 45: 2D Surface Profile Script

```
# -*- coding: utf-8 -*-
"""
Created on Tues July  1 13:49:32 2019

@author: Emerald Ryan
"""

import csv
```

```
import math
import os


#Read the file and gather information
frame = int(Scene.CurrentFrame) - 1
path = 'E:\Emerald\Optimizer\Test Code\Small_Box\Measurements\_' +
"%04d" % frame + ".csv"
if frame >= 0 and os.path.isfile(path):
    inputFile = open(path,'r')
    inputReader = csv.reader(inputFile, delimiter=',')
    data = list(inputReader)
    particlePositions1 = []
    particlePositions2 = []
    inputFile.close()
    os.remove(path)

    #Get the values wanted
    #Axis 1 is x, 2 is y, 3 is z
    axis1 = 3
    axis2 = 2
    for i in range(0, len(data)):
        if len(data[i]) > 0:
            value1 = float(data[i][axis1])
            value2 = float(data[i][axis2])
            particlePositions1.append(value1)
            particlePositions2.append(value2)

    #Determine the surface profile
    spacing = NIISphSolver_1.ParticleSize / 2
    length = len(particlePositions1)
    minimum = min(particlePositions1)
    maximum = max(particlePositions1)
    minEnd = minimum
    maxEnd = minimum + spacing
    maxs = []
    max1 = []
    max2 = []

    while minEnd < maximum:
        maxValue = -float('inf')
        maxPos = float('inf')
        for i in range(0, length):
            if particlePositions1[i] >= minEnd and
particlePositions1[i] < maxEnd:
                if particlePositions2[i] > maxValue:
                    maxValue = particlePositions2[i]
                    maxPos = i

        if maxPos != float('inf'):
            maxs.append([particlePositions1[maxPos],
particlePositions2[maxPos]])
            max1.append(particlePositions1[maxPos])
            max2.append(particlePositions2[maxPos])

        minEnd = maxEnd
        maxEnd = maxEnd + spacing
```

```
    #Write surface profile to file
    outPath = 'E:\Emerald\Optimizer\Test
Code\Small_Box\Measurements\Profile' + str(frame) + '.csv'
    with open(outPath, 'w') as outputFile:
        writer = csv.writer(outputFile, delimiter=',')
        for i in range(0, len(maxs)):
            writer.writerow(maxs[i])
```

3D Surface Profile Script

Table 46 shows the 3D surface profile extraction script.

Table 46: 3D Surface Profile Script

```
# -*- coding: utf-8 -*-
"""
Created on Tues July  1 13:49:32 2019

@author: Emerald Ryan
"""

import csv
import math
import os

#Read the file and gather information
frame = int(Scene.CurrentFrame) - 1
path = 'E:\Emerald\Optimizer\Test Code\Small_Box\Measurements\height_'
+ "%04d" % frame + ".csv"
if frame >= 0 and os.path.isfile(path):
    inputFile = open(path,'r')
    inputReader = csv.reader(inputFile, delimiter=',')
    data = list(inputReader)
    particlePositions1 = []
    particlePositions2 = []
    particlePositions3 = []
    inputFile.close()
    os.remove(path)

    #Get the values wanted
    for i in range(0, len(data)):
        value1 = float(data[i][1])
        value2 = float(data[i][2])
        value3 = float(data[i][3])
        particlePositions1.append(value1)
        particlePositions2.append(value2)
        particlePositions3.append(value3)

    #Determine the surface profile
    spacing = NIISphSolver_1.ParticleSize * 1.5
    length = len(particlePositions1)
    minimumX = min(particlePositions1)
```

```
    maximumX = max(particlePositions1)
    minimumZ = min(particlePositions3)
    maximumZ = max(particlePositions3)
    minEndX = minimumX
    maxEndX = minimumX + spacing
    minEndZ = minimumZ
    maxEndZ = minimumZ + spacing
    maxs = []
    maxX = []
    maxY = []
    maxZ = []

    while minEndX < maximumX:
        while minEndZ < maximumZ:
            maxValue = -float('inf')
            maxPos = float('inf')
            for i in range(0, length):
                if particlePositions1[i] >= minEndX and
particlePositions1[i] < maxEndX and particlePositions3[i] >= minEndZ
and particlePositions3[i] < maxEndZ:
                    if particlePositions2[i] > maxValue:
                        maxValue = particlePositions2[i]
                        maxPos = i

            if maxPos != float('inf'):
                maxs.append([particlePositions1[maxPos],
particlePositions2[maxPos], particlePositions3[maxPos]])
                maxX.append(particlePositions1[maxPos])
                maxY.append(particlePositions2[maxPos])
                maxZ.append(particlePositions3[maxPos])

            minEndZ = maxEndZ
            maxEndZ = maxEndZ + spacing

        minEndX = maxEndX
        maxEndX = maxEndX + spacing
        minEndZ = minimumZ
        maxEndZ = minimumZ + spacing


    #Write surface profile to file
    outPath = 'E:\Emerald\Optimizer\Test
Code\Small_Box\Measurements\Profile' + str(frame) + '.csv'
    with open(outPath, 'w') as outputFile:
        writer = csv.writer(outputFile, delimiter=',')
        for i in range(0, len(maxs)):
            writer.writerow(maxs[i])
```

Improved Average Height Script

Table 47 shows the improved average height script.

Table 47: Improved Average Height Script

```
# -*- coding: utf-8 -*-
"""
Created on Tues July  1 13:49:32 2019

@author: Emerald Ryan
"""

import csv
import math
import os


#Read the file and gather information
frame = int(Scene.CurrentFrame) - 1
path = 'E:\Emerald\Optimizer\Test Code\Small_Box\Measurements\height_'
+ "%04d" % frame + ".csv"
if frame >= 0 and os.path.isfile(path):
    inputFile = open(path,'r')
    inputReader = csv.reader(inputFile, delimiter=',')
    data = list(inputReader)
    particlePositions1 = []
    particlePositions2 = []
    inputFile.close()
    os.remove(path)

    #Get the values wanted
    #Axis 1 is x, 2 is y, 3 is z
    axis1 = 1
    axis2 = 2
    for i in range(0, len(data)):
        if len(data[i]) > 0:
            value1 = float(data[i][axis1])
            value2 = float(data[i][axis2])
            particlePositions1.append(value1)
            particlePositions2.append(value2)

    #Determine the surface profile
    spacing = NIISphSolver_1.ParticleSize / 2
    length = len(particlePositions1)
    minimum = min(particlePositions1)
    maximum = max(particlePositions1)
    minEnd = minimum
    maxEnd = minimum + spacing
    maxs = []
    max1 = []
    max2 = []

    while minEnd < maximum:
        maxValue = -float('inf')
        maxPos = float('inf')
        for i in range(0, length):
            if particlePositions1[i] >= minEnd and
particlePositions1[i] < maxEnd:
                if particlePositions2[i] > maxValue:
                    maxValue = particlePositions2[i]
                    maxPos = i
```

```
        if maxPos != float('inf'):
            maxs.append([particlePositions1[maxPos],
particlePositions2[maxPos]])
            max1.append(particlePositions1[maxPos])
            max2.append(particlePositions2[maxPos])

        minEnd = maxEnd
        maxEnd = maxEnd + spacing

    #Determine the average height and adjust to taking into out the
geometry coordinates
    #Bottom is the bottom position of the where the measurement is
being taken
    bottom = -0.25
    avg = sum(max2) / len(max2)
    avg = avg - bottom
    line = str(frame) + ',' + str(avg) + '\n'

    #Write surface profile to file
    outPath = 'E:\Emerald\Optimizer\Test
Code\Small_Box\Measurements\Average.csv'
    with open(outPath, 'a+') as outputFile:
        outputFile.write(line)
```

3D Surface Profile Plotting Script

Table 48 shows the Python script used to plot the 3D surface profiles.

Table 48: 3D Surface Profile Plotting Script

```
# -*- coding: utf-8 -*-
"""
Created on Tue Jul  2 09:25:24 2019

@author: Emerald
"""

from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import csv


#Read the file and gather information
#frame = int(Scene.CurrentFrame) - 1
path = r'C:\Users\Emerald\Documents\Doctoral Dissertation\Particle
Packing\height_0301.csv'
#if frame >= 0 and os.path.isfile(path):
inputFile = open(path,'r')
inputReader = csv.reader(inputFile, delimiter=',')
data = list(inputReader)
particlePositions1 = []
```

```
particlePositions2 = []
particlePositions3 = []
inputFile.close()
#os.remove(path)

#Get the values wanted
for i in range(0, len(data)):
    value1 = float(data[i][1])
    value2 = float(data[i][2])
    value3 = float(data[i][3])
    particlePositions1.append(value1)
    particlePositions2.append(value2)
    particlePositions3.append(value3)

#Read the file and gather information
#frame = int(Scene.CurrentFrame) - 1
path = r'C:\Users\Emerald\Documents\Doctoral Dissertation\Particle
Packing\Profile_2x.csv'
#if frame >= 0 and os.path.isfile(path):
inputFile = open(path,'r')
inputReader = csv.reader(inputFile, delimiter=',')
datap = list(inputReader)
particlePositions1p = []
particlePositions2p = []
particlePositions3p = []
inputFile.close()
#os.remove(path)

#Get the values wanted
for i in range(0, len(datap)):
    value1p = float(datap[i][0])
    value2p = float(datap[i][1])
    value3p = float(datap[i][2])
    particlePositions1p.append(value1p)
    particlePositions2p.append(value2p)
    particlePositions3p.append(value3p)


fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')


ax.scatter(particlePositions1, particlePositions3, particlePositions2,
c='k', marker='o')
ax.scatter(particlePositions1p, particlePositions3p,
particlePositions2p, c='r', marker='s', s=50)

ax.set_xlabel('X Position (m)')
ax.set_ylabel('Z Position (m)')
ax.set_zlabel('Y Position (m)')

plt.show()
```

Appendix G: Spillway Files

Neutrino Height Results

Table 49 shows the Neutrino height results for the last 1,500 time steps, or 30 seconds of real time, for the ten runs.

Table 49: Neutrino Height Results for Spillway Comparison

Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
25.2101	26.4861	27.6969	28.8168	30.1956	31.5762	32.8076	33.8829	35.1709	36.3285
25.2016	26.4861	27.6969	28.8151	30.2111	31.5694	32.7853	33.876	35.1451	36.3165
25.2016	26.4861	27.6969	28.8151	30.2111	31.5694	32.7853	33.876	35.1451	36.3439
25.1998	26.4827	27.7243	28.8614	30.2128	31.5916	32.8042	33.852	35.1263	36.3285
25.2016	26.4947	27.7278	28.8494	30.2059	31.6019	32.7887	33.864	35.152	36.3439
25.2016	26.4947	27.7278	28.8494	30.2059	31.6019	32.7887	33.864	35.152	36.3714
25.2136	26.4775	27.6883	28.8254	30.2351	31.5934	32.7921	33.9086	35.1846	36.3559
25.2136	26.4775	27.6883	28.8254	30.2351	31.5934	32.7921	33.9086	35.1846	36.3559
25.2153	26.4878	27.7055	28.8322	30.2351	31.5899	32.8127	33.8949	35.1966	36.3576
25.2153	26.4878	27.714	28.8219	30.2385	31.6071	32.8059	33.9035	35.1914	36.3593
25.2153	26.4878	27.714	28.8219	30.2385	31.6071	32.8059	33.9035	35.1914	36.3491
25.2084	26.4809	27.7106	28.8099	30.2402	31.6311	32.799	33.8777	35.1743	36.3816
25.2204	26.4878	27.7192	28.8133	30.2265	31.6054	32.7801	33.8915	35.164	36.3885
25.2204	26.4878	27.7192	28.8133	30.2265	31.6054	32.7801	33.8915	35.164	36.4159
25.2187	26.4827	27.7295	28.8082	30.2505	31.6071	32.7921	33.8932	35.2	36.4537
25.2273	26.4809	27.7157	28.7825	30.2419	31.5934	32.7973	33.864	35.1811	36.4159
25.2273	26.4809	27.7157	28.7825	30.2419	31.5934	32.7973	33.864	35.1811	36.4108
25.2341	26.5015	27.7157	28.7808	30.2368	31.5796	32.7921	33.8417	35.164	36.4279
25.2341	26.4827	27.7329	28.7825	30.2094	31.6002	32.7561	33.864	35.1503	36.4262
25.2341	26.4827	27.7329	28.7825	30.2094	31.6002	32.7561	33.864	35.1503	36.3988
25.2444	26.4861	27.7398	28.7739	30.2437	31.5796	32.787	33.8503	35.1263	36.4142
25.2444	26.4861	27.7398	28.7739	30.2437	31.5796	32.787	33.8503	35.1263	36.3748
25.2582	26.4758	27.7123	28.7825	30.2248	31.5642	32.7921	33.8795	35.1554	36.3816
25.2461	26.4809	27.7037	28.7911	30.2196	31.5916	32.787	33.8297	35.1588	36.4382
25.2461	26.4809	27.7037	28.7911	30.2196	31.5916	32.787	33.8297	35.1588	36.4074
25.241	26.4604	27.69	28.7825	30.1836	31.5625	32.7664	33.8263	35.1588	36.3799
25.2479	26.4638	27.6729	28.7756	30.2282	31.5556	32.7767	33.8726	35.1588	36.4022
25.2479	26.4638	27.6729	28.7756	30.2282	31.5556	32.7767	33.8726	35.1588	36.3696
25.2787	26.4638	27.6557	28.7825	30.2574	31.5985	32.7321	33.8434	35.1297	36.3971
25.2804	26.4621	27.6386	28.8168	30.2385	31.5865	32.7836	33.8263	35.1468	36.3834
25.2804	26.4621	27.6386	28.8168	30.2385	31.5865	32.7836	33.8263	35.1468	36.3765

25.2753	26.4758	27.6763	28.8254	30.2248	31.6054	32.7956	33.8434	35.1503	36.3765
25.2753	26.4758	27.6763	28.8254	30.2248	31.6054	32.7956	33.8434	35.1503	36.3919
25.2753	26.4604	27.6832	28.8356	30.2162	31.5796	32.7973	33.8657	35.1434	36.3696
25.2822	26.4724	27.69	28.7876	30.2694	31.6002	32.7664	33.8434	35.1314	36.3799
25.2822	26.4724	27.69	28.7876	30.2694	31.6002	32.7664	33.8434	35.1314	36.3645
25.301	26.4775	27.6797	28.7722	30.2557	31.5762	32.7699	33.8606	35.1228	36.4074
25.2976	26.4655	27.6849	28.7722	30.2711	31.5625	32.8042	33.84	35.1606	36.3696
25.2976	26.4655	27.6849	28.7722	30.2711	31.5625	32.8042	33.84	35.1606	36.3919
25.3199	26.4226	27.6952	28.7859	30.2677	31.5831	32.7904	33.8297	35.1503	36.3919
25.313	26.4226	27.702	28.7773	30.2419	31.6002	32.8144	33.8503	35.1537	36.3902
25.313	26.4226	27.702	28.7773	30.2419	31.6002	32.8144	33.8503	35.1537	36.3971
25.3216	26.4415	27.6986	28.7722	30.2539	31.6157	32.8179	33.864	35.1366	36.3919
25.3336	26.4329	27.69	28.7602	30.2385	31.6294	32.7647	33.8589	35.164	36.3954
25.3336	26.4329	27.69	28.7602	30.2385	31.6294	32.7647	33.8589	35.164	36.3851
25.337	26.4415	27.6694	28.7979	30.2299	31.6242	32.7887	33.8366	35.1417	36.3782
25.337	26.4415	27.6694	28.7979	30.2299	31.6242	32.7887	33.8366	35.1417	36.3576
25.337	26.4415	27.6712	28.7842	30.2488	31.6002	32.7853	33.8314	35.1571	36.3473
25.3285	26.4569	27.6832	28.7653	30.2299	31.6054	32.7939	33.8503	35.1417	36.3422
25.3285	26.4569	27.6832	28.7653	30.2299	31.6054	32.7939	33.8503	35.1417	36.3868
25.3405	26.4569	27.6746	28.7842	30.2025	31.6122	32.8059	33.8486	35.1263	36.3611
25.3525	26.4569	27.6626	28.7739	30.2248	31.6191	32.7699	33.8503	35.1434	36.3662
25.3525	26.4569	27.6626	28.7739	30.2248	31.6191	32.7699	33.8503	35.1434	36.3388
25.3508	26.4569	27.642	28.7996	30.2368	31.6259	32.8076	33.8692	35.1263	36.3714
25.3628	26.4518	27.6523	28.7945	30.2179	31.638	32.7956	33.852	35.1228	36.3542
25.3628	26.4518	27.6523	28.7945	30.2179	31.638	32.7956	33.852	35.1228	36.3611
25.3542	26.4587	27.6643	28.7739	30.2488	31.6225	32.7973	33.828	35.1245	36.3559
25.3542	26.4587	27.6643	28.7739	30.2488	31.6225	32.7973	33.828	35.1245	36.3731
25.3525	26.4621	27.6677	28.7859	30.2419	31.6019	32.7853	33.8434	35.1331	36.3593
25.3439	26.4552	27.6626	28.8322	30.2351	31.6208	32.8179	33.8383	35.1125	36.3868
25.3439	26.4552	27.6626	28.8322	30.2351	31.6208	32.8179	33.8383	35.1125	36.3816
25.3422	26.4569	27.6592	28.8459	30.2334	31.6242	32.8127	33.84	35.1125	36.3851
25.337	26.4621	27.6523	28.7688	30.2196	31.6517	32.8213	33.8383	35.1177	36.3731
25.337	26.4621	27.6523	28.7688	30.2196	31.6517	32.8213	33.8383	35.1177	36.3834
25.3422	26.4707	27.6386	28.7893	30.2402	31.6191	32.8127	33.8897	35.1297	36.3559
25.3353	26.4809	27.6712	28.7825	30.2454	31.6345	32.7939	33.8795	35.1383	36.3714
25.3353	26.4809	27.6712	28.7825	30.2454	31.6345	32.7939	33.8795	35.1383	36.3456
25.3439	26.4689	27.6609	28.7396	30.2488	31.6242	32.8007	33.8589	35.1074	36.3748
25.3473	26.4672	27.6609	28.743	30.2505	31.6242	32.7904	33.8726	35.1314	36.3405
25.3473	26.4672	27.6609	28.743	30.2505	31.6242	32.7904	33.8726	35.1314	36.3285
25.337	26.4809	27.6832	28.7465	30.2591	31.6174	32.8247	33.864	35.1091	36.3319
25.337	26.4809	27.6832	28.7465	30.2591	31.6174	32.8247	33.864	35.1091	36.3611
25.3439	26.4758	27.69	28.7482	30.2934	31.6242	32.8007	33.8675	35.1005	36.3405
25.325	26.4724	27.6609	28.7619	30.2539	31.6088	32.8333	33.8537	35.0971	36.3576

25.325	26.4724	27.6609	28.7619	30.2539	31.6088	32.8333	33.8537	35.0971	36.3405
25.3285	26.4518	27.702	28.7413	30.2865	31.6259	32.787	33.8417	35.1125	36.3628
25.3319	26.4569	27.7003	28.7516	30.2591	31.6157	32.787	33.8503	35.1623	36.3851
25.3319	26.4569	27.7003	28.7516	30.2591	31.6157	32.787	33.8503	35.1623	36.3593
25.3302	26.4758	27.69	28.7653	30.2694	31.6242	32.7767	33.864	35.152	36.3405
25.3302	26.4724	27.7003	28.779	30.2677	31.6088	32.8144	33.8709	35.128	36.3542
25.3302	26.4724	27.7003	28.779	30.2677	31.6088	32.8144	33.8709	35.128	36.3422
25.3302	26.4981	27.6986	28.7465	30.2368	31.6311	32.799	33.8417	35.1331	36.3645
25.3302	26.4981	27.6986	28.7465	30.2368	31.6311	32.799	33.8417	35.1331	36.3559
25.3285	26.4827	27.6917	28.7705	30.2196	31.6105	32.8144	33.8743	35.0954	36.3611
25.325	26.4638	27.6729	28.7739	30.2111	31.6174	32.835	33.888	35.1194	36.3422
25.325	26.4638	27.6729	28.7739	30.2111	31.6174	32.835	33.888	35.1194	36.3268
25.3165	26.4707	27.6917	28.7619	30.2179	31.6362	32.7801	33.8554	35.1486	36.3268
25.3353	26.4587	27.6832	28.7825	30.2145	31.638	32.8247	33.8692	35.1348	36.3679
25.3353	26.4587	27.6832	28.7825	30.2145	31.638	32.8247	33.8692	35.1348	36.3388
25.349	26.4707	27.6643	28.7842	30.2231	31.6139	32.7801	33.8777	35.1486	36.3353
25.3473	26.4689	27.6643	28.7568	30.2145	31.6208	32.7561	33.9138	35.128	36.3216
25.3473	26.4689	27.6643	28.7568	30.2145	31.6208	32.7561	33.9138	35.128	36.3233
25.3593	26.4638	27.6763	28.7619	30.2317	31.6242	32.7596	33.864	35.1417	36.3268
25.3525	26.4809	27.6814	28.7722	30.2128	31.6345	32.7784	33.8383	35.1486	36.3336
25.3525	26.4809	27.6814	28.7722	30.2128	31.6345	32.7784	33.8383	35.1486	36.3148
25.3593	26.493	27.6917	28.7825	30.2076	31.6414	32.8127	33.8332	35.1348	36.3096
25.3593	26.493	27.6917	28.7825	30.2076	31.6414	32.8127	33.8332	35.1348	36.3268
25.3576	26.4998	27.6952	28.7825	30.2008	31.6568	32.8076	33.8109	35.1417	36.3491
25.3628	26.5101	27.6694	28.767	30.2059	31.6345	32.8059	33.7971	35.1417	36.3388
25.3628	26.5101	27.6694	28.767	30.2059	31.6345	32.8059	33.7971	35.1417	36.3559
25.3456	26.5324	27.6883	28.7756	30.2059	31.65	32.8179	33.8332	35.1451	36.3371
25.3473	26.5204	27.678	28.7653	30.2111	31.6294	32.823	33.8091	35.1314	36.3336
25.3473	26.5204	27.678	28.7653	30.2111	31.6294	32.823	33.8091	35.1314	36.3714
25.3748	26.5135	27.654	28.7808	30.1853	31.6362	32.8367	33.8211	35.1537	36.3525
25.3782	26.5187	27.6592	28.8116	30.1853	31.6139	32.8093	33.816	35.1383	36.3439
25.3782	26.5187	27.6592	28.8116	30.1853	31.6139	32.8093	33.816	35.1383	36.3422
25.3868	26.5273	27.6403	28.8151	30.2351	31.6551	32.823	33.8383	35.1554	36.3216
25.3868	26.5273	27.6403	28.8151	30.2351	31.6551	32.823	33.8383	35.1554	36.3405
25.3919	26.529	27.6249	28.8219	30.2299	31.6397	32.8093	33.792	35.1691	36.3233
25.3902	26.5307	27.618	28.7945	30.2162	31.6242	32.8007	33.8417	35.1468	36.3405
25.3902	26.5307	27.618	28.7945	30.2162	31.6242	32.8007	33.8417	35.1468	36.2873
25.3988	26.5324	27.642	28.7808	30.2094	31.638	32.8162	33.8589	35.1383	36.301
25.4074	26.4947	27.642	28.7653	30.2471	31.6242	32.8007	33.8966	35.1743	36.3028
25.4074	26.4947	27.642	28.7653	30.2471	31.6242	32.8007	33.8966	35.1743	36.2633
25.4142	26.529	27.6283	28.779	30.2231	31.6294	32.7939	33.8966	35.1297	36.277
25.4142	26.5444	27.5974	28.8065	30.2317	31.6362	32.8024	33.888	35.1297	36.2342
25.4142	26.5444	27.5974	28.8065	30.2317	31.6362	32.8024	33.888	35.1297	36.2685

25.4056	26.5135	27.6386	28.7876	30.1819	31.6311	32.8042	33.9035	35.1726	36.2599
25.4005	26.5101	27.6506	28.779	30.1991	31.6362	32.8453	33.9103	35.164	36.2496
25.4005	26.5101	27.6506	28.779	30.1991	31.6362	32.8453	33.9103	35.164	36.2564
25.3971	26.5067	27.6437	28.7499	30.1974	31.6328	32.835	33.8743	35.1674	36.2547
25.3971	26.5067	27.6437	28.7499	30.1974	31.6328	32.835	33.8743	35.1674	36.2239
25.4005	26.5187	27.6574	28.7602	30.2299	31.6122	32.8282	33.8932	35.1794	36.2221
25.3902	26.4947	27.6197	28.7945	30.2025	31.6088	32.8282	33.8554	35.1468	36.2256
25.3902	26.4947	27.6197	28.7945	30.2025	31.6088	32.8282	33.8554	35.1468	36.2239
25.3936	26.4775	27.6283	28.7842	30.2111	31.6105	32.8042	33.84	35.1743	36.2342
25.3885	26.4689	27.6283	28.7516	30.1888	31.6088	32.8179	33.8366	35.1743	36.2393
25.3885	26.4689	27.6283	28.7516	30.1888	31.6088	32.8179	33.8366	35.1743	36.2256
25.3851	26.4947	27.6214	28.7636	30.1699	31.6054	32.7956	33.8074	35.1571	36.241
25.3799	26.5032	27.6111	28.767	30.1408	31.6002	32.7921	33.8246	35.1691	36.2427
25.3799	26.5032	27.6111	28.767	30.1408	31.6002	32.7921	33.8246	35.1691	36.2633
25.3713	26.505	27.6197	28.7859	30.151	31.5848	32.799	33.84	35.1434	36.2513
25.3713	26.505	27.6197	28.7859	30.151	31.5848	32.799	33.84	35.1434	36.2547
25.3542	26.505	27.5991	28.7739	30.1459	31.5762	32.8093	33.8297	35.1691	36.2547
25.349	26.4912	27.6317	28.7705	30.1768	31.6071	32.8316	33.8589	35.1606	36.2753
25.349	26.4912	27.6317	28.7705	30.1768	31.6071	32.8316	33.8589	35.1606	36.2359
25.3388	26.4895	27.6334	28.7739	30.1819	31.6259	32.8144	33.8229	35.1537	36.2427
25.3439	26.4827	27.6231	28.7773	30.1785	31.5728	32.8402	33.828	35.1434	36.265
25.3439	26.4827	27.6231	28.7773	30.1785	31.5728	32.8402	33.828	35.1434	36.2856
25.3473	26.4758	27.6386	28.7619	30.1871	31.5916	32.8693	33.828	35.164	36.2856
25.3473	26.4689	27.6386	28.7705	30.1905	31.5728	32.8213	33.84	35.1794	36.265
25.3473	26.4689	27.6386	28.7705	30.1905	31.5728	32.8213	33.84	35.1794	36.2736
25.3508	26.5032	27.6437	28.7447	30.2059	31.5951	32.7939	33.8434	35.2069	36.2719
25.3473	26.5238	27.6643	28.7447	30.2317	31.6071	32.799	33.8177	35.2103	36.2564
25.3473	26.5238	27.6643	28.7447	30.2317	31.6071	32.799	33.8177	35.2103	36.2667
25.3748	26.5204	27.642	28.7225	30.2162	31.6208	32.7784	33.8675	35.2034	36.2719
25.3748	26.5204	27.642	28.7225	30.2162	31.6208	32.7784	33.8675	35.2034	36.3182
25.3936	26.505	27.6557	28.7362	30.2111	31.6294	32.7956	33.8469	35.164	36.2805
25.3971	26.4981	27.642	28.7842	30.1905	31.6654	32.8024	33.8434	35.1897	36.2839
25.3971	26.4981	27.642	28.7842	30.1905	31.6654	32.8024	33.8434	35.1897	36.3062
25.4022	26.5032	27.6146	28.7773	30.1699	31.6311	32.7939	33.84	35.1897	36.3233
25.4039	26.4964	27.618	28.7722	30.1802	31.6328	32.7819	33.8537	35.1983	36.325
25.4039	26.4964	27.618	28.7722	30.1802	31.6328	32.7819	33.8537	35.1983	36.3216
25.4074	26.4964	27.6626	28.7362	30.1528	31.6191	32.7887	33.8211	35.176	36.325
25.3971	26.4998	27.6557	28.7259	30.1562	31.6208	32.8093	33.7971	35.2086	36.3491
25.3971	26.4998	27.6557	28.7259	30.1562	31.6208	32.8093	33.7971	35.2086	36.3268
25.3936	26.493	27.6694	28.7756	30.1545	31.6191	32.751	33.8194	35.1743	36.3473
25.3936	26.493	27.6694	28.7756	30.1545	31.6191	32.751	33.8194	35.1743	36.3319
25.3868	26.5015	27.6797	28.7688	30.1545	31.6002	32.7441	33.828	35.1931	36.3353
25.4005	26.4809	27.6557	28.7447	30.1631	31.6019	32.8007	33.828	35.2206	36.3508

25.4005	26.4809	27.6557	28.7447	30.1631	31.6019	32.8007	33.828	35.2206	36.3233
25.4005	26.4758	27.6471	28.7447	30.1802	31.6002	32.787	33.816	35.2103	36.3353
25.4194	26.5101	27.6592	28.7396	30.1974	31.5694	32.7681	33.8263	35.1846	36.2719
25.4194	26.5101	27.6592	28.7396	30.1974	31.5694	32.7681	33.8263	35.1846	36.289
25.4228	26.5135	27.6729	28.7499	30.2368	31.5625	32.7973	33.8297	35.1777	36.2582
25.4228	26.517	27.69	28.7533	30.2402	31.5796	32.7887	33.84	35.1863	36.2547
25.4228	26.517	27.69	28.7533	30.2402	31.5796	32.7887	33.84	35.1863	36.2513
25.4074	26.5204	27.69	28.7465	30.2505	31.5711	32.7921	33.8486	35.1931	36.2702
25.4091	26.5118	27.6832	28.7447	30.2419	31.5796	32.8076	33.8537	35.2017	36.3045
25.4091	26.5118	27.6832	28.7447	30.2419	31.5796	32.8076	33.8537	35.2017	36.2753
25.4056	26.5015	27.6952	28.7482	30.2317	31.6105	32.7801	33.8572	35.2154	36.2633
25.4056	26.5015	27.6952	28.7482	30.2317	31.6105	32.7801	33.8572	35.2154	36.2753
25.3988	26.5032	27.7003	28.743	30.2368	31.6071	32.8024	33.9138	35.2223	36.2942
25.4005	26.4861	27.6952	28.743	30.2214	31.5968	32.799	33.8915	35.2223	36.2839
25.4005	26.4861	27.6952	28.743	30.2214	31.5968	32.799	33.8915	35.2223	36.301
25.3851	26.4792	27.666	28.7602	30.1991	31.5848	32.8179	33.8829	35.2223	36.2907
25.3765	26.4655	27.6643	28.7773	30.1836	31.6054	32.8059	33.9069	35.2446	36.2873
25.3765	26.4655	27.6643	28.7773	30.1836	31.6054	32.8059	33.9069	35.2446	36.2719
25.3988	26.493	27.6677	28.7945	30.1922	31.6122	32.8196	33.8675	35.2172	36.2753
25.4022	26.4809	27.6677	28.779	30.2042	31.6037	32.8282	33.8846	35.212	36.2925
25.4022	26.4809	27.6677	28.779	30.2042	31.6037	32.8282	33.8846	35.212	36.3028
25.4091	26.4758	27.6797	28.7756	30.2179	31.5694	32.8522	33.8726	35.176	36.289
25.4091	26.4758	27.6797	28.7756	30.2179	31.5694	32.8522	33.8726	35.176	36.2856
25.3885	26.4792	27.6609	28.7773	30.2351	31.5831	32.8213	33.8657	35.1726	36.2685
25.3988	26.4741	27.6883	28.7876	30.2162	31.5608	32.8059	33.8846	35.1743	36.3165
25.3988	26.4741	27.6883	28.7876	30.2162	31.5608	32.8059	33.8846	35.1743	36.3302
25.4142	26.4844	27.678	28.7979	30.1939	31.5659	32.8007	33.8983	35.1709	36.3216
25.4194	26.4809	27.666	28.7859	30.2248	31.5985	32.8024	33.888	35.2052	36.301
25.4194	26.4809	27.666	28.7859	30.2248	31.5985	32.8024	33.888	35.2052	36.2873
25.4228	26.4844	27.7123	28.7739	30.2334	31.638	32.7801	33.9069	35.1777	36.3045
25.4211	26.4895	27.6797	28.7876	30.2128	31.6139	32.775	33.9086	35.1931	36.2942
25.4211	26.4895	27.6797	28.7876	30.2128	31.6139	32.775	33.9086	35.1931	36.3268
25.4485	26.4895	27.6763	28.7996	30.2385	31.5916	32.7887	33.8949	35.1726	36.325
25.4434	26.4775	27.678	28.7979	30.2539	31.6191	32.7973	33.8726	35.1863	36.3336
25.4434	26.4775	27.678	28.7979	30.2539	31.6191	32.7973	33.8726	35.1863	36.3473
25.4451	26.4792	27.678	28.7842	30.1939	31.6294	32.7904	33.876	35.2	36.3559
25.4451	26.4792	27.678	28.7842	30.1939	31.6294	32.7904	33.876	35.2	36.3319
25.4382	26.4827	27.654	28.7996	30.2111	31.6208	32.8059	33.8606	35.1863	36.3216
25.4468	26.4792	27.6712	28.7773	30.2299	31.6019	32.7904	33.8452	35.2086	36.3834
25.4468	26.4792	27.6712	28.7773	30.2299	31.6019	32.7904	33.8452	35.2086	36.3542
25.4485	26.493	27.666	28.7585	30.2162	31.5796	32.799	33.8623	35.2326	36.3611
25.4502	26.5032	27.6574	28.7568	30.2231	31.5951	32.7699	33.8846	35.224	36.3954
25.4502	26.5032	27.6574	28.7568	30.2231	31.5951	32.7699	33.8846	35.224	36.3868

25.4485	26.493	27.678	28.7619	30.1888	31.6122	32.7424	33.852	35.2395	36.3559
25.4365	26.5067	27.6523	28.7653	30.1648	31.6105	32.751	33.8726	35.2069	36.3645
25.4365	26.5067	27.6523	28.7653	30.1648	31.6105	32.751	33.8726	35.2069	36.3696
25.4605	26.5118	27.6746	28.755	30.1802	31.6054	32.7321	33.8863	35.2069	36.3851
25.4605	26.5118	27.6746	28.755	30.1802	31.6054	32.7321	33.8863	35.2069	36.3593
25.4571	26.5152	27.6935	28.7619	30.2145	31.6225	32.7441	33.8726	35.188	36.3336
25.4571	26.4964	27.6712	28.7893	30.2282	31.6037	32.6978	33.8503	35.1949	36.3576
25.4571	26.4964	27.6712	28.7893	30.2282	31.6037	32.6978	33.8503	35.1949	36.3353
25.4605	26.493	27.6592	28.7842	30.2282	31.5985	32.6858	33.8417	35.1846	36.3508
25.4502	26.5032	27.678	28.7825	30.2334	31.6174	32.7458	33.816	35.1623	36.3473
25.4502	26.5032	27.678	28.7825	30.2334	31.6174	32.7458	33.816	35.1623	36.3902
25.4434	26.4981	27.6849	28.7465	30.2419	31.6277	32.799	33.8589	35.1726	36.3851
25.4399	26.4981	27.6866	28.7499	30.2419	31.6037	32.7956	33.84	35.1966	36.4125
25.4399	26.4981	27.6866	28.7499	30.2419	31.6037	32.7956	33.84	35.1966	36.3919
25.4468	26.4792	27.7037	28.7619	30.2282	31.6019	32.7887	33.8211	35.1794	36.4262
25.4537	26.4741	27.6746	28.7773	30.2128	31.5951	32.7716	33.8469	35.2	36.44
25.4537	26.4741	27.6746	28.7773	30.2128	31.5951	32.7716	33.8469	35.2	36.4279
25.4554	26.4672	27.6814	28.7636	30.2385	31.6037	32.7939	33.8503	35.188	36.4365
25.4554	26.4672	27.6814	28.7636	30.2385	31.6037	32.7939	33.8503	35.188	36.3936
25.4571	26.4501	27.6814	28.7533	30.2454	31.6242	32.7819	33.8657	35.2103	36.4108
25.4417	26.4535	27.6712	28.7516	30.2299	31.6105	32.8076	33.8469	35.2223	36.4039
25.4417	26.4535	27.6712	28.7516	30.2299	31.6105	32.8076	33.8469	35.2223	36.3988
25.4485	26.4672	27.6763	28.7568	30.2351	31.6037	32.8007	33.84	35.2343	36.3851
25.4502	26.4724	27.6574	28.7602	30.2317	31.6208	32.8024	33.8263	35.2069	36.3936
25.4502	26.4724	27.6574	28.7602	30.2317	31.6208	32.8024	33.8263	35.2069	36.3782
25.4365	26.4638	27.6866	28.7447	30.2265	31.5985	32.7819	33.8383	35.1451	36.3919
25.4399	26.4758	27.6935	28.7516	30.2471	31.6208	32.7681	33.8349	35.1777	36.3593
25.4399	26.4758	27.6935	28.7516	30.2471	31.6208	32.7681	33.8349	35.1777	36.3782
25.4468	26.4621	27.6917	28.7327	30.2677	31.6191	32.7664	33.8606	35.1794	36.3919
25.4468	26.4621	27.6917	28.7327	30.2677	31.6191	32.7664	33.8606	35.1794	36.4005
25.4519	26.4569	27.6797	28.7619	30.2951	31.6071	32.7338	33.864	35.1846	36.4245
25.4382	26.4398	27.6643	28.7533	30.2797	31.5951	32.7716	33.8383	35.1709	36.4348
25.4382	26.4398	27.6643	28.7533	30.2797	31.5951	32.7716	33.8383	35.1709	36.4142
25.4348	26.4398	27.6557	28.7568	30.2831	31.6002	32.7664	33.8692	35.1709	36.4194
25.4331	26.4535	27.666	28.7636	30.2437	31.5968	32.7801	33.864	35.1468	36.4571
25.4331	26.4535	27.666	28.7636	30.2437	31.5968	32.7801	33.864	35.1468	36.464
25.4297	26.4501	27.6763	28.7602	30.2711	31.5899	32.7561	33.9018	35.1245	36.4965
25.4211	26.4432	27.6849	28.7602	30.2694	31.5814	32.7613	33.888	35.1486	36.4622
25.4211	26.4432	27.6849	28.7602	30.2694	31.5814	32.7613	33.888	35.1486	36.488
25.4056	26.4449	27.6917	28.7482	30.2677	31.5934	32.763	33.8503	35.116	36.5086
25.4108	26.4312	27.6694	28.7585	30.2625	31.6345	32.7596	33.8709	35.1331	36.4468
25.4108	26.4312	27.6694	28.7585	30.2625	31.6345	32.7596	33.8709	35.1331	36.4468
25.4211	26.4312	27.666	28.7979	30.2865	31.5882	32.7853	33.8623	35.116	36.4588

25.4211	26.4312	27.666	28.7979	30.2865	31.5882	32.7853	33.8623	35.116	36.4228
25.4159	26.4158	27.6592	28.7979	30.2814	31.6088	32.8316	33.84	35.1057	36.4297
25.4125	26.4381	27.6557	28.7825	30.2625	31.65	32.8453	33.8366	35.1194	36.4417
25.4125	26.4381	27.6557	28.7825	30.2625	31.65	32.8453	33.8366	35.1194	36.4348
25.3954	26.4192	27.6609	28.7808	30.2008	31.6345	32.7819	33.8006	35.1005	36.4537
25.3988	26.4209	27.6592	28.7808	30.2231	31.65	32.8196	33.8297	35.0885	36.4605
25.3988	26.4209	27.6592	28.7808	30.2231	31.65	32.8196	33.8297	35.0885	36.4451
25.4022	26.4398	27.6523	28.8219	30.2402	31.6277	32.8144	33.8452	35.1348	36.4948
25.4022	26.4226	27.666	28.7962	30.2248	31.6259	32.8264	33.8554	35.1537	36.4897
25.4022	26.4226	27.666	28.7962	30.2248	31.6259	32.8264	33.8554	35.1537	36.5
25.4005	26.4312	27.6694	28.7808	30.2437	31.6174	32.895	33.8246	35.128	36.4708
25.4005	26.4312	27.6694	28.7808	30.2437	31.6174	32.895	33.8246	35.128	36.4863
25.3954	26.4381	27.6694	28.7825	30.2179	31.6225	32.823	33.8589	35.1588	36.4777
25.3954	26.4552	27.6952	28.7979	30.2402	31.6088	32.8505	33.8229	35.1366	36.4725
25.3954	26.4552	27.6952	28.7979	30.2402	31.6088	32.8505	33.8229	35.1366	36.4691
25.4022	26.4587	27.7003	28.8116	30.2591	31.6071	32.8316	33.8589	35.1674	36.4845
25.4108	26.4621	27.6814	28.7842	30.2059	31.6019	32.8385	33.8537	35.1606	36.4605
25.4108	26.4621	27.6814	28.7842	30.2059	31.6019	32.8385	33.8537	35.1606	36.4468
25.4108	26.4518	27.6952	28.7876	30.2351	31.6019	32.847	33.8606	35.1794	36.4451
25.4091	26.4604	27.6935	28.7825	30.2299	31.6174	32.8385	33.8726	35.2	36.4502
25.4091	26.4604	27.6935	28.7825	30.2299	31.6174	32.8385	33.8726	35.2	36.4382
25.4022	26.4449	27.6866	28.7705	30.2522	31.6345	32.835	33.8434	35.1503	36.4485
25.4091	26.4638	27.6832	28.7705	30.2419	31.65	32.8419	33.8572	35.1949	36.488
25.4091	26.4638	27.6832	28.7705	30.2419	31.65	32.8419	33.8572	35.1949	36.4983
25.3988	26.4775	27.6609	28.7636	30.2437	31.6225	32.835	33.8314	35.1709	36.4931
25.3988	26.4775	27.6609	28.7636	30.2437	31.6225	32.835	33.8314	35.1709	36.4828
25.4039	26.4775	27.6506	28.7876	30.2368	31.6414	32.847	33.8314	35.1451	36.4931
25.3988	26.4621	27.6557	28.7876	30.2248	31.6122	32.8453	33.8554	35.1263	36.4811
25.3988	26.4621	27.6557	28.7876	30.2248	31.6122	32.8453	33.8554	35.1263	36.5308
25.3833	26.4518	27.6369	28.7893	30.1991	31.662	32.8385	33.8452	35.1177	36.5034
25.3799	26.4552	27.6386	28.8151	30.2317	31.6311	32.8367	33.852	35.1297	36.488
25.3799	26.4552	27.6386	28.8151	30.2317	31.6311	32.8367	33.852	35.1297	36.4983
25.3868	26.4415	27.6523	28.8048	30.2179	31.6071	32.8659	33.8554	35.1468	36.5051
25.3816	26.4346	27.6746	28.8082	30.2231	31.5985	32.811	33.8297	35.1383	36.4863
25.3816	26.4346	27.6746	28.8082	30.2231	31.5985	32.811	33.8297	35.1383	36.4743
25.3748	26.4552	27.666	28.7893	30.2145	31.6431	32.8196	33.8417	35.1245	36.5308
25.3748	26.4552	27.666	28.7893	30.2145	31.6431	32.8196	33.8417	35.1245	36.5086
25.3662	26.4449	27.6694	28.7653	30.2248	31.6157	32.8042	33.8692	35.128	36.512
25.3422	26.4569	27.6729	28.743	30.1991	31.6002	32.8007	33.8314	35.128	36.4897
25.3422	26.4569	27.6729	28.743	30.1991	31.6002	32.8007	33.8314	35.128	36.4828
25.3525	26.4466	27.678	28.7619	30.2179	31.6345	32.7836	33.8177	35.128	36.4828
25.3593	26.4415	27.6952	28.7533	30.2402	31.6414	32.8282	33.8452	35.1314	36.4897
25.3593	26.4415	27.6952	28.7533	30.2402	31.6414	32.8282	33.8452	35.1314	36.5188

25.3559	26.4518	27.666	28.7911	30.2179	31.6139	32.8093	33.8572	35.1228	36.4794
25.3525	26.4587	27.6643	28.7705	30.2522	31.5916	32.8299	33.8332	35.1194	36.4554
25.3525	26.4587	27.6643	28.7705	30.2522	31.5916	32.8299	33.8332	35.1194	36.4485
25.3473	26.4741	27.666	28.7225	30.2454	31.6157	32.7853	33.8417	35.1348	36.4777
25.3336	26.4638	27.678	28.7413	30.2385	31.6277	32.787	33.8777	35.1486	36.4451
25.3336	26.4638	27.678	28.7413	30.2385	31.6277	32.787	33.8777	35.1486	36.4811
25.3182	26.4518	27.6626	28.7722	30.2454	31.6311	32.8127	33.8332	35.1383	36.4777
25.3182	26.4518	27.6626	28.7722	30.2454	31.6311	32.8127	33.8332	35.1383	36.5257
25.3233	26.4466	27.678	28.7705	30.2059	31.5951	32.7939	33.8074	35.1297	36.4554
25.3336	26.4552	27.6626	28.7705	30.1991	31.5831	32.823	33.792	35.1366	36.4897
25.3336	26.4552	27.6626	28.7705	30.1991	31.5831	32.823	33.792	35.1366	36.4777
25.3147	26.4587	27.6746	28.7636	30.2111	31.5934	32.8264	33.8486	35.1331	36.4039
25.3113	26.4569	27.7037	28.7499	30.1871	31.6019	32.8299	33.8692	35.1468	36.4708
25.3113	26.4569	27.7037	28.7499	30.1871	31.6019	32.8299	33.8692	35.1468	36.4828
25.301	26.4587	27.6952	28.743	30.2179	31.5848	32.8316	33.8383	35.1314	36.488
25.3027	26.4604	27.6746	28.7225	30.2557	31.5865	32.8196	33.8572	35.1297	36.4605
25.3027	26.4604	27.6746	28.7225	30.2557	31.5865	32.8196	33.8572	35.1297	36.4708
25.2942	26.4569	27.6814	28.7259	30.2248	31.5745	32.8007	33.8743	35.1143	36.4863
25.2942	26.4569	27.6814	28.7259	30.2248	31.5745	32.8007	33.8743	35.1143	36.4863
25.2839	26.4638	27.6866	28.719	30.2351	31.5985	32.8247	33.8417	35.0748	36.5154
25.2959	26.4364	27.6694	28.7568	30.2196	31.5848	32.8247	33.8743	35.1143	36.5017
25.2959	26.4364	27.6694	28.7568	30.2196	31.5848	32.8247	33.8743	35.1143	36.4914
25.2976	26.4329	27.6832	28.7739	30.2265	31.6191	32.8264	33.8675	35.0851	36.4794
25.2959	26.4381	27.6917	28.7705	30.266	31.5968	32.8367	33.8675	35.1194	36.4674
25.2959	26.4381	27.6917	28.7705	30.266	31.5968	32.8367	33.8675	35.1194	36.5017
25.289	26.4432	27.6729	28.767	30.2505	31.6397	32.8333	33.8692	35.1108	36.4657
25.2907	26.4346	27.6797	28.7516	30.2539	31.638	32.8367	33.8915	35.1417	36.4348
25.2907	26.4346	27.6797	28.7516	30.2539	31.638	32.8367	33.8915	35.1417	36.4794
25.2907	26.4484	27.6694	28.7276	30.2539	31.6414	32.8419	33.8606	35.1245	36.4691
25.289	26.4689	27.678	28.7568	30.2317	31.6311	32.8213	33.8537	35.1177	36.4828
25.289	26.4689	27.678	28.7568	30.2317	31.6311	32.8213	33.8537	35.1177	36.4571
25.277	26.4947	27.6986	28.767	30.2317	31.6174	32.8144	33.8692	35.1177	36.4948
25.277	26.4947	27.6986	28.767	30.2317	31.6174	32.8144	33.8692	35.1177	36.4194
25.2736	26.4912	27.678	28.7705	30.2265	31.5899	32.7801	33.8417	35.0902	36.4674
25.265	26.4947	27.6746	28.7688	30.2299	31.6397	32.8024	33.8503	35.1091	36.4417
25.265	26.4947	27.6746	28.7688	30.2299	31.6397	32.8024	33.8503	35.1091	36.4434
25.2427	26.4827	27.6763	28.7568	30.2179	31.6328	32.8264	33.8417	35.1314	36.4262
25.2444	26.4861	27.6986	28.7602	30.2454	31.674	32.8179	33.8314	35.1091	36.4297
25.2444	26.4861	27.6986	28.7602	30.2454	31.674	32.8179	33.8314	35.1091	36.4485
25.2307	26.4895	27.6797	28.7756	30.2694	31.674	32.8179	33.8349	35.1503	36.4382
25.2341	26.4809	27.6626	28.7996	30.2762	31.6774	32.8453	33.8314	35.1606	36.4331
25.2341	26.4809	27.6626	28.7996	30.2762	31.6774	32.8453	33.8314	35.1606	36.4554
25.2256	26.4981	27.6694	28.7825	30.2368	31.6774	32.8385	33.84	35.1571	36.4502

25.2256	26.4981	27.6694	28.7825	30.2368	31.6774	32.8385	33.84	35.1571	36.4794
25.2101	26.4861	27.6609	28.7688	30.2265	31.6517	32.8522	33.8383	35.1503	36.4588
25.229	26.4998	27.6814	28.7773	30.2265	31.6088	32.8213	33.876	35.1623	36.4674
25.229	26.4998	27.6814	28.7773	30.2265	31.6088	32.8213	33.876	35.1623	36.4434
25.2359	26.5032	27.6712	28.7619	30.2265	31.6277	32.8264	33.8657	35.1709	36.4502
25.2324	26.4998	27.6814	28.7842	30.2368	31.6414	32.799	33.8657	35.1314	36.4262
25.2324	26.4998	27.6814	28.7842	30.2368	31.6414	32.799	33.8657	35.1314	36.4074
25.2273	26.4895	27.702	28.7962	30.2248	31.6242	32.7904	33.8006	35.1245	36.4262
25.2187	26.4758	27.726	28.7842	30.2299	31.5916	32.7939	33.8023	35.0988	36.4245
25.2187	26.4758	27.726	28.7842	30.2299	31.5916	32.7939	33.8023	35.0988	36.4245
25.2256	26.493	27.7243	28.7688	30.2094	31.6105	32.8316	33.8057	35.1177	36.4074
25.2239	26.4844	27.7106	28.7876	30.2196	31.6414	32.8316	33.7937	35.164	36.3988
25.2239	26.4844	27.7106	28.7876	30.2196	31.6414	32.8316	33.7937	35.164	36.3816
25.2239	26.4792	27.7072	28.7756	30.2111	31.6482	32.8196	33.8143	35.176	36.3765
25.2239	26.4792	27.7072	28.7756	30.2111	31.6482	32.8196	33.8143	35.176	36.3902
25.217	26.4981	27.7157	28.7876	30.1768	31.6311	32.8093	33.8332	35.152	36.3954
25.193	26.4964	27.6866	28.7568	30.1802	31.5951	32.811	33.8006	35.1571	36.3696
25.193	26.4964	27.6866	28.7568	30.1802	31.5951	32.811	33.8006	35.1571	36.3902
25.205	26.4861	27.7055	28.7447	30.1853	31.6088	32.7956	33.804	35.1863	36.3731
25.2033	26.4964	27.7055	28.7705	30.1888	31.6002	32.8024	33.8349	35.2	36.3696
25.2033	26.4964	27.7055	28.7705	30.1888	31.6002	32.8024	33.8349	35.2	36.3851
25.1981	26.4912	27.7037	28.7619	30.1991	31.6054	32.7853	33.8194	35.1606	36.3799
25.2101	26.4844	27.702	28.7516	30.2025	31.6054	32.8042	33.8177	35.152	36.3491
25.2101	26.4844	27.702	28.7516	30.2025	31.6054	32.8042	33.8177	35.152	36.3388
25.2101	26.4707	27.7226	28.767	30.1819	31.6054	32.7716	33.8349	35.1023	36.3971
25.2101	26.4707	27.7226	28.767	30.1819	31.6054	32.7716	33.8349	35.1023	36.3799
25.217	26.4672	27.714	28.7585	30.1802	31.5848	32.7939	33.8143	35.1468	36.3851
25.2187	26.4741	27.6917	28.7619	30.2025	31.5762	32.8453	33.7868	35.152	36.3919
25.2187	26.4741	27.6917	28.7619	30.2025	31.5762	32.8453	33.7868	35.152	36.3988
25.2187	26.4672	27.7243	28.7396	30.1956	31.6071	32.8042	33.7989	35.1623	36.3936
25.2153	26.4947	27.7209	28.7568	30.2094	31.5848	32.799	33.7851	35.1434	36.4074
25.2153	26.4947	27.7209	28.7568	30.2094	31.5848	32.799	33.7851	35.1434	36.4125
25.2101	26.4964	27.6935	28.755	30.1631	31.5814	32.7956	33.8246	35.14	36.3902
25.2153	26.4827	27.7243	28.7773	30.1905	31.6019	32.8076	33.792	35.1829	36.3799
25.2153	26.4827	27.7243	28.7773	30.1905	31.6019	32.8076	33.792	35.1829	36.3422
25.2256	26.4689	27.7123	28.7859	30.2008	31.5985	32.8144	33.816	35.1451	36.3456
25.2324	26.4638	27.7449	28.7893	30.2231	31.5985	32.8127	33.8109	35.1571	36.3182
25.2324	26.4638	27.7449	28.7893	30.2231	31.5985	32.8127	33.8109	35.1571	36.3216
25.2273	26.4569	27.7415	28.7842	30.1888	31.6088	32.7853	33.828	35.164	36.301
25.2273	26.4569	27.7415	28.7842	30.1888	31.6088	32.7853	33.828	35.164	36.3662
25.2221	26.4398	27.7398	28.7756	30.1733	31.6071	32.8436	33.8143	35.1829	36.3593
25.229	26.4449	27.7483	28.7842	30.2059	31.5934	32.7819	33.8349	35.1914	36.3628
25.229	26.4449	27.7483	28.7842	30.2059	31.5934	32.7819	33.8349	35.1914	36.3799

25.2444	26.4569	27.7483	28.7739	30.1939	31.5916	32.787	33.8109	35.1743	36.3748
25.2427	26.4552	27.7295	28.7636	30.1871	31.5934	32.7819	33.8177	35.212	36.3508
25.2427	26.4552	27.7295	28.7636	30.1871	31.5934	32.7819	33.8177	35.212	36.3473
25.2444	26.4398	27.7175	28.7447	30.1991	31.6105	32.7819	33.828	35.2446	36.3662
25.253	26.4569	27.714	28.7585	30.2162	31.6328	32.7939	33.8366	35.224	36.3662
25.253	26.4569	27.714	28.7585	30.2162	31.6328	32.7939	33.8366	35.224	36.3645
25.253	26.4741	27.7175	28.731	30.2008	31.5951	32.8179	33.8143	35.2069	36.3645
25.253	26.4741	27.7175	28.731	30.2008	31.5951	32.8179	33.8143	35.2069	36.3302
25.253	26.4724	27.7106	28.7533	30.2059	31.6019	32.8024	33.8366	35.2	36.3679
25.253	26.4792	27.6917	28.7722	30.2317	31.6259	32.8299	33.8657	35.2154	36.3525
25.253	26.4792	27.6917	28.7722	30.2317	31.6259	32.8299	33.8657	35.2154	36.3542
25.253	26.4689	27.7106	28.7602	30.2299	31.6431	32.8127	33.8692	35.2206	36.3696
25.2444	26.4878	27.702	28.779	30.2419	31.6174	32.8333	33.852	35.2137	36.3731
25.2444	26.4878	27.702	28.779	30.2419	31.6174	32.8333	33.852	35.2137	36.3285
25.2461	26.4724	27.6746	28.7773	30.2042	31.6414	32.8179	33.8726	35.212	36.3576
25.2393	26.4689	27.6832	28.7979	30.2008	31.6071	32.8093	33.8623	35.2326	36.3422
25.2393	26.4689	27.6832	28.7979	30.2008	31.6071	32.8093	33.8623	35.2326	36.3576
25.241	26.4621	27.7089	28.7722	30.2677	31.6311	32.823	33.8486	35.2566	36.3731
25.241	26.4638	27.6969	28.7585	30.2711	31.6191	32.8162	33.8452	35.2292	36.3748
25.241	26.4638	27.6969	28.7585	30.2711	31.6191	32.8162	33.8452	35.2292	36.3542
25.2667	26.4741	27.6917	28.7447	30.2214	31.5934	32.7836	33.8297	35.2309	36.3748
25.2667	26.4741	27.6917	28.7447	30.2214	31.5934	32.7836	33.8297	35.2309	36.3371
25.2616	26.4655	27.7072	28.7516	30.2334	31.5814	32.7767	33.8126	35.2549	36.3525
25.2582	26.4707	27.7003	28.7447	30.2179	31.6002	32.7939	33.8143	35.2652	36.3491
25.2582	26.4707	27.7003	28.7447	30.2179	31.6002	32.7939	33.8143	35.2652	36.3353
25.2736	26.4604	27.7003	28.7207	30.2128	31.5848	32.7458	33.8211	35.2635	36.3199
25.265	26.4415	27.7037	28.7173	30.1939	31.5471	32.7407	33.8572	35.1914	36.3165
25.265	26.4415	27.7037	28.7173	30.1939	31.5471	32.7407	33.8572	35.1914	36.325
25.2547	26.4381	27.6917	28.707	30.1888	31.5865	32.7681	33.8314	35.2172	36.3491
25.2633	26.4518	27.6832	28.7053	30.1956	31.6019	32.7664	33.852	35.2429	36.3491
25.2633	26.4518	27.6832	28.7053	30.1956	31.6019	32.7664	33.852	35.2429	36.3422
25.2736	26.4621	27.6797	28.7293	30.2008	31.5539	32.7904	33.7971	35.2429	36.3816
25.2736	26.4621	27.6797	28.7293	30.2008	31.5539	32.7904	33.7971	35.2429	36.3576
25.2667	26.4707	27.702	28.731	30.2145	31.5573	32.7767	33.8177	35.2669	36.3988
25.2787	26.4518	27.7123	28.7499	30.2059	31.5985	32.775	33.8297	35.2652	36.3765
25.2787	26.4518	27.7123	28.7499	30.2059	31.5985	32.775	33.8297	35.2652	36.3868
25.2822	26.4638	27.7003	28.7568	30.2008	31.6088	32.8179	33.8332	35.2909	36.3508
25.2856	26.4827	27.702	28.7533	30.2351	31.5814	32.7819	33.8246	35.2806	36.3645
25.2856	26.4827	27.702	28.7533	30.2351	31.5814	32.7819	33.8246	35.2806	36.3851
25.2907	26.4707	27.702	28.755	30.2351	31.6157	32.7819	33.8263	35.2652	36.3954
25.2839	26.4638	27.7072	28.7602	30.2522	31.5899	32.7836	33.7903	35.284	36.3988
25.2839	26.4638	27.7072	28.7602	30.2522	31.5899	32.7836	33.7903	35.284	36.3919
25.2804	26.4569	27.7055	28.7396	30.2745	31.5916	32.7767	33.8229	35.26	36.3714

25.277	26.4569	27.7055	28.7104	30.2728	31.6002	32.7596	33.8229	35.2669	36.4194
25.277	26.4569	27.7055	28.7104	30.2728	31.6002	32.7596	33.8229	35.2669	36.4348
25.2804	26.4278	27.7295	28.743	30.2111	31.6071	32.7664	33.8452	35.284	36.4262
25.2804	26.4278	27.7295	28.743	30.2111	31.6071	32.7664	33.8452	35.284	36.4159
25.277	26.4261	27.738	28.7705	30.2008	31.6002	32.7784	33.8314	35.2875	36.4674
25.2633	26.4449	27.7089	28.7602	30.2694	31.5882	32.7544	33.8126	35.248	36.4382
25.2633	26.4449	27.7089	28.7602	30.2694	31.5882	32.7544	33.8126	35.248	36.4468
25.2822	26.4398	27.6969	28.7499	30.2145	31.5985	32.7801	33.8657	35.2154	36.4314
25.289	26.4552	27.7106	28.7465	30.1922	31.6019	32.7664	33.8537	35.2858	36.4074
25.289	26.4552	27.7106	28.7465	30.1922	31.6019	32.7664	33.8537	35.2858	36.3748
25.3027	26.4261	27.714	28.7379	30.1699	31.6225	32.7493	33.8537	35.2669	36.3834
25.3096	26.4295	27.7055	28.743	30.1819	31.6551	32.7527	33.84	35.2257	36.3868
25.3096	26.4295	27.7055	28.743	30.1819	31.6551	32.7527	33.84	35.2257	36.3782
25.3199	26.4398	27.6935	28.7602	30.2042	31.6225	32.799	33.888	35.2257	36.3919
25.3199	26.4398	27.6935	28.7602	30.2042	31.6225	32.799	33.888	35.2257	36.3525
25.3302	26.4415	27.6643	28.7585	30.2179	31.6294	32.7784	33.8812	35.272	36.3799
25.3439	26.4278	27.6454	28.7568	30.2076	31.6345	32.8059	33.8795	35.2617	36.4005
25.3439	26.4278	27.6454	28.7568	30.2076	31.6345	32.8059	33.8795	35.2617	36.4245
25.3525	26.4329	27.6386	28.7499	30.2008	31.6139	32.7767	33.8829	35.2463	36.4314
25.3611	26.4106	27.6523	28.7602	30.2282	31.5934	32.8282	33.8572	35.2755	36.4228
25.3611	26.4106	27.6523	28.7602	30.2282	31.5934	32.8282	33.8572	35.2755	36.4194
25.3799	26.4278	27.6266	28.7139	30.2368	31.5848	32.7973	33.852	35.2463	36.4177
25.3885	26.4295	27.642	28.7225	30.2334	31.5728	32.7544	33.864	35.2429	36.4194
25.3885	26.4295	27.642	28.7225	30.2334	31.5728	32.7544	33.864	35.2429	36.4108
25.3851	26.4192	27.666	28.7242	30.2539	31.5488	32.787	33.8554	35.2532	36.4057
25.3799	26.4278	27.6609	28.719	30.2454	31.5814	32.7647	33.8589	35.2755	36.4279
25.3799	26.4278	27.6609	28.719	30.2454	31.5814	32.7647	33.8589	35.2755	36.4314
25.3868	26.4261	27.6386	28.743	30.2488	31.5659	32.7887	33.8983	35.2652	36.4622
25.3868	26.4261	27.6386	28.743	30.2488	31.5659	32.7887	33.8983	35.2652	36.4125
25.3833	26.4106	27.6369	28.7207	30.2076	31.5796	32.7921	33.8863	35.2669	36.3851
25.3782	26.4381	27.6214	28.7396	30.2282	31.5728	32.8042	33.8863	35.2635	36.3971
25.3782	26.4381	27.6214	28.7396	30.2282	31.5728	32.8042	33.8863	35.2635	36.4057
25.3851	26.4278	27.642	28.7516	30.2042	31.5951	32.8162	33.8863	35.2429	36.4022
25.3816	26.4226	27.6369	28.7482	30.1922	31.5711	32.8213	33.8623	35.2703	36.3971
25.3816	26.4226	27.6369	28.7482	30.1922	31.5711	32.8213	33.8623	35.2703	36.3936
25.3679	26.4038	27.6249	28.7362	30.1991	31.6002	32.8264	33.8589	35.2206	36.4108
25.3559	26.3866	27.606	28.7465	30.2196	31.6002	32.8179	33.8486	35.2395	36.4108
25.3559	26.3866	27.606	28.7465	30.2196	31.6002	32.8179	33.8486	35.2395	36.4125
25.349	26.4072	27.6369	28.7825	30.1956	31.5779	32.799	33.8486	35.2549	36.4468
25.349	26.4072	27.6369	28.7825	30.1956	31.5779	32.799	33.8486	35.2549	36.4177
25.3508	26.4123	27.6369	28.7688	30.2076	31.5676	32.8299	33.864	35.2532	36.4622
25.3456	26.4244	27.6249	28.7499	30.2299	31.5436	32.8162	33.8469	35.2755	36.4365
25.3456	26.4244	27.6249	28.7499	30.2299	31.5436	32.8162	33.8469	35.2755	36.4314

25.337	26.4364	27.6094	28.7533	30.2368	31.5779	32.8213	33.8434	35.2532	36.4417
25.3336	26.4398	27.594	28.755	30.2591	31.5865	32.7853	33.8554	35.2617	36.4142
25.3336	26.4398	27.594	28.755	30.2591	31.5865	32.7853	33.8554	35.2617	36.4262
25.3285	26.4158	27.6077	28.7619	30.2488	31.5694	32.7784	33.8572	35.2446	36.4297
25.325	26.4123	27.6197	28.7533	30.2265	31.5728	32.7561	33.8606	35.2463	36.4245
25.325	26.4123	27.6197	28.7533	30.2265	31.5728	32.7561	33.8606	35.2463	36.44
25.313	26.4432	27.6334	28.7259	30.2625	31.5899	32.7544	33.8572	35.2463	36.4159
25.3045	26.4398	27.6249	28.7259	30.278	31.6122	32.7939	33.8486	35.2274	36.4245
25.3045	26.4398	27.6249	28.7259	30.278	31.6122	32.7939	33.8486	35.2274	36.4159
25.325	26.4552	27.6369	28.7156	30.2625	31.5951	32.7904	33.8897	35.1949	36.3971
25.325	26.4552	27.6369	28.7156	30.2625	31.5951	32.7904	33.8897	35.1949	36.3919
25.3216	26.4449	27.6369	28.7276	30.2402	31.5951	32.751	33.8897	35.1949	36.4485
25.3302	26.4552	27.6643	28.7362	30.2814	31.5951	32.7801	33.876	35.2017	36.4125
25.3302	26.4552	27.6643	28.7362	30.2814	31.5951	32.7801	33.876	35.2017	36.4211
25.3268	26.4569	27.6454	28.7259	30.2128	31.6019	32.7767	33.8366	35.2515	36.4571
25.3268	26.4655	27.6489	28.7259	30.1991	31.6448	32.7716	33.8314	35.2292	36.4468
25.3268	26.4655	27.6489	28.7259	30.1991	31.6448	32.7716	33.8314	35.2292	36.4708
25.3165	26.4621	27.6592	28.7465	30.2111	31.6654	32.763	33.816	35.224	36.4674
25.325	26.4638	27.6677	28.7465	30.2179	31.6551	32.787	33.852	35.2446	36.4811
25.325	26.4638	27.6677	28.7465	30.2179	31.6551	32.787	33.852	35.2446	36.4708
25.3165	26.493	27.6746	28.755	30.2196	31.6139	32.7784	33.8623	35.2412	36.4434
25.3165	26.493	27.6746	28.755	30.2196	31.6139	32.7784	33.8623	35.2412	36.4605
25.3216	26.4809	27.666	28.7705	30.1974	31.6345	32.7716	33.8469	35.2086	36.4262
25.3319	26.4827	27.6523	28.7945	30.2128	31.6448	32.7578	33.864	35.2069	36.4057
25.3319	26.4827	27.6523	28.7945	30.2128	31.6448	32.7578	33.864	35.2069	36.4451
25.337	26.4827	27.6454	28.7996	30.2248	31.6088	32.7647	33.8657	35.2172	36.4005
25.3388	26.4655	27.666	28.7825	30.1956	31.5779	32.7716	33.8709	35.2223	36.3988
25.3388	26.4655	27.666	28.7825	30.1956	31.5779	32.7716	33.8709	35.2223	36.3902
25.3662	26.4587	27.6523	28.7636	30.1956	31.6002	32.8007	33.8554	35.1966	36.3885
25.3731	26.4655	27.6454	28.7825	30.2008	31.6225	32.7904	33.8572	35.2052	36.4177
25.3731	26.4655	27.6454	28.7825	30.2008	31.6225	32.7904	33.8572	35.2052	36.4331
25.3816	26.4569	27.6351	28.767	30.2111	31.6105	32.799	33.8023	35.248	36.4468
25.3885	26.4638	27.6454	28.7602	30.1802	31.6122	32.7836	33.8366	35.2172	36.4485
25.3885	26.4638	27.6454	28.7602	30.1802	31.6122	32.7836	33.8366	35.2172	36.4468
25.3919	26.4689	27.6592	28.7653	30.1682	31.6088	32.7939	33.8434	35.2343	36.44
25.3919	26.4689	27.6592	28.7653	30.1682	31.6088	32.7939	33.8434	35.2343	36.4468
25.4039	26.4827	27.6506	28.7499	30.1682	31.6002	32.7939	33.8452	35.236	36.4554
25.4056	26.4689	27.6437	28.7876	30.1956	31.6037	32.7716	33.8297	35.2772	36.4314
25.4056	26.4689	27.6437	28.7876	30.1956	31.6037	32.7716	33.8297	35.2772	36.4417
25.3988	26.4587	27.6506	28.7842	30.1991	31.5848	32.7853	33.8452	35.2515	36.4845
25.4074	26.4621	27.6454	28.7842	30.1682	31.5951	32.7904	33.8503	35.272	36.4811
25.4074	26.4621	27.6454	28.7842	30.1682	31.5951	32.7904	33.8503	35.272	36.4965
25.4108	26.4295	27.6832	28.7842	30.1596	31.5985	32.7733	33.8897	35.2755	36.4657

25.4125	26.4484	27.6643	28.8048	30.1751	31.5934	32.7819	33.8812	35.284	36.4965
25.4125	26.4484	27.6643	28.8048	30.1751	31.5934	32.7819	33.8812	35.284	36.4657
25.4091	26.4484	27.6694	28.7756	30.1751	31.5676	32.8127	33.9018	35.2583	36.4725
25.4091	26.4484	27.6694	28.7756	30.1751	31.5676	32.8127	33.9018	35.2583	36.4897
25.4159	26.4604	27.666	28.8031	30.1665	31.5676	32.8127	33.864	35.2412	36.5154
25.4091	26.4707	27.6746	28.8099	30.1871	31.5831	32.7956	33.8657	35.1931	36.4743
25.4091	26.4707	27.6746	28.8099	30.1871	31.5831	32.7956	33.8657	35.1931	36.4657
25.4056	26.4672	27.6712	28.7842	30.2025	31.5865	32.8076	33.8383	35.2103	36.4657
25.3782	26.4655	27.6935	28.755	30.2008	31.5985	32.8196	33.852	35.1674	36.4502
25.3782	26.4655	27.6935	28.755	30.2008	31.5985	32.8196	33.852	35.1674	36.4622
25.3833	26.4466	27.6626	28.7516	30.2008	31.5779	32.8093	33.8452	35.1931	36.4571
25.3816	26.4604	27.6403	28.755	30.2076	31.5762	32.7973	33.8229	35.2343	36.4468
25.3816	26.4604	27.6403	28.755	30.2076	31.5762	32.7973	33.8229	35.2343	36.4451
25.3576	26.4707	27.6454	28.7808	30.1888	31.6054	32.8522	33.8623	35.2497	36.464
25.3473	26.4587	27.666	28.7773	30.2128	31.6174	32.8247	33.8211	35.2223	36.4588
25.3473	26.4587	27.666	28.7773	30.2128	31.6174	32.8247	33.8211	35.2223	36.4777
25.3542	26.4569	27.6832	28.7842	30.2248	31.6328	32.8385	33.8229	35.2052	36.476
25.3542	26.4569	27.6832	28.7842	30.2248	31.6328	32.8385	33.8229	35.2052	36.5034
25.3628	26.4552	27.6832	28.7996	30.1888	31.6174	32.8385	33.8246	35.2069	36.476
25.3628	26.4501	27.7037	28.7928	30.1339	31.6088	32.8093	33.816	35.1811	36.4845
25.3628	26.4501	27.7037	28.7928	30.1339	31.6088	32.8093	33.816	35.1811	36.476
25.3611	26.4827	27.7072	28.7568	30.1613	31.6002	32.8144	33.8074	35.1674	36.4777
25.3508	26.4792	27.7055	28.7705	30.1785	31.6037	32.8059	33.8417	35.1451	36.4674
25.3508	26.4792	27.7055	28.7705	30.1785	31.6037	32.8059	33.8417	35.1451	36.4622
25.3439	26.4827	27.6866	28.7636	30.1888	31.5951	32.8059	33.8623	35.1571	36.4725
25.349	26.4672	27.678	28.7688	30.1974	31.6019	32.8316	33.8709	35.1143	36.4434
25.349	26.4672	27.678	28.7688	30.1974	31.6019	32.8316	33.8709	35.1143	36.4417
25.3576	26.4449	27.6746	28.7962	30.1974	31.6328	32.8282	33.8469	35.1571	36.4537
25.3576	26.4449	27.6746	28.7962	30.1974	31.6328	32.8282	33.8469	35.1571	36.4828
25.3611	26.4312	27.6694	28.7996	30.2196	31.6465	32.7939	33.8469	35.1674	36.4725
25.3851	26.4466	27.6557	28.7653	30.1939	31.6191	32.8247	33.864	35.2	36.4948
25.3851	26.4466	27.6557	28.7653	30.1939	31.6191	32.8247	33.864	35.2	36.4914
25.3782	26.4501	27.6609	28.8048	30.1733	31.6002	32.8385	33.8486	35.1863	36.4777
25.3885	26.4484	27.63	28.7825	30.1785	31.6362	32.823	33.864	35.1914	36.4743
25.3885	26.4484	27.63	28.7825	30.1785	31.6362	32.823	33.864	35.1914	36.4691
25.3988	26.4415	27.6746	28.7773	30.2059	31.6362	32.8144	33.8726	35.1949	36.4897
25.4176	26.4432	27.6935	28.7756	30.2094	31.5882	32.8316	33.8554	35.1966	36.5051
25.4176	26.4432	27.6935	28.7756	30.2094	31.5882	32.8316	33.8554	35.1966	36.5034
25.4211	26.4398	27.6849	28.7636	30.2042	31.6071	32.8367	33.8366	35.1897	36.5446
25.4245	26.4501	27.6952	28.7893	30.2248	31.6105	32.8436	33.8606	35.1794	36.5446
25.4245	26.4501	27.6952	28.7893	30.2248	31.6105	32.8436	33.8606	35.1794	36.5274
25.4348	26.4587	27.6746	28.779	30.2265	31.5676	32.8402	33.8726	35.128	36.5429
25.4348	26.4587	27.6746	28.779	30.2265	31.5676	32.8402	33.8726	35.128	36.4794

25.4365	26.4569	27.6729	28.7756	30.2591	31.5711	32.8625	33.8349	35.1606	36.5086
25.4297	26.4535	27.6609	28.7928	30.2282	31.5916	32.8402	33.8434	35.1503	36.4965
25.4297	26.4535	27.6609	28.7928	30.2282	31.5916	32.8402	33.8434	35.1503	36.4794
25.4382	26.4621	27.6849	28.7516	30.2025	31.6088	32.8419	33.8469	35.1554	36.5034
25.4262	26.4518	27.69	28.7327	30.1785	31.6105	32.8402	33.8349	35.1588	36.5017
25.4262	26.4518	27.69	28.7327	30.1785	31.6105	32.8402	33.8349	35.1588	36.4845
25.4382	26.4912	27.69	28.7327	30.1991	31.5916	32.8316	33.8366	35.1468	36.5274
25.4399	26.4775	27.69	28.743	30.2299	31.6037	32.8539	33.8023	35.1709	36.5188
25.4399	26.4775	27.69	28.743	30.2299	31.6037	32.8539	33.8023	35.1709	36.5291
25.4571	26.4655	27.6712	28.767	30.1853	31.5659	32.8453	33.8229	35.1537	36.5206
25.4571	26.4655	27.6712	28.767	30.1853	31.5659	32.8453	33.8229	35.1537	36.5
25.4691	26.4689	27.666	28.755	30.1853	31.5762	32.8333	33.8332	35.1829	36.4537
25.4571	26.4587	27.6746	28.743	30.1648	31.5985	32.8436	33.8486	35.1931	36.4828
25.4571	26.4587	27.6746	28.743	30.1648	31.5985	32.8436	33.8486	35.1931	36.5017
25.4588	26.4672	27.6454	28.7585	30.1768	31.6037	32.823	33.8589	35.1777	36.5103
25.4485	26.4501	27.678	28.7396	30.2094	31.5865	32.8419	33.8332	35.1434	36.4965
25.4485	26.4501	27.678	28.7396	30.2094	31.5865	32.8419	33.8332	35.1434	36.4965
25.4554	26.4432	27.6952	28.7345	30.2076	31.6311	32.8385	33.8349	35.14	36.4811
25.4657	26.4449	27.6883	28.7362	30.2094	31.6328	32.8264	33.8211	35.164	36.512
25.4657	26.4449	27.6883	28.7362	30.2094	31.6328	32.8264	33.8211	35.164	36.464
25.4451	26.4484	27.6386	28.7516	30.2214	31.6225	32.8385	33.828	35.164	36.5068
25.4468	26.4466	27.6506	28.7447	30.1819	31.6037	32.8264	33.7937	35.1623	36.4983
25.4468	26.4466	27.6506	28.7447	30.1819	31.6037	32.8264	33.7937	35.1623	36.4983
25.4571	26.4587	27.6592	28.7327	30.2179	31.6208	32.8556	33.8057	35.1674	36.4537
25.4571	26.4587	27.6592	28.7327	30.2179	31.6208	32.8556	33.8057	35.1674	36.4159
25.464	26.4518	27.6557	28.7482	30.2111	31.6208	32.8419	33.8452	35.1726	36.4297
25.476	26.4484	27.654	28.7276	30.2025	31.6242	32.8367	33.8314	35.152	36.4691
25.476	26.4484	27.654	28.7276	30.2025	31.6242	32.8367	33.8314	35.152	36.4777
25.4554	26.4312	27.6454	28.7482	30.2299	31.6311	32.8367	33.8246	35.1623	36.4502
25.4554	26.4381	27.6471	28.755	30.2248	31.6277	32.835	33.8469	35.1606	36.4743
25.4554	26.4381	27.6471	28.755	30.2248	31.6277	32.835	33.8469	35.1606	36.4554
25.4519	26.4381	27.6643	28.7242	30.2162	31.6311	32.8367	33.84	35.1897	36.4691
25.4434	26.4244	27.6506	28.719	30.2471	31.6397	32.8196	33.852	35.1966	36.4588
25.4434	26.4244	27.6506	28.719	30.2471	31.6397	32.8196	33.852	35.1966	36.4794
25.4331	26.4158	27.6712	28.7379	30.2522	31.6362	32.8162	33.8211	35.1743	36.4811
25.4331	26.4158	27.6712	28.7379	30.2522	31.6362	32.8162	33.8211	35.1743	36.4502
25.4417	26.4158	27.6592	28.7619	30.2625	31.6277	32.8179	33.8366	35.1829	36.476
25.4485	26.4072	27.6489	28.755	30.2642	31.6328	32.8162	33.8383	35.1554	36.4382
25.4485	26.4072	27.6489	28.755	30.2642	31.6328	32.8162	33.8383	35.1554	36.4845
25.4468	26.4089	27.6643	28.7345	30.2625	31.6294	32.8059	33.8589	35.1451	36.4297
25.4519	26.4192	27.6609	28.7362	30.2694	31.5899	32.7921	33.828	35.1863	36.4022
25.4519	26.4192	27.6609	28.7362	30.2694	31.5899	32.7921	33.828	35.1863	36.4159
25.4605	26.4106	27.654	28.7327	30.2539	31.6294	32.8093	33.7851	35.188	36.4297

25.4605	26.4158	27.6729	28.7104	30.2557	31.6225	32.8213	33.8194	35.2257	36.4279
25.4605	26.4158	27.6729	28.7104	30.2557	31.6225	32.8213	33.8194	35.2257	36.4177
25.4725	26.4106	27.666	28.7276	30.2522	31.5831	32.8299	33.8091	35.1966	36.452
25.4777	26.4141	27.6643	28.7396	30.2934	31.5728	32.8436	33.7697	35.1931	36.4039
25.4777	26.4141	27.6643	28.7396	30.2934	31.5728	32.8436	33.7697	35.1931	36.4005
25.4862	26.4209	27.6832	28.779	30.2437	31.6122	32.8282	33.792	35.1931	36.4022
25.4862	26.4209	27.6832	28.779	30.2437	31.6122	32.8282	33.792	35.1931	36.4262
25.488	26.4226	27.6797	28.7413	30.2265	31.6397	32.8179	33.7989	35.2052	36.4177
25.4828	26.4021	27.6334	28.7499	30.2437	31.6105	32.811	33.792	35.2086	36.4194
25.4828	26.4021	27.6334	28.7499	30.2437	31.6105	32.811	33.792	35.2086	36.3971
25.4742	26.3986	27.6471	28.7688	30.2231	31.5985	32.8024	33.828	35.1623	36.3868
25.4622	26.4003	27.6386	28.7499	30.2317	31.6054	32.7921	33.8177	35.1829	36.3765
25.4622	26.4003	27.6386	28.7499	30.2317	31.6054	32.7921	33.8177	35.1829	36.3679
25.464	26.4158	27.6523	28.7327	30.2437	31.6294	32.7767	33.7937	35.1623	36.3902
25.4571	26.4141	27.6643	28.7293	30.2248	31.6259	32.7767	33.78	35.1743	36.3731
25.4571	26.4141	27.6643	28.7293	30.2248	31.6259	32.7767	33.78	35.1743	36.3542
25.4502	26.4226	27.6746	28.7242	30.2402	31.6431	32.751	33.8263	35.1674	36.3902
25.4502	26.4226	27.6746	28.7242	30.2402	31.6431	32.751	33.8263	35.1674	36.3885
25.4519	26.4123	27.6592	28.7362	30.2642	31.6534	32.7819	33.8246	35.1674	36.3645
25.4502	26.4329	27.63	28.7756	30.2762	31.6517	32.7716	33.7989	35.2154	36.3628
25.4502	26.4329	27.63	28.7756	30.2762	31.6517	32.7716	33.7989	35.2154	36.3542
25.4468	26.4466	27.6454	28.767	30.2642	31.638	32.763	33.8109	35.1949	36.3765
25.4331	26.4106	27.6489	28.7499	30.2728	31.5985	32.7664	33.8211	35.1846	36.3782
25.4331	26.4106	27.6489	28.7499	30.2728	31.5985	32.7664	33.8211	35.1846	36.4091
25.4228	26.4261	27.6677	28.755	30.2625	31.6242	32.7819	33.8109	35.176	36.3885
25.4176	26.4278	27.6557	28.7602	30.2797	31.6191	32.7853	33.804	35.1606	36.3868
25.4176	26.4278	27.6557	28.7602	30.2797	31.6191	32.7853	33.804	35.1606	36.4057
25.4091	26.4141	27.6369	28.7533	30.2694	31.6037	32.8076	33.8297	35.1417	36.4228
25.4125	26.4089	27.642	28.7825	30.2934	31.6174	32.7921	33.8177	35.1434	36.4108
25.4125	26.4089	27.642	28.7825	30.2934	31.6174	32.7921	33.8177	35.1434	36.3593
25.3971	26.4175	27.6334	28.7756	30.266	31.6225	32.8385	33.828	35.1588	36.3799
25.3971	26.4175	27.6334	28.7756	30.266	31.6225	32.8385	33.828	35.1588	36.3662
25.3971	26.4278	27.6266	28.755	30.2694	31.6225	32.8127	33.8006	35.1537	36.4297
25.4091	26.4261	27.642	28.779	30.2248	31.5659	32.7819	33.8126	35.1709	36.4142
25.4091	26.4261	27.642	28.779	30.2248	31.5659	32.7819	33.8126	35.1709	36.3645
25.4039	26.4261	27.63	28.7396	30.2865	31.5951	32.7716	33.8091	35.2	36.3782
25.4005	26.4192	27.6369	28.7705	30.2522	31.5968	32.7853	33.7954	35.2069	36.3816
25.4005	26.4192	27.6369	28.7705	30.2522	31.5968	32.7853	33.7954	35.2069	36.3491
25.3954	26.4329	27.6506	28.7739	30.2608	31.6002	32.7664	33.7954	35.2034	36.3559
25.3988	26.4329	27.6592	28.7688	30.2351	31.6037	32.7853	33.7971	35.2052	36.3696
25.3988	26.4329	27.6592	28.7688	30.2351	31.6037	32.7853	33.7971	35.2052	36.3782
25.3799	26.4346	27.6729	28.7242	30.2591	31.6054	32.7853	33.8143	35.212	36.3456
25.3799	26.4346	27.6729	28.7242	30.2591	31.6054	32.7853	33.8143	35.212	36.3782

25.3799	26.4278	27.678	28.7087	30.2471	31.5745	32.7836	33.8023	35.1777	36.3371
25.3559	26.4364	27.6626	28.7327	30.2728	31.6157	32.7647	33.828	35.2103	36.3508
25.3559	26.4364	27.6626	28.7327	30.2728	31.6157	32.7647	33.828	35.2103	36.3508
25.3611	26.4381	27.642	28.7533	30.2591	31.6328	32.7956	33.8452	35.1794	36.3353
25.3439	26.4244	27.6626	28.6967	30.3003	31.6071	32.7853	33.8177	35.1983	36.3491
25.3439	26.4244	27.6626	28.6967	30.3003	31.6071	32.7853	33.8177	35.1983	36.3611
25.3473	26.4329	27.6832	28.7345	30.2882	31.6242	32.7699	33.816	35.2	36.3336
25.3456	26.4518	27.6849	28.7225	30.2694	31.6019	32.7647	33.8349	35.1966	36.3268
25.3456	26.4518	27.6849	28.7225	30.2694	31.6019	32.7647	33.8349	35.1966	36.3233
25.3388	26.4569	27.6969	28.7447	30.2505	31.5934	32.7613	33.8126	35.2292	36.301
25.3353	26.4569	27.6849	28.7653	30.2831	31.5934	32.7733	33.8246	35.1691	36.2993
25.3353	26.4569	27.6849	28.7653	30.2831	31.5934	32.7733	33.8246	35.1691	36.2753
25.3302	26.4672	27.69	28.7379	30.2625	31.6139	32.7853	33.8057	35.176	36.2719
25.3302	26.4672	27.69	28.7379	30.2625	31.6139	32.7853	33.8057	35.176	36.2685
25.3147	26.4724	27.7003	28.7465	30.2814	31.6191	32.7819	33.8246	35.1726	36.2667
25.3199	26.4844	27.6883	28.7773	30.2437	31.6242	32.7784	33.8572	35.1657	36.2959
25.3199	26.4844	27.6883	28.7773	30.2437	31.6242	32.7784	33.8572	35.1657	36.2907
25.3113	26.4792	27.6935	28.7499	30.2351	31.6345	32.7921	33.8572	35.1468	36.3422
25.3147	26.4741	27.6797	28.7379	30.2094	31.6208	32.7801	33.852	35.152	36.289
25.3147	26.4741	27.6797	28.7379	30.2094	31.6208	32.7801	33.852	35.152	36.3268
25.301	26.4655	27.6969	28.7465	30.2454	31.6088	32.7647	33.852	35.1366	36.3422
25.2993	26.4672	27.69	28.7602	30.2488	31.5951	32.7699	33.8623	35.1383	36.3353
25.2993	26.4672	27.69	28.7602	30.2488	31.5951	32.7699	33.8623	35.1383	36.3714
25.2959	26.4552	27.6832	28.7602	30.2437	31.5882	32.7836	33.8606	35.1228	36.3336
25.2959	26.4552	27.6832	28.7602	30.2437	31.5882	32.7836	33.8606	35.1228	36.3508
25.2942	26.4449	27.6797	28.767	30.266	31.5985	32.7647	33.8434	35.1623	36.3439
25.2804	26.4621	27.6729	28.755	30.2488	31.5865	32.7733	33.8709	35.1709	36.3525
25.2804	26.4621	27.6729	28.755	30.2488	31.5865	32.7733	33.8709	35.1709	36.3422
25.2856	26.4484	27.6677	28.7482	30.2831	31.5968	32.7613	33.828	35.152	36.3576
25.2873	26.4398	27.6592	28.7516	30.2677	31.5934	32.7613	33.8297	35.1777	36.3473
25.2873	26.4398	27.6592	28.7516	30.2677	31.5934	32.7613	33.8297	35.1777	36.3576
25.2839	26.4552	27.6814	28.7465	30.2145	31.5796	32.7578	33.8554	35.1691	36.3542
25.277	26.4518	27.6797	28.755	30.2059	31.5556	32.7596	33.8452	35.1657	36.3714
25.277	26.4518	27.6797	28.755	30.2059	31.5556	32.7596	33.8452	35.1657	36.3885
25.2702	26.4466	27.6712	28.7705	30.2385	31.5985	32.7304	33.8692	35.1846	36.3302
25.265	26.4552	27.6694	28.7842	30.2419	31.5934	32.7047	33.8623	35.1691	36.3388
25.265	26.4552	27.6694	28.7842	30.2419	31.5934	32.7047	33.8623	35.1691	36.3216
25.265	26.4587	27.6729	28.7516	30.2334	31.5916	32.7253	33.816	35.212	36.301
25.265	26.4587	27.6729	28.7516	30.2334	31.5916	32.7253	33.816	35.212	36.3285
25.265	26.4587	27.678	28.7465	30.2334	31.5865	32.7356	33.8177	35.1811	36.3542
25.265	26.4792	27.6814	28.779	30.1956	31.6054	32.7664	33.8074	35.1897	36.3371
25.265	26.4792	27.6814	28.779	30.1956	31.6054	32.7664	33.8074	35.1897	36.3662
25.2444	26.4587	27.6969	28.767	30.2248	31.6071	32.7716	33.8229	35.1571	36.3268

25.2359	26.4707	27.6866	28.7465	30.2248	31.6328	32.7647	33.7937	35.1623	36.3422
25.2359	26.4707	27.6866	28.7465	30.2248	31.6328	32.7647	33.7937	35.1623	36.3645
25.241	26.4621	27.654	28.7893	30.2231	31.5951	32.7733	33.8126	35.1383	36.3662
25.2444	26.4638	27.6746	28.7893	30.2402	31.6122	32.7819	33.78	35.164	36.4074
25.2444	26.4638	27.6746	28.7893	30.2402	31.6122	32.7819	33.78	35.164	36.4091
25.2479	26.4569	27.7175	28.7842	30.2042	31.6088	32.8007	33.8314	35.128	36.3885
25.2479	26.4569	27.7175	28.7842	30.2042	31.6088	32.8007	33.8314	35.128	36.3868
25.2479	26.4552	27.69	28.7876	30.2385	31.5779	32.7904	33.8383	35.1211	36.3954
25.2599	26.4398	27.6832	28.7945	30.2454	31.5934	32.7904	33.8177	35.1417	36.3902
25.2599	26.4398	27.6832	28.7945	30.2454	31.5934	32.7904	33.8177	35.1417	36.4022
25.2616	26.4381	27.6814	28.7911	30.2591	31.6208	32.811	33.816	35.1331	36.4125
25.2427	26.4329	27.6763	28.7911	30.2471	31.6019	32.7699	33.7937	35.1348	36.4022
25.2427	26.4329	27.6763	28.7911	30.2471	31.6019	32.7699	33.7937	35.1348	36.3816
25.2393	26.4295	27.6712	28.8442	30.2677	31.6071	32.7836	33.8057	35.0885	36.3954
25.2341	26.4449	27.654	28.8442	30.2814	31.6037	32.7887	33.7851	35.1297	36.4074
25.2341	26.4449	27.654	28.8442	30.2814	31.6037	32.7887	33.7851	35.1297	36.4279
25.2204	26.4346	27.6574	28.8271	30.2574	31.5985	32.7887	33.7937	35.1383	36.4057
25.217	26.4518	27.6403	28.8236	30.2557	31.6019	32.775	33.7937	35.14	36.3954
25.217	26.4518	27.6403	28.8236	30.2557	31.6019	32.775	33.7937	35.14	36.4177
25.2084	26.4638	27.6403	28.8065	30.2625	31.5694	32.7853	33.8109	35.1726	36.4108
25.2084	26.4638	27.6403	28.8065	30.2625	31.5694	32.7853	33.8109	35.1726	36.4228
25.2067	26.4604	27.6454	28.8374	30.2334	31.5762	32.7716	33.828	35.1726	36.4228
25.205	26.4655	27.6369	28.8013	30.2248	31.5779	32.8093	33.8503	35.1588	36.3902
25.205	26.4655	27.6369	28.8013	30.2248	31.5779	32.8093	33.8503	35.1588	36.4142
25.2153	26.4604	27.6197	28.8099	30.2076	31.5659	32.8024	33.8297	35.1743	36.3834
25.2067	26.4501	27.6197	28.8082	30.2076	31.5694	32.7887	33.8452	35.1194	36.3936
25.2067	26.4501	27.6197	28.8082	30.2076	31.5694	32.7887	33.8452	35.1194	36.3868
25.1981	26.4535	27.6437	28.8339	30.2111	31.5539	32.7921	33.8297	35.1005	36.4091
25.1964	26.4672	27.654	28.8151	30.2368	31.5711	32.8385	33.852	35.1211	36.3868
25.1964	26.4672	27.654	28.8151	30.2368	31.5711	32.8385	33.852	35.1211	36.3645
25.2033	26.4432	27.6626	28.8288	30.2214	31.5882	32.7956	33.8211	35.0937	36.3954
25.2033	26.4432	27.6626	28.8288	30.2214	31.5882	32.7956	33.8211	35.0937	36.4125
25.205	26.4518	27.6712	28.8031	30.2008	31.5848	32.799	33.816	35.116	36.4177
25.1981	26.4415	27.6832	28.7962	30.1768	31.6122	32.8059	33.8349	35.0937	36.3936
25.1981	26.4415	27.6832	28.7962	30.1768	31.6122	32.8059	33.8349	35.0937	36.3868
25.1964	26.4501	27.6935	28.7928	30.1802	31.5882	32.8453	33.8126	35.0971	36.4297
25.2101	26.4775	27.6746	28.8065	30.2351	31.5728	32.8093	33.8383	35.0851	36.3628
25.2101	26.4775	27.6746	28.8065	30.2351	31.5728	32.8093	33.8383	35.0851	36.4022
25.2118	26.4758	27.6489	28.8013	30.2059	31.6002	32.8299	33.8332	35.1366	36.4262
25.2187	26.4758	27.6557	28.8219	30.2196	31.5882	32.8247	33.8229	35.1194	36.4142
25.2187	26.4758	27.6557	28.8219	30.2196	31.5882	32.8247	33.8229	35.1194	36.4417
25.2341	26.4707	27.6952	28.7893	30.2248	31.5934	32.8247	33.828	35.1588	36.4417
25.2273	26.4724	27.6952	28.8013	30.2282	31.5659	32.8299	33.8589	35.1554	36.4845

25.2273	26.4724	27.6952	28.8013	30.2282	31.5659	32.8299	33.8589	35.1554	36.4777
25.2341	26.4689	27.6814	28.7876	30.2128	31.5814	32.8522	33.8143	35.1211	36.524
25.2341	26.4689	27.6814	28.7876	30.2128	31.5814	32.8522	33.8143	35.1211	36.4948
25.2324	26.4621	27.6746	28.7945	30.2162	31.5591	32.8059	33.8143	35.1177	36.5257
25.2273	26.4604	27.6729	28.8202	30.2179	31.5676	32.799	33.816	35.1263	36.5343
25.2273	26.4604	27.6729	28.8202	30.2179	31.5676	32.799	33.816	35.1263	36.5497
25.2221	26.4689	27.69	28.7859	30.2111	31.6019	32.8333	33.84	35.0902	36.4983
25.2221	26.4707	27.7037	28.7688	30.2042	31.5762	32.8419	33.852	35.0577	36.5034
25.2221	26.4707	27.7037	28.7688	30.2042	31.5762	32.8419	33.852	35.0577	36.5257
25.2239	26.4638	27.7123	28.7773	30.1802	31.5762	32.8573	33.8366	35.0765	36.5068
25.2187	26.4689	27.6866	28.7893	30.1579	31.5745	32.8264	33.84	35.0371	36.5257
25.2187	26.4689	27.6866	28.7893	30.1579	31.5745	32.8264	33.84	35.0371	36.5068
25.2239	26.4655	27.6866	28.7808	30.2025	31.5951	32.8505	33.8194	35.0714	36.512
25.2239	26.4655	27.6866	28.7808	30.2025	31.5951	32.8505	33.8194	35.0714	36.488
25.229	26.4535	27.69	28.7842	30.1991	31.5608	32.8316	33.8452	35.0371	36.5017
25.2136	26.4518	27.7003	28.8133	30.1648	31.5608	32.8453	33.8194	35.0525	36.5206
25.2136	26.4518	27.7003	28.8133	30.1648	31.5608	32.8453	33.8194	35.0525	36.5137
25.2239	26.4484	27.7123	28.8082	30.1699	31.5711	32.8487	33.8297	35.0508	36.4811
25.2187	26.4329	27.6883	28.779	30.2025	31.5659	32.8573	33.8486	35.0868	36.4897
25.2187	26.4329	27.6883	28.779	30.2025	31.5659	32.8573	33.8486	35.0868	36.4931
25.2204	26.4003	27.7037	28.8133	30.2214	31.5299	32.8607	33.8246	35.0422	36.4691
25.2033	26.3901	27.69	28.8202	30.2471	31.5059	32.8573	33.8211	35.068	36.4914
25.2033	26.3901	27.69	28.8202	30.2471	31.5059	32.8573	33.8211	35.068	36.5531
25.217	26.3901	27.7003	28.755	30.2385	31.5333	32.8607	33.8469	35.0182	36.5411
25.217	26.3832	27.702	28.7585	30.2214	31.5694	32.8607	33.8263	35.0542	36.5188
25.217	26.3832	27.702	28.7585	30.2214	31.5694	32.8607	33.8263	35.0542	36.5188
25.2118	26.3918	27.7192	28.7516	30.2042	31.5591	32.8848	33.8486	35.0594	36.56
25.2118	26.3918	27.7192	28.7516	30.2042	31.5591	32.8848	33.8486	35.0594	36.5377
25.2033	26.4106	27.7209	28.7808	30.1922	31.5625	32.871	33.8726	35.0714	36.5737
25.2084	26.4158	27.7089	28.7808	30.1922	31.5608	32.8659	33.8126	35.0971	36.5772
25.2084	26.4158	27.7089	28.7808	30.1922	31.5608	32.8659	33.8126	35.0971	36.5857
25.2204	26.4192	27.6986	28.7636	30.1768	31.5934	32.8693	33.8537	35.0834	36.5874
25.229	26.4346	27.6969	28.7825	30.2059	31.5796	32.8196	33.8452	35.0714	36.632
25.229	26.4346	27.6969	28.7825	30.2059	31.5796	32.8196	33.8452	35.0714	36.5874
25.2324	26.4346	27.7329	28.7773	30.1974	31.5865	32.847	33.8383	35.0988	36.5754
25.2376	26.4175	27.726	28.7739	30.1853	31.5985	32.8436	33.84	35.1211	36.6029
25.2376	26.4175	27.726	28.7739	30.1853	31.5985	32.8436	33.84	35.1211	36.608
25.2427	26.4364	27.726	28.7396	30.2299	31.6174	32.8247	33.8383	35.1348	36.5892
25.2427	26.4364	27.726	28.7396	30.2299	31.6174	32.8247	33.8383	35.1348	36.5549
25.2341	26.4312	27.7278	28.7516	30.2214	31.6054	32.8179	33.828	35.128	36.5823
25.2273	26.4055	27.7518	28.7688	30.2385	31.5745	32.7904	33.8263	35.1297	36.5634
25.2273	26.4055	27.7518	28.7688	30.2385	31.5745	32.7904	33.8263	35.1297	36.5514
25.2239	26.4141	27.7483	28.767	30.2094	31.5985	32.811	33.8349	35.1366	36.5686

25.2479	26.4158	27.7243	28.7842	30.2231	31.5676	32.8316	33.8126	35.0937	36.5291
25.2479	26.4158	27.7243	28.7842	30.2231	31.5676	32.8316	33.8126	35.0937	36.536
25.2393	26.4175	27.69	28.7756	30.2454	31.5694	32.8024	33.8074	35.0937	36.572
25.2427	26.4141	27.666	28.7585	30.2454	31.5728	32.8076	33.8023	35.0902	36.6046
25.2427	26.4141	27.666	28.7585	30.2454	31.5728	32.8076	33.8023	35.0902	36.5497
25.229	26.4226	27.6883	28.7773	30.2385	31.6037	32.7887	33.8177	35.1125	36.5737
25.2376	26.4209	27.6917	28.7859	30.2814	31.6328	32.8007	33.8057	35.0937	36.5772
25.2376	26.4209	27.6917	28.7859	30.2814	31.6328	32.8007	33.8057	35.0937	36.5394
25.2341	26.4123	27.6763	28.7842	30.2694	31.5694	32.7853	33.8349	35.0388	36.5686
25.2341	26.4123	27.6763	28.7842	30.2694	31.5694	32.7853	33.8349	35.0388	36.5463
25.2393	26.4192	27.6729	28.7825	30.2282	31.5779	32.8007	33.8434	35.0594	36.5926
25.2324	26.4209	27.6677	28.7962	30.2488	31.5985	32.8213	33.792	35.0422	36.5651
25.2324	26.4209	27.6677	28.7962	30.2488	31.5985	32.8213	33.792	35.0422	36.5892
25.2187	26.4123	27.6557	28.7619	30.2368	31.5831	32.8385	33.8057	35.0662	36.5789
25.2341	26.4123	27.6489	28.7825	30.2351	31.5642	32.8573	33.8143	35.0954	36.572
25.2341	26.4123	27.6489	28.7825	30.2351	31.5642	32.8573	33.8143	35.0954	36.584
25.2513	26.4226	27.6334	28.7962	30.1836	31.5402	32.8933	33.8194	35.1091	36.5909
25.2599	26.4278	27.6574	28.7911	30.2248	31.5573	32.871	33.8349	35.1143	36.596
25.2599	26.4278	27.6574	28.7911	30.2248	31.5573	32.871	33.8349	35.1143	36.608
25.2582	26.4415	27.6746	28.7893	30.2145	31.5471	32.8779	33.8486	35.0971	36.5857
25.2582	26.4415	27.6746	28.7893	30.2145	31.5471	32.8779	33.8486	35.0971	36.5686
25.2667	26.4381	27.6729	28.7585	30.2162	31.5676	32.9002	33.8383	35.0834	36.5994
25.2599	26.4415	27.6814	28.7568	30.2368	31.5573	32.8607	33.8229	35.0851	36.5892
25.2599	26.4415	27.6814	28.7568	30.2368	31.5573	32.8607	33.8229	35.0851	36.6012
25.2616	26.4432	27.6814	28.7756	30.2471	31.5728	32.8539	33.8109	35.0594	36.572
25.2753	26.4535	27.6917	28.7636	30.2368	31.5711	32.835	33.8332	35.1228	36.6115
25.2753	26.4535	27.6917	28.7636	30.2368	31.5711	32.835	33.8332	35.1228	36.6012
25.2719	26.4398	27.702	28.7756	30.2694	31.5899	32.8179	33.8177	35.1177	36.6166
25.2702	26.4432	27.6849	28.8151	30.2299	31.5934	32.8024	33.7903	35.1125	36.5874
25.2702	26.4432	27.6849	28.8151	30.2299	31.5934	32.8024	33.7903	35.1125	36.6217
25.2702	26.4209	27.6694	28.7911	30.2385	31.5676	32.8076	33.7903	35.0971	36.5909
25.2787	26.4278	27.6712	28.7773	30.2471	31.5642	32.8127	33.8006	35.0765	36.62
25.2787	26.4278	27.6712	28.7773	30.2471	31.5642	32.8127	33.8006	35.0765	36.5994
25.2822	26.4072	27.666	28.8048	30.2059	31.5882	32.8179	33.7697	35.0782	36.5806
25.2822	26.4072	27.666	28.8048	30.2059	31.5882	32.8179	33.7697	35.0782	36.6132
25.2839	26.4175	27.6489	28.8048	30.1991	31.5916	32.8042	33.7594	35.1245	36.6183
25.2873	26.4158	27.6506	28.7773	30.2059	31.5882	32.7973	33.7491	35.0988	36.5806
25.2873	26.4158	27.6506	28.7773	30.2059	31.5882	32.7973	33.7491	35.0988	36.5737
25.2925	26.4209	27.6935	28.7859	30.2196	31.5711	32.8179	33.7577	35.0868	36.5977
25.2856	26.4021	27.7055	28.7842	30.2265	31.5865	32.8059	33.7783	35.1125	36.5874
25.2856	26.4021	27.7055	28.7842	30.2265	31.5865	32.8059	33.7783	35.1125	36.5806
25.2942	26.4209	27.6797	28.7928	30.2076	31.5659	32.7733	33.7817	35.0628	36.572
25.289	26.4261	27.6523	28.7979	30.1716	31.6037	32.8179	33.7731	35.0594	36.5686

25.289	26.4261	27.6523	28.7979	30.1716	31.6037	32.8179	33.7731	35.0594	36.5857
25.2959	26.4089	27.666	28.7808	30.1853	31.6105	32.7956	33.8349	35.0319	36.5686
25.2959	26.4089	27.666	28.7808	30.1853	31.6105	32.7956	33.8349	35.0319	36.5892
25.3027	26.3986	27.6746	28.7722	30.1991	31.5951	32.7956	33.8023	35.0371	36.572
25.3096	26.3883	27.6746	28.7928	30.1751	31.6328	32.7973	33.8177	35.0885	36.5463
25.3096	26.3883	27.6746	28.7928	30.1751	31.6328	32.7973	33.8177	35.0885	36.5463
25.3113	26.3849	27.6592	28.779	30.1631	31.5985	32.8213	33.8177	35.1005	36.5977
25.3113	26.3815	27.6471	28.7808	30.1905	31.6019	32.7836	33.8297	35.1108	36.5274
25.3113	26.3815	27.6471	28.7808	30.1905	31.6019	32.7836	33.8297	35.1108	36.5772
25.3233	26.4055	27.6694	28.7688	30.2025	31.6071	32.7973	33.8143	35.0971	36.5634
25.313	26.3969	27.6849	28.7876	30.1853	31.6174	32.799	33.8194	35.1091	36.572
25.313	26.3969	27.6849	28.7876	30.1853	31.6174	32.799	33.8194	35.1091	36.5772
25.3062	26.3729	27.6729	28.7825	30.1785	31.6071	32.8247	33.8486	35.1108	36.5703
25.3113	26.3575	27.6797	28.7773	30.1836	31.6002	32.8127	33.8074	35.1331	36.5514
25.3113	26.3575	27.6797	28.7773	30.1836	31.6002	32.8127	33.8074	35.1331	36.5326
25.3045	26.3643	27.6729	28.7568	30.2025	31.6191	32.8316	33.8417	35.1263	36.512
25.3045	26.3643	27.6729	28.7568	30.2025	31.6191	32.8316	33.8417	35.1263	36.5429
25.3045	26.3678	27.6694	28.7533	30.1871	31.6277	32.8162	33.8606	35.0937	36.5223
25.301	26.366	27.6832	28.767	30.1819	31.6191	32.8402	33.8383	35.1366	36.5034
25.301	26.366	27.6832	28.767	30.1819	31.6191	32.8402	33.8383	35.1366	36.5068
25.313	26.3472	27.69	28.7688	30.1716	31.6157	32.8196	33.8469	35.0971	36.5377
25.3182	26.3472	27.6763	28.7465	30.1853	31.5814	32.823	33.84	35.1211	36.5206
25.3182	26.3472	27.6763	28.7465	30.1853	31.5814	32.823	33.84	35.1211	36.4931
25.3439	26.3575	27.7003	28.7379	30.2025	31.5916	32.7904	33.8246	35.1228	36.5274
25.3542	26.3678	27.6952	28.7413	30.2145	31.6071	32.7699	33.8126	35.1177	36.5206
25.3542	26.3678	27.6952	28.7413	30.2145	31.6071	32.7699	33.8126	35.1177	36.5
25.3679	26.3901	27.6849	28.7465	30.1768	31.5831	32.7664	33.7766	35.1314	36.4897
25.3679	26.3901	27.6849	28.7465	30.1768	31.5831	32.7664	33.7766	35.1314	36.4948
25.3679	26.4089	27.6797	28.7447	30.1888	31.5985	32.799	33.7851	35.1297	36.4743
25.3713	26.4226	27.6797	28.7516	30.2334	31.5728	32.8076	33.8109	35.0954	36.4657
25.3713	26.4226	27.6797	28.7516	30.2334	31.5728	32.8076	33.8109	35.0954	36.476
25.3696	26.4209	27.6832	28.7465	30.2145	31.5711	32.8402	33.792	35.1005	36.5326
25.3799	26.3952	27.6609	28.7362	30.2196	31.6054	32.8196	33.8246	35.092	36.4965
25.3799	26.3952	27.6609	28.7362	30.2196	31.6054	32.8196	33.8246	35.092	36.5446
25.3782	26.3935	27.6712	28.7122	30.2265	31.5968	32.8024	33.8246	35.08	36.5103
25.3833	26.3883	27.6574	28.7173	30.2248	31.5951	32.8093	33.8503	35.1023	36.5223
25.3833	26.3883	27.6574	28.7173	30.2248	31.5951	32.8093	33.8503	35.1023	36.4777
25.4022	26.3935	27.666	28.7516	30.2128	31.6054	32.8247	33.8074	35.1383	36.5
25.4022	26.4106	27.6729	28.7602	30.2059	31.6105	32.8299	33.8109	35.1794	36.4554
25.4022	26.4106	27.6729	28.7602	30.2059	31.6105	32.8299	33.8109	35.1794	36.4777
25.4056	26.4141	27.6626	28.7739	30.1922	31.5934	32.8007	33.8177	35.1554	36.4348
25.4056	26.4141	27.6626	28.7739	30.1922	31.5934	32.8007	33.8177	35.1554	36.4622
25.4074	26.4261	27.6643	28.7568	30.2059	31.5676	32.8127	33.8091	35.1331	36.4262

25.3988	26.4261	27.6797	28.7465	30.2179	31.5659	32.811	33.8006	35.1383	36.4382
25.3988	26.4261	27.6797	28.7465	30.2179	31.5659	32.811	33.8006	35.1383	36.464
25.3971	26.4209	27.6969	28.7345	30.2008	31.5934	32.7853	33.8297	35.1314	36.4657
25.3902	26.4192	27.6935	28.7447	30.1905	31.5985	32.8162	33.8229	35.1297	36.452
25.3902	26.4192	27.6935	28.7447	30.1905	31.5985	32.8162	33.8229	35.1297	36.4743
25.3748	26.4141	27.6935	28.7327	30.1888	31.6037	32.8024	33.8675	35.1245	36.4502
25.3748	26.4192	27.7123	28.7225	30.1768	31.6191	32.8024	33.8452	35.1091	36.4691
25.3748	26.4192	27.7123	28.7225	30.1768	31.6191	32.8024	33.8452	35.1091	36.4451
25.3611	26.4089	27.7175	28.7379	30.1751	31.5831	32.7836	33.8332	35.0868	36.4211
25.3611	26.4089	27.7175	28.7379	30.1751	31.5831	32.7836	33.8332	35.0868	36.3902
25.3576	26.4072	27.7295	28.7327	30.1768	31.6294	32.7973	33.8091	35.1091	36.3834
25.349	26.4089	27.7072	28.7207	30.1922	31.5916	32.7904	33.816	35.1417	36.4005
25.349	26.4089	27.7072	28.7207	30.1922	31.5916	32.7904	33.816	35.1417	36.3645
25.3525	26.4158	27.7055	28.7002	30.2248	31.5831	32.7836	33.8349	35.1571	36.3731
25.3405	26.4415	27.714	28.7516	30.2214	31.5762	32.823	33.8074	35.1348	36.3508
25.3405	26.4415	27.714	28.7516	30.2214	31.5762	32.823	33.8074	35.1348	36.3765
25.3439	26.4158	27.7055	28.7225	30.2402	31.5865	32.8179	33.8126	35.1348	36.3782
25.3508	26.4175	27.7072	28.7036	30.2351	31.5728	32.8144	33.8211	35.128	36.3679
25.3508	26.4175	27.7072	28.7036	30.2351	31.5728	32.8144	33.8211	35.128	36.3456
25.349	26.4226	27.7003	28.7173	30.2299	31.5745	32.8076	33.828	35.1314	36.3731
25.3593	26.4175	27.6986	28.7379	30.2179	31.6019	32.8144	33.8572	35.1314	36.3576
25.3593	26.4175	27.6986	28.7379	30.2179	31.6019	32.8144	33.8572	35.1314	36.3731
25.3576	26.3901	27.714	28.7173	30.1974	31.6174	32.8076	33.876	35.1314	36.3456
25.3576	26.3901	27.714	28.7173	30.1974	31.6174	32.8076	33.876	35.1314	36.3885
25.3576	26.3798	27.726	28.6984	30.2025	31.6088	32.8162	33.8589	35.1074	36.4039
25.3611	26.3883	27.7192	28.767	30.1974	31.6225	32.799	33.8486	35.1606	36.3954
25.3611	26.3883	27.7192	28.767	30.1974	31.6225	32.799	33.8486	35.1606	36.4039
25.349	26.378	27.7003	28.7842	30.1871	31.5968	32.8076	33.8554	35.1588	36.4331
25.3525	26.3763	27.7037	28.7602	30.2111	31.6139	32.8076	33.8143	35.1674	36.4057
25.3525	26.3763	27.7037	28.7602	30.2111	31.6139	32.8076	33.8143	35.1674	36.3885
25.3559	26.3952	27.7055	28.7499	30.2111	31.6088	32.8247	33.8383	35.1468	36.3593
25.3645	26.4038	27.7003	28.7636	30.2265	31.6037	32.8127	33.8349	35.164	36.3954
25.3645	26.4038	27.7003	28.7636	30.2265	31.6037	32.8127	33.8349	35.164	36.3731
25.3851	26.4261	27.666	28.7379	30.2539	31.5899	32.823	33.8023	35.1383	36.3765
25.3851	26.4261	27.666	28.7379	30.2539	31.5899	32.823	33.8023	35.1383	36.3799
25.3919	26.4072	27.6917	28.7362	30.2454	31.6277	32.7939	33.7989	35.1245	36.3679
25.4091	26.4038	27.6763	28.7499	30.2265	31.6345	32.8453	33.8143	35.1348	36.313
25.4091	26.4038	27.6763	28.7499	30.2265	31.6345	32.8453	33.8143	35.1348	36.3593
25.4211	26.4055	27.666	28.7396	30.2179	31.6208	32.8059	33.7714	35.1366	36.3542
25.4159	26.3952	27.678	28.755	30.2094	31.6191	32.8059	33.7937	35.1623	36.301
25.4159	26.3952	27.678	28.755	30.2094	31.6191	32.8059	33.7937	35.1623	36.3456
25.4228	26.4055	27.678	28.7327	30.2385	31.65	32.8179	33.7937	35.1691	36.3628
25.4211	26.4141	27.6814	28.7447	30.2368	31.6328	32.8179	33.8109	35.1537	36.3388

25.4211	26.4141	27.6814	28.7447	30.2368	31.6328	32.8179	33.8109	35.1537	36.3473
25.4365	26.4072	27.6746	28.7739	30.2711	31.6242	32.7836	33.8229	35.1383	36.3628
25.4399	26.4123	27.6677	28.7739	30.2591	31.6002	32.8385	33.8332	35.1486	36.3799
25.4399	26.4123	27.6677	28.7739	30.2591	31.6002	32.8385	33.8332	35.1486	36.3662
25.4451	26.4158	27.6935	28.743	30.2111	31.6105	32.8505	33.8091	35.128	36.3679
25.4451	26.4158	27.6935	28.743	30.2111	31.6105	32.8505	33.8091	35.128	36.3731
25.4468	26.4278	27.702	28.7516	30.2145	31.6311	32.835	33.8263	35.1074	36.4057
25.4314	26.4261	27.6712	28.7413	30.2042	31.6637	32.7956	33.8589	35.116	36.313
25.4314	26.4261	27.6712	28.7413	30.2042	31.6637	32.7956	33.8589	35.116	36.3268
25.4262	26.4244	27.678	28.7739	30.2145	31.5899	32.7819	33.864	35.1331	36.3268
25.4279	26.4141	27.6917	28.7773	30.2368	31.6122	32.7973	33.8795	35.1366	36.3233
25.4279	26.4141	27.6917	28.7773	30.2368	31.6122	32.7973	33.8795	35.1366	36.3062
25.4108	26.4226	27.7037	28.7516	30.2385	31.6294	32.8316	33.8623	35.1434	36.3439
25.4039	26.4123	27.7037	28.7619	30.2351	31.6088	32.8402	33.8777	35.1348	36.3268
25.4039	26.4123	27.7037	28.7619	30.2351	31.6088	32.8402	33.8777	35.1348	36.3216
25.3851	26.4175	27.7089	28.7516	30.2334	31.5899	32.8419	33.8709	35.1383	36.3439
25.3851	26.4175	27.7089	28.7516	30.2334	31.5899	32.8419	33.8709	35.1383	36.3199
25.3782	26.4209	27.7055	28.7482	30.2419	31.5814	32.8316	33.8572	35.116	36.3079
25.3885	26.4123	27.69	28.743	30.2196	31.6208	32.8367	33.8812	35.1228	36.313
25.3885	26.4123	27.69	28.743	30.2196	31.6208	32.8367	33.8812	35.1228	36.2959
25.3799	26.4089	27.6866	28.7516	30.2111	31.6345	32.8162	33.8589	35.1091	36.2993
25.3713	26.4003	27.6917	28.7653	30.2094	31.5951	32.835	33.876	35.1314	36.2805
25.3713	26.4003	27.6917	28.7653	30.2094	31.5951	32.835	33.876	35.1314	36.277
25.3508	26.3832	27.6969	28.7636	30.2145	31.6105	32.787	33.828	35.1177	36.2959
25.3405	26.3986	27.6883	28.7602	30.2282	31.5848	32.8007	33.8572	35.1383	36.2582
25.3405	26.3986	27.6883	28.7602	30.2282	31.5848	32.8007	33.8572	35.1383	36.2976
25.3508	26.4278	27.6986	28.755	30.2128	31.5711	32.799	33.8263	35.1074	36.2616
25.3611	26.4209	27.7003	28.7516	30.1905	31.5968	32.7613	33.8366	35.1331	36.2942
25.3611	26.4209	27.7003	28.7516	30.1905	31.5968	32.7613	33.8366	35.1331	36.3422
25.3971	26.4158	27.6986	28.7482	30.1888	31.6208	32.7784	33.8589	35.116	36.3165
25.3971	26.4158	27.6986	28.7482	30.1888	31.6208	32.7784	33.8589	35.116	36.2942
25.4039	26.4261	27.6952	28.7396	30.1733	31.5968	32.8042	33.8452	35.1434	36.2925
25.4074	26.4209	27.6883	28.7602	30.1528	31.6002	32.787	33.8349	35.1503	36.313
25.4074	26.4209	27.6883	28.7602	30.1528	31.6002	32.787	33.8349	35.1503	36.3045
25.4194	26.4295	27.6866	28.7499	30.1665	31.5796	32.7784	33.8417	35.1451	36.2993
25.4245	26.4346	27.6883	28.7619	30.1631	31.5659	32.7973	33.8572	35.1211	36.313
25.4245	26.4346	27.6883	28.7619	30.1631	31.5659	32.7973	33.8572	35.1211	36.2719
25.4297	26.4484	27.678	28.7413	30.1819	31.5848	32.7613	33.8366	35.0868	36.2959
25.4331	26.4261	27.6917	28.7225	30.1768	31.5831	32.7699	33.8537	35.092	36.2599
25.4331	26.4261	27.6917	28.7225	30.1768	31.5831	32.7699	33.8537	35.092	36.2856
25.4417	26.4123	27.7037	28.7773	30.1819	31.5728	32.8247	33.8349	35.0782	36.2513
25.4417	26.4123	27.7037	28.7773	30.1819	31.5728	32.8247	33.8349	35.0782	36.2907
25.4434	26.4106	27.6883	28.7808	30.2042	31.5899	32.8316	33.8675	35.092	36.2805

25.4417	26.4021	27.6609	28.7499	30.2385	31.5659	32.8093	33.8417	35.092	36.289
25.4417	26.4021	27.6609	28.7499	30.2385	31.5659	32.8093	33.8417	35.092	36.2633
25.4468	26.3969	27.6643	28.7533	30.2179	31.5711	32.7956	33.828	35.104	36.2787
25.4519	26.4106	27.6677	28.7362	30.2334	31.5436	32.799	33.8726	35.0714	36.2736
25.4519	26.4106	27.6677	28.7362	30.2334	31.5436	32.799	33.8726	35.0714	36.2479
25.4468	26.4003	27.6523	28.7225	30.2437	31.5711	32.8042	33.8366	35.0354	36.2393
25.4451	26.3901	27.6506	28.7516	30.2419	31.6105	32.8213	33.8469	35.0885	36.2342
25.4451	26.3901	27.6506	28.7516	30.2419	31.6105	32.8213	33.8469	35.0885	36.241
25.4468	26.3866	27.6557	28.7499	30.2231	31.5968	32.8144	33.8023	35.0748	36.2204
25.4554	26.378	27.6489	28.7482	30.2522	31.5796	32.7956	33.828	35.068	36.2719
25.4554	26.378	27.6489	28.7482	30.2522	31.5796	32.7956	33.828	35.068	36.253
25.4434	26.342	27.6317	28.7705	30.2574	31.5865	32.7973	33.8434	35.0868	36.2153
25.4434	26.342	27.6317	28.7705	30.2574	31.5865	32.7973	33.8434	35.0868	36.217
25.4502	26.3626	27.6506	28.7842	30.266	31.5745	32.7733	33.8366	35.0885	36.2273
25.4519	26.3643	27.6403	28.7808	30.2282	31.6088	32.8042	33.792	35.1297	36.2702
25.4519	26.3643	27.6403	28.7808	30.2282	31.6088	32.8042	33.792	35.1297	36.2204
25.4502	26.3695	27.654	28.7859	30.2162	31.5848	32.8333	33.7903	35.1194	36.2187
25.4485	26.3832	27.6403	28.8031	30.2094	31.5916	32.8144	33.7971	35.1211	36.253
25.4485	26.3832	27.6403	28.8031	30.2094	31.5916	32.8144	33.7971	35.1211	36.2564
25.4622	26.3729	27.6214	28.7808	30.1991	31.5934	32.811	33.8297	35.0748	36.2753
25.4622	26.378	27.6111	28.7688	30.2248	31.5934	32.7681	33.852	35.0731	36.2582
25.4622	26.378	27.6111	28.7688	30.2248	31.5934	32.7681	33.852	35.0731	36.2839
25.4622	26.3695	27.6403	28.7482	30.2488	31.6122	32.7218	33.8314	35.0577	36.2273
25.4622	26.3695	27.6403	28.7482	30.2488	31.6122	32.7218	33.8314	35.0577	36.2599
25.4554	26.3729	27.6609	28.7602	30.2454	31.6071	32.739	33.8434	35.104	36.2479
25.4657	26.3729	27.6386	28.767	30.2471	31.5728	32.7407	33.8434	35.1108	36.2873
25.4657	26.3729	27.6386	28.767	30.2471	31.5728	32.7407	33.8434	35.1108	36.2753
25.4725	26.378	27.6489	28.7756	30.2539	31.5762	32.7441	33.8554	35.1023	36.2753
25.4519	26.3798	27.642	28.7756	30.2642	31.5951	32.739	33.8692	35.1331	36.2702
25.4519	26.3798	27.642	28.7756	30.2642	31.5951	32.739	33.8692	35.1331	36.253
25.4519	26.3729	27.6403	28.7516	30.2248	31.5796	32.7836	33.8469	35.1005	36.2633
25.4434	26.3609	27.6557	28.779	30.2196	31.5659	32.7887	33.8795	35.1314	36.2702
25.4434	26.3609	27.6557	28.779	30.2196	31.5659	32.7887	33.8795	35.1314	36.2753
25.4571	26.3472	27.6574	28.7516	30.2831	31.5899	32.7973	33.8452	35.0594	36.2856
25.4605	26.3437	27.6643	28.7413	30.2642	31.5694	32.8264	33.8349	35.0765	36.2685
25.4605	26.3437	27.6643	28.7413	30.2642	31.5694	32.8264	33.8349	35.0765	36.2907
25.4485	26.366	27.666	28.731	30.2385	31.5831	32.8333	33.8503	35.1245	36.2907
25.4485	26.366	27.666	28.731	30.2385	31.5831	32.8333	33.8503	35.1245	36.265
25.4451	26.3746	27.6609	28.731	30.2574	31.5591	32.8076	33.8452	35.0971	36.3045
25.4314	26.3883	27.6454	28.7619	30.2488	31.5676	32.8007	33.8434	35.1108	36.2925
25.4314	26.3883	27.6454	28.7619	30.2488	31.5676	32.8007	33.8434	35.1108	36.2925
25.4245	26.3866	27.6729	28.7516	30.2265	31.5625	32.7973	33.8143	35.0954	36.3302
25.4228	26.3849	27.6729	28.767	30.2128	31.5419	32.8093	33.8314	35.1005	36.3165

25.4228	26.3849	27.6729	28.767	30.2128	31.5419	32.8093	33.8314	35.1005	36.3096
25.4176	26.3918	27.6626	28.7242	30.2042	31.5471	32.8213	33.8469	35.1263	36.3062
25.4245	26.3986	27.6626	28.7447	30.2145	31.5488	32.799	33.8486	35.1194	36.3388
25.4245	26.3986	27.6626	28.7447	30.2145	31.5488	32.799	33.8486	35.1194	36.2976
25.4211	26.3866	27.6609	28.7482	30.1853	31.5951	32.8419	33.8537	35.1297	36.2993
25.4211	26.3866	27.6609	28.7482	30.1853	31.5951	32.8419	33.8537	35.1297	36.325
25.4262	26.4021	27.6609	28.7345	30.2179	31.5522	32.823	33.8469	35.1091	36.3199
25.4091	26.3866	27.6917	28.7499	30.2385	31.5796	32.8127	33.8366	35.1091	36.2959
25.4091	26.3866	27.6917	28.7499	30.2385	31.5796	32.8127	33.8366	35.1091	36.2993
25.4039	26.3763	27.678	28.7499	30.2488	31.5659	32.8487	33.8572	35.0594	36.3045
25.3851	26.3643	27.6523	28.7482	30.2557	31.5368	32.811	33.8091	35.0988	36.313
25.3851	26.3643	27.6523	28.7482	30.2557	31.5368	32.811	33.8091	35.0988	36.301
25.4056	26.3712	27.6437	28.7859	30.2471	31.5573	32.8093	33.7817	35.068	36.3028
25.4074	26.3695	27.6437	28.779	30.2488	31.5659	32.7939	33.7766	35.1091	36.3199
25.4074	26.3695	27.6437	28.779	30.2488	31.5659	32.7939	33.7766	35.1091	36.3165
25.4005	26.366	27.6351	28.7859	30.266	31.5848	32.8247	33.8229	35.152	36.3148
25.4108	26.378	27.6386	28.7876	30.2385	31.5745	32.8024	33.8349	35.1211	36.3439
25.4108	26.378	27.6386	28.7876	30.2385	31.5745	32.8024	33.8349	35.1211	36.3199
25.4159	26.3866	27.6471	28.8133	30.2522	31.5728	32.775	33.7954	35.116	36.3611
25.4159	26.3866	27.6471	28.8133	30.2522	31.5728	32.775	33.7954	35.116	36.3353
25.4142	26.378	27.6557	28.8013	30.2625	31.5659	32.8024	33.816	35.1331	36.3525
25.4022	26.3678	27.6266	28.7825	30.2711	31.5625	32.8059	33.828	35.1297	36.3662
25.4022	26.3678	27.6266	28.7825	30.2711	31.5625	32.8059	33.828	35.1297	36.3371
25.4108	26.3575	27.6574	28.8031	30.2402	31.5728	32.7904	33.8417	35.1211	36.325
25.4022	26.3506	27.6351	28.7996	30.2471	31.5728	32.7887	33.8366	35.1451	36.3371
25.4022	26.3506	27.6351	28.7996	30.2471	31.5728	32.7887	33.8366	35.1451	36.3593
25.3988	26.3335	27.6351	28.7928	30.2642	31.5865	32.775	33.78	35.1571	36.3285
25.3988	26.3283	27.6506	28.8048	30.2557	31.5934	32.7956	33.8143	35.1726	36.325
25.3988	26.3283	27.6506	28.8048	30.2557	31.5934	32.7956	33.8143	35.1726	36.3473
25.3902	26.3369	27.6609	28.7911	30.2351	31.5728	32.8162	33.8383	35.1194	36.3319
25.3902	26.3369	27.6609	28.7911	30.2351	31.5728	32.8162	33.8383	35.1194	36.3662
25.3868	26.3455	27.6677	28.7739	30.2368	31.5762	32.7921	33.852	35.0885	36.3422
25.3799	26.3472	27.6523	28.7499	30.2385	31.5882	32.7733	33.84	35.1005	36.3628
25.3799	26.3472	27.6523	28.7499	30.2385	31.5882	32.7733	33.84	35.1005	36.3456
25.3799	26.3558	27.642	28.7688	30.2608	31.6122	32.7561	33.8211	35.0851	36.3628
25.3799	26.3437	27.6489	28.7876	30.2214	31.6191	32.7407	33.8675	35.1348	36.4039
25.3799	26.3437	27.6489	28.7876	30.2214	31.6191	32.7407	33.8675	35.1348	36.4074
25.3748	26.354	27.6334	28.7928	30.2454	31.5745	32.7321	33.8229	35.1143	36.3456
25.3731	26.3678	27.6489	28.7928	30.2282	31.5831	32.7338	33.8177	35.116	36.3268
25.3731	26.3678	27.6489	28.7928	30.2282	31.5831	32.7338	33.8177	35.116	36.3353
25.3696	26.3403	27.6437	28.7911	30.2488	31.5745	32.7647	33.7971	35.0902	36.3028
25.3662	26.3386	27.6489	28.7808	30.2505	31.5694	32.7733	33.7731	35.1143	36.3491
25.3662	26.3386	27.6489	28.7808	30.2505	31.5694	32.7733	33.7731	35.1143	36.3302

25.3611	26.3575	27.6334	28.7653	30.2334	31.5934	32.7544	33.792	35.1194	36.3388
25.3611	26.3575	27.6334	28.7653	30.2334	31.5934	32.7544	33.792	35.1194	36.3714
25.3645	26.366	27.6489	28.7396	30.2299	31.5968	32.7647	33.7851	35.1297	36.3525
25.3679	26.3729	27.6214	28.7345	30.2059	31.5762	32.775	33.7611	35.1623	36.3679
25.3679	26.3729	27.6214	28.7345	30.2059	31.5762	32.775	33.7611	35.1623	36.3559
25.3748	26.3712	27.6454	28.7482	30.2334	31.5676	32.7681	33.7731	35.1434	36.3473
25.3748	26.3815	27.6574	28.7636	30.2385	31.5676	32.7356	33.7577	35.1263	36.3405
25.3748	26.3815	27.6574	28.7636	30.2385	31.5676	32.7356	33.7577	35.1263	36.3233
25.3576	26.4003	27.6471	28.7705	30.2334	31.5796	32.7458	33.7903	35.0937	36.3199
25.3593	26.4089	27.6471	28.7499	30.1922	31.5608	32.7819	33.7834	35.1211	36.3371
25.3593	26.4089	27.6471	28.7499	30.1922	31.5608	32.7819	33.7834	35.1211	36.3302
25.3542	26.3815	27.6506	28.7602	30.1665	31.5316	32.7733	33.7766	35.1743	36.3696
25.3542	26.3815	27.6506	28.7602	30.1665	31.5316	32.7733	33.7766	35.1743	36.3405
25.3508	26.3935	27.6437	28.7808	30.1974	31.5659	32.775	33.7543	35.1434	36.3696
25.337	26.3832	27.6283	28.743	30.2162	31.6002	32.7733	33.8297	35.1091	36.3611
25.337	26.3832	27.6283	28.743	30.2162	31.6002	32.7733	33.8297	35.1091	36.3319
25.3302	26.3712	27.6403	28.7242	30.2317	31.5899	32.7578	33.8023	35.1091	36.3439
25.3165	26.3695	27.6609	28.7362	30.2094	31.5985	32.7664	33.8023	35.1245	36.3851
25.3165	26.3695	27.6609	28.7362	30.2094	31.5985	32.7664	33.8023	35.1245	36.4142
25.3062	26.366	27.6369	28.731	30.2214	31.5968	32.7647	33.8194	35.1125	36.3834
25.3096	26.3746	27.6369	28.7602	30.2282	31.5796	32.7819	33.8349	35.0868	36.4177
25.3096	26.3746	27.6369	28.7602	30.2282	31.5796	32.7819	33.8349	35.0868	36.4177
25.3079	26.3729	27.6351	28.7533	30.2368	31.5985	32.7784	33.8503	35.0868	36.4554
25.2959	26.3729	27.654	28.7465	30.2282	31.6122	32.7647	33.8469	35.1143	36.4228
25.2959	26.3729	27.654	28.7465	30.2282	31.6122	32.7647	33.8469	35.1143	36.4674
25.2942	26.3506	27.6746	28.7533	30.2368	31.6105	32.7767	33.8091	35.0988	36.4434
25.2942	26.3506	27.6746	28.7533	30.2368	31.6105	32.7767	33.8091	35.0988	36.4297
25.2942	26.3489	27.6643	28.755	30.2111	31.6311	32.8162	33.7851	35.0885	36.4211
25.2925	26.3558	27.6574	28.707	30.2059	31.6328	32.7801	33.828	35.128	36.4279
25.2925	26.3558	27.6574	28.707	30.2059	31.6328	32.7801	33.828	35.128	36.4005
25.2787	26.366	27.6489	28.7825	30.1751	31.6174	32.7544	33.8263	35.0971	36.4142
25.2633	26.3678	27.6883	28.7533	30.1699	31.5934	32.7664	33.8229	35.0782	36.4211
25.2633	26.3678	27.6883	28.7533	30.1699	31.5934	32.7664	33.8229	35.0782	36.3868
25.2564	26.3678	27.6592	28.7327	30.2008	31.5865	32.7304	33.8263	35.0817	36.4005
25.2547	26.3472	27.6626	28.7276	30.2145	31.5831	32.7767	33.7937	35.0611	36.4005
25.2547	26.3472	27.6626	28.7276	30.2145	31.5831	32.7767	33.7937	35.0611	36.4091
25.2393	26.3386	27.678	28.7259	30.1974	31.6002	32.7561	33.8006	35.0971	36.4159
25.2393	26.3386	27.678	28.7259	30.1974	31.6002	32.7561	33.8006	35.0971	36.4022
25.2341	26.342	27.666	28.7499	30.2008	31.5985	32.7458	33.8057	35.0611	36.4057
25.2393	26.3455	27.6797	28.719	30.2265	31.6037	32.7681	33.8297	35.1297	36.3679
25.2393	26.3455	27.6797	28.719	30.2265	31.6037	32.7681	33.8297	35.1297	36.3559
25.229	26.3472	27.6643	28.743	30.2539	31.5985	32.7716	33.7817	35.1194	36.3559
25.2341	26.342	27.6814	28.7688	30.2642	31.6054	32.7527	33.8126	35.1383	36.3302

25.2341	26.342	27.6814	28.7688	30.2642	31.6054	32.7527	33.8126	35.1383	36.3508
25.241	26.3403	27.6935	28.7602	30.2385	31.6191	32.7578	33.816	35.1451	36.3268
25.2359	26.3437	27.6694	28.7379	30.2299	31.5968	32.7613	33.8177	35.1691	36.3422
25.2359	26.3437	27.6694	28.7379	30.2299	31.5968	32.7613	33.8177	35.1691	36.3456
25.2273	26.354	27.6797	28.7413	30.2642	31.5865	32.787	33.804	35.1451	36.3182
25.2273	26.3575	27.666	28.7619	30.2454	31.5882	32.7819	33.8023	35.1434	36.3182
25.2273	26.3575	27.666	28.7619	30.2454	31.5882	32.7819	33.8023	35.1434	36.2976
25.2187	26.3283	27.7055	28.7362	30.2368	31.5985	32.7561	33.84	35.1143	36.3028
25.2187	26.3283	27.7055	28.7362	30.2368	31.5985	32.7561	33.84	35.1143	36.2925
25.2221	26.3403	27.7037	28.7276	30.2214	31.5899	32.7767	33.8126	35.1537	36.3148
25.2324	26.342	27.6832	28.7327	30.2214	31.6019	32.7733	33.8469	35.152	36.301
25.2324	26.342	27.6832	28.7327	30.2214	31.6019	32.7733	33.8469	35.152	36.3045
25.2393	26.3146	27.6814	28.7482	30.2334	31.5916	32.7784	33.8057	35.1314	36.2856
25.241	26.3266	27.6952	28.7482	30.2094	31.5865	32.7853	33.8006	35.1228	36.3216
25.241	26.3266	27.6952	28.7482	30.2094	31.5865	32.7853	33.8006	35.1228	36.2599
25.2479	26.3437	27.6969	28.7225	30.2145	31.5951	32.7956	33.852	35.1245	36.2925
25.2496	26.3403	27.7157	28.7345	30.1991	31.6002	32.763	33.84	35.1468	36.3319
25.2496	26.3403	27.7157	28.7345	30.1991	31.6002	32.763	33.84	35.1468	36.3439
25.241	26.3335	27.7089	28.7396	30.2265	31.6088	32.7904	33.8194	35.1451	36.3714
25.241	26.3335	27.7089	28.7396	30.2265	31.6088	32.7904	33.8194	35.1451	36.3079
25.2221	26.3215	27.6986	28.7345	30.2094	31.6037	32.7767	33.8091	35.1434	36.3302
25.2136	26.3575	27.7037	28.7396	30.2076	31.5985	32.7819	33.7989	35.1486	36.3182
25.2136	26.3575	27.7037	28.7396	30.2076	31.5985	32.7819	33.7989	35.1486	36.3371
25.2204	26.3695	27.6917	28.7225	30.2025	31.6105	32.7939	33.8074	35.1571	36.325
25.2324	26.3678	27.6763	28.7533	30.2042	31.5556	32.8162	33.8109	35.1949	36.3525
25.2324	26.3678	27.6763	28.7533	30.2042	31.5556	32.8162	33.8109	35.1949	36.3576
25.241	26.3678	27.69	28.7379	30.1974	31.5968	32.7767	33.8469	35.1743	36.3353
25.2376	26.3815	27.6883	28.7636	30.2059	31.5814	32.7853	33.8091	35.2172	36.3079
25.2376	26.3815	27.6883	28.7636	30.2059	31.5814	32.7853	33.8091	35.2172	36.313
25.2444	26.3626	27.6763	28.7362	30.1991	31.5608	32.7647	33.7886	35.188	36.301
25.2393	26.3575	27.6746	28.7259	30.2214	31.5522	32.7407	33.7989	35.1777	36.3062
25.2393	26.3575	27.6746	28.7259	30.2214	31.5522	32.7407	33.7989	35.1777	36.2822
25.2359	26.3575	27.6694	28.7396	30.2231	31.5762	32.7699	33.8091	35.1571	36.2736
25.2359	26.3575	27.6694	28.7396	30.2231	31.5762	32.7699	33.8091	35.1571	36.2393
25.2307	26.3575	27.6557	28.7293	30.2179	31.5642	32.7201	33.8091	35.1811	36.2873
25.2324	26.3437	27.6574	28.7345	30.2402	31.5676	32.7441	33.8434	35.1623	36.253
25.2324	26.3437	27.6574	28.7345	30.2402	31.5676	32.7441	33.8434	35.1623	36.2787
25.2239	26.3609	27.6643	28.7242	30.2128	31.5796	32.7544	33.7989	35.1348	36.2633
25.2204	26.3472	27.6643	28.7396	30.1871	31.5762	32.7304	33.8177	35.1468	36.2753
25.2204	26.3472	27.6643	28.7396	30.1871	31.5762	32.7304	33.8177	35.1468	36.2993
25.2101	26.3592	27.6763	28.7482	30.1974	31.5505	32.787	33.8297	35.1571	36.3028
25.205	26.3626	27.6557	28.7482	30.2111	31.5694	32.7973	33.8452	35.1674	36.2787
25.205	26.3626	27.6557	28.7482	30.2111	31.5694	32.7973	33.8452	35.1674	36.2907

25.1998	26.3746	27.6866	28.7876	30.1768	31.5831	32.7853	33.8452	35.1949	36.2856
25.1998	26.3746	27.6866	28.7876	30.1768	31.5831	32.7853	33.8452	35.1949	36.3216
25.2033	26.3763	27.6883	28.7722	30.2265	31.5814	32.7921	33.828	35.1709	36.2805
25.205	26.3746	27.6609	28.7893	30.2214	31.5471	32.7767	33.804	35.1606	36.2993
25.205	26.3746	27.6609	28.7893	30.2214	31.5471	32.7767	33.804	35.1606	36.2547
25.2067	26.3815	27.6557	28.7945	30.2111	31.5591	32.799	33.8006	35.1571	36.2667
25.1981	26.3643	27.6557	28.7876	30.2419	31.5659	32.8144	33.8143	35.1829	36.2633
25.1981	26.3643	27.6557	28.7876	30.2419	31.5659	32.8144	33.8143	35.1829	36.2736
25.2118	26.3763	27.6557	28.767	30.2522	31.5728	32.8316	33.8246	35.2034	36.2667
25.2221	26.3746	27.654	28.7636	30.266	31.6019	32.8179	33.8263	35.2172	36.2702
25.2221	26.3746	27.654	28.7636	30.266	31.6019	32.8179	33.8263	35.2172	36.2462
25.2324	26.3746	27.6832	28.7653	30.2059	31.6037	32.8093	33.8366	35.224	36.2393
25.2324	26.3678	27.6746	28.7859	30.2128	31.5916	32.8264	33.7783	35.2017	36.241
25.2324	26.3678	27.6746	28.7859	30.2128	31.5916	32.8264	33.7783	35.2017	36.2667
25.2307	26.366	27.6729	28.7653	30.2385	31.5831	32.8402	33.8297	35.2172	36.2616
25.2307	26.366	27.6729	28.7653	30.2385	31.5831	32.8402	33.8297	35.2172	36.253
25.229	26.3832	27.6626	28.7808	30.2334	31.5916	32.8453	33.8177	35.2206	36.2582
25.2307	26.3695	27.666	28.7482	30.2076	31.5694	32.8144	33.7886	35.188	36.2667
25.2307	26.3695	27.666	28.7482	30.2076	31.5694	32.8144	33.7886	35.188	36.2256
25.2256	26.3695	27.6609	28.7636	30.1871	31.6174	32.8247	33.78	35.188	36.2307
25.2341	26.3763	27.6832	28.7602	30.1819	31.6139	32.8453	33.7834	35.1811	36.2479
25.2341	26.3763	27.6832	28.7602	30.1819	31.6139	32.8453	33.7834	35.1811	36.277
25.2513	26.4123	27.6849	28.7825	30.2042	31.638	32.847	33.7937	35.1846	36.2496
25.2719	26.4072	27.6866	28.7585	30.1922	31.6071	32.8144	33.7937	35.2017	36.2393
25.2719	26.4072	27.6866	28.7585	30.1922	31.6071	32.8144	33.7937	35.2017	36.2376
25.277	26.3952	27.6969	28.7345	30.1768	31.6191	32.8419	33.8126	35.1777	36.1947
25.277	26.3952	27.6969	28.7345	30.1768	31.6191	32.8419	33.8126	35.1777	36.2153
25.2736	26.3918	27.6969	28.7327	30.2248	31.6002	32.811	33.8383	35.164	36.193
25.2736	26.3729	27.6797	28.7722	30.2368	31.5865	32.8693	33.816	35.1674	36.1913
25.2736	26.3729	27.6797	28.7722	30.2368	31.5865	32.8693	33.816	35.1674	36.1638
25.2684	26.3609	27.6866	28.767	30.2539	31.6037	32.847	33.8126	35.1863	36.145
25.2667	26.3678	27.6814	28.7962	30.2248	31.5951	32.8367	33.7748	35.2154	36.1656
25.2667	26.3678	27.6814	28.7962	30.2248	31.5951	32.8367	33.7748	35.2154	36.1999
25.2667	26.3746	27.6609	28.8013	30.2248	31.6122	32.8024	33.7525	35.236	36.1776
25.2684	26.378	27.6609	28.779	30.2402	31.5916	32.8042	33.7594	35.2446	36.1896
25.2684	26.378	27.6609	28.779	30.2402	31.5916	32.8042	33.7594	35.2446	36.1467
25.265	26.3729	27.666	28.7979	30.2368	31.5779	32.8179	33.7971	35.2292	36.133
25.265	26.3712	27.6643	28.8048	30.2505	31.6105	32.8213	33.8023	35.2343	36.1244
25.265	26.3712	27.6643	28.8048	30.2505	31.6105	32.8213	33.8023	35.2343	36.1621
25.2582	26.3369	27.6626	28.7842	30.2214	31.6071	32.8505	33.804	35.2257	36.157
25.2582	26.3369	27.6626	28.7842	30.2214	31.6071	32.8505	33.804	35.2257	36.1313
25.2599	26.3489	27.6574	28.7705	30.2094	31.6037	32.8539	33.7783	35.2223	36.1192
25.2719	26.3352	27.6643	28.7859	30.2111	31.5814	32.8607	33.7954	35.212	36.169

25.2719	26.3352	27.6643	28.7859	30.2111	31.5814	32.8607	33.7954	35.212	36.1415
25.2856	26.3472	27.6712	28.7773	30.2076	31.6122	32.8385	33.7543	35.2052	36.1501
25.2787	26.3386	27.678	28.7705	30.2231	31.6105	32.8385	33.7817	35.2052	36.1535
25.2787	26.3386	27.678	28.7705	30.2231	31.6105	32.8385	33.7817	35.2052	36.1535
25.2787	26.3506	27.69	28.7516	30.2265	31.5934	32.8642	33.8109	35.2	36.1415
25.277	26.3729	27.6986	28.755	30.2042	31.6277	32.8556	33.8109	35.2223	36.1741
25.277	26.3729	27.6986	28.755	30.2042	31.6277	32.8556	33.8109	35.2223	36.1981
25.2736	26.3592	27.6917	28.7327	30.2094	31.5899	32.8367	33.804	35.2377	36.1467
25.2736	26.3592	27.6917	28.7327	30.2094	31.5899	32.8367	33.804	35.2377	36.1621
25.277	26.3575	27.6763	28.7242	30.2145	31.5916	32.8179	33.8297	35.2412	36.1364
25.2702	26.3575	27.6763	28.7293	30.2248	31.6139	32.8316	33.8091	35.2017	36.1364
25.2702	26.3575	27.6763	28.7293	30.2248	31.6139	32.8316	33.8091	35.2017	36.121
25.2753	26.354	27.6609	28.7242	30.2505	31.5831	32.835	33.7954	35.2137	36.1535
25.2719	26.3558	27.6626	28.731	30.2248	31.5865	32.823	33.804	35.2549	36.145
25.2719	26.3558	27.6626	28.731	30.2248	31.5865	32.823	33.804	35.2549	36.1724
25.2667	26.3609	27.6797	28.7207	30.2402	31.6105	32.8247	33.8057	35.2463	36.1638
25.2719	26.3609	27.6832	28.7104	30.2368	31.5951	32.8402	33.8332	35.2429	36.1398
25.2719	26.3609	27.6832	28.7104	30.2368	31.5951	32.8402	33.8332	35.2429	36.181
25.2856	26.3386	27.6917	28.7002	30.2694	31.5814	32.8487	33.8246	35.2086	36.145
25.2839	26.3437	27.6935	28.7156	30.278	31.6054	32.8299	33.8332	35.1931	36.1175
25.2839	26.3437	27.6935	28.7156	30.278	31.6054	32.8299	33.8332	35.1931	36.1141
25.3096	26.3386	27.6883	28.7259	30.2608	31.6448	32.7956	33.8897	35.164	36.1347
25.3096	26.3386	27.6883	28.7259	30.2608	31.6448	32.7956	33.8897	35.164	36.1364
25.3182	26.3489	27.6814	28.7139	30.2677	31.6208	32.8007	33.84	35.2052	36.1433
25.3268	26.3437	27.7003	28.7036	30.2608	31.6105	32.799	33.8795	35.2103	36.0918
25.3268	26.3437	27.7003	28.7036	30.2608	31.6105	32.799	33.8795	35.2103	36.1107
25.3422	26.3472	27.7226	28.731	30.2728	31.6259	32.8402	33.8452	35.1931	36.121
25.3388	26.3437	27.7226	28.7482	30.2591	31.6277	32.8213	33.8366	35.1846	36.1398
25.3388	26.3437	27.7226	28.7482	30.2591	31.6277	32.8213	33.8366	35.1846	36.157
25.3439	26.3249	27.7003	28.731	30.2625	31.6225	32.8196	33.8297	35.1726	36.1433
25.3508	26.342	27.6866	28.7276	30.2591	31.5779	32.8179	33.852	35.212	36.157
25.3508	26.342	27.6866	28.7276	30.2591	31.5779	32.8179	33.852	35.212	36.1776
25.3645	26.3386	27.6883	28.731	30.2797	31.5573	32.8024	33.8469	35.1709	36.1656
25.3645	26.3386	27.6883	28.731	30.2797	31.5573	32.8024	33.8469	35.1709	36.1656
25.3679	26.3455	27.6986	28.743	30.2608	31.5676	32.8453	33.8606	35.1794	36.1758
25.3611	26.3386	27.7089	28.7276	30.2574	31.6088	32.8522	33.8657	35.1691	36.169
25.3611	26.3386	27.7089	28.7276	30.2574	31.6088	32.8522	33.8657	35.1691	36.1724
25.3628	26.3335	27.6832	28.7242	30.278	31.6019	32.8127	33.8692	35.1623	36.1501
25.3662	26.3335	27.6883	28.731	30.2728	31.6037	32.8453	33.8366	35.1949	36.1656
25.3662	26.3335	27.6883	28.731	30.2728	31.6037	32.8453	33.8366	35.1949	36.181
25.3662	26.33	27.6969	28.7516	30.2248	31.5916	32.823	33.8297	35.1966	36.1638
25.349	26.3266	27.7089	28.7602	30.2454	31.5711	32.8316	33.828	35.2154	36.1861
25.349	26.3266	27.7089	28.7602	30.2454	31.5711	32.8316	33.828	35.2154	36.1947

25.3439	26.3009	27.7003	28.7499	30.2539	31.5899	32.8539	33.8709	35.2034	36.1673
25.3473	26.306	27.714	28.719	30.2848	31.5865	32.8385	33.8692	35.1897	36.1844
25.3473	26.306	27.714	28.719	30.2848	31.5865	32.8385	33.8692	35.1897	36.1827
25.3388	26.3094	27.6935	28.731	30.2677	31.5985	32.8522	33.8777	35.2	36.145
25.3388	26.3094	27.6935	28.731	30.2677	31.5985	32.8522	33.8777	35.2	36.1553
25.3319	26.3146	27.6797	28.7139	30.2608	31.6174	32.8659	33.8915	35.188	36.1758
25.3268	26.3283	27.6969	28.7259	30.2454	31.6174	32.8453	33.8846	35.1811	36.1913
25.3268	26.3283	27.6969	28.7259	30.2454	31.6174	32.8453	33.8846	35.1811	36.1793
25.3216	26.2923	27.7072	28.7345	30.2437	31.6362	32.8213	33.8589	35.2	36.2016
25.3182	26.2803	27.7329	28.7207	30.2317	31.6191	32.8127	33.8434	35.1931	36.1776
25.3182	26.2803	27.7329	28.7207	30.2317	31.6191	32.8127	33.8434	35.1931	36.1501
25.3182	26.3026	27.6712	28.7293	30.2385	31.5899	32.8162	33.8623	35.1657	36.1261
25.3216	26.3163	27.6866	28.7413	30.2419	31.5899	32.8264	33.8623	35.128	36.1484
25.3216	26.3163	27.6866	28.7413	30.2419	31.5899	32.8264	33.8623	35.128	36.1741
25.313	26.3215	27.6883	28.7259	30.2694	31.5728	32.8076	33.8572	35.1691	36.1553
25.313	26.3215	27.6883	28.7259	30.2694	31.5728	32.8076	33.8572	35.1691	36.1518
25.3199	26.3215	27.6763	28.7207	30.2317	31.5591	32.7956	33.8434	35.1417	36.1587
25.3199	26.3283	27.6917	28.7293	30.2437	31.5968	32.8024	33.8675	35.1623	36.1415
25.3199	26.3283	27.6917	28.7293	30.2437	31.5968	32.8024	33.8675	35.1623	36.1227
25.3096	26.3266	27.6866	28.7327	30.2728	31.6054	32.7801	33.8589	35.1486	36.1347
25.2942	26.3386	27.6677	28.7396	30.2625	31.6002	32.7939	33.8606	35.1486	36.1295
25.2942	26.3386	27.6677	28.7396	30.2625	31.6002	32.7939	33.8606	35.1486	36.1553
25.2907	26.3215	27.6437	28.6967	30.2797	31.6054	32.7887	33.9018	35.1554	36.1896
25.2993	26.3215	27.6609	28.6933	30.2694	31.5951	32.7767	33.9189	35.1691	36.2239
25.2993	26.3215	27.6609	28.6933	30.2694	31.5951	32.7767	33.9189	35.1691	36.1844
25.3147	26.3026	27.6454	28.7173	30.2814	31.5882	32.7836	33.8812	35.1811	36.2256
25.313	26.3129	27.6609	28.7139	30.2762	31.5831	32.7819	33.8915	35.1794	36.2033
25.313	26.3129	27.6609	28.7139	30.2762	31.5831	32.7819	33.8915	35.1794	36.1604
25.3336	26.342	27.6832	28.7293	30.2642	31.6037	32.7767	33.8777	35.176	36.1758
25.3336	26.342	27.6832	28.7293	30.2642	31.6037	32.7767	33.8777	35.176	36.1878
25.3353	26.3386	27.6626	28.7207	30.2265	31.5728	32.787	33.8606	35.1503	36.181
25.3542	26.3506	27.6694	28.7482	30.2419	31.5985	32.8059	33.8623	35.1691	36.1981
25.3542	26.3506	27.6694	28.7482	30.2419	31.5985	32.8059	33.8623	35.1691	36.2016
25.3576	26.3489	27.6746	28.7225	30.2231	31.5728	32.8247	33.8332	35.1143	36.2136
25.3679	26.3437	27.6592	28.7499	30.2591	31.5402	32.811	33.8675	35.1211	36.1501
25.3679	26.3437	27.6592	28.7499	30.2591	31.5402	32.811	33.8675	35.1211	36.1947
25.3731	26.3437	27.6832	28.743	30.2711	31.5694	32.8264	33.8383	35.1468	36.1844
25.3765	26.3437	27.6694	28.731	30.2368	31.5591	32.8127	33.852	35.1297	36.2239
25.3765	26.3437	27.6694	28.731	30.2368	31.5591	32.8127	33.852	35.1297	36.2221
25.3833	26.3523	27.6797	28.7396	30.2128	31.5625	32.8076	33.8469	35.1177	36.205
25.3833	26.3523	27.6797	28.7396	30.2128	31.5625	32.8076	33.8469	35.1177	36.2101
25.3919	26.3575	27.6592	28.7276	30.2419	31.5814	32.8042	33.8383	35.1366	36.193
25.4005	26.3558	27.6814	28.7362	30.2282	31.5642	32.7904	33.9035	35.1074	36.2239

25.4005	26.3558	27.6814	28.7362	30.2282	31.5642	32.7904	33.9035	35.1074	36.2376
25.4125	26.3455	27.6883	28.7327	30.2385	31.5796	32.7904	33.8589	35.1091	36.2427
25.4074	26.342	27.6609	28.7379	30.2196	31.5848	32.8076	33.8675	35.1228	36.2685
25.4074	26.342	27.6609	28.7379	30.2196	31.5848	32.8076	33.8675	35.1228	36.253
25.4091	26.3472	27.6557	28.731	30.2008	31.5796	32.847	33.8812	35.1023	36.2719
25.4022	26.3489	27.678	28.7225	30.2299	31.5694	32.8093	33.8863	35.0885	36.277
25.4022	26.3489	27.678	28.7225	30.2299	31.5694	32.8093	33.8863	35.0885	36.2616
25.4005	26.33	27.6814	28.7122	30.2076	31.6277	32.823	33.8623	35.0765	36.2376
25.3988	26.33	27.6643	28.7413	30.2111	31.6088	32.8247	33.8743	35.0868	36.2221
25.3988	26.33	27.6643	28.7413	30.2111	31.6088	32.8247	33.8743	35.0868	36.2273
25.3919	26.3232	27.6643	28.7362	30.2076	31.6294	32.7716	33.8537	35.0594	36.2633
25.3919	26.3232	27.6643	28.7362	30.2076	31.6294	32.7716	33.8537	35.0594	36.2753
25.3902	26.33	27.6471	28.731	30.2008	31.6157	32.7887	33.8589	35.0354	36.2856
25.3748	26.3369	27.6506	28.7019	30.1716	31.6157	32.787	33.8623	35.0714	36.2616
25.3748	26.3369	27.6506	28.7019	30.1716	31.6157	32.787	33.8623	35.0714	36.2685
25.3662	26.3317	27.6574	28.7139	30.2094	31.5865	32.8213	33.8452	35.0988	36.265
25.3679	26.3386	27.6506	28.731	30.2179	31.5659	32.8213	33.8795	35.1023	36.2599
25.3679	26.3386	27.6506	28.731	30.2179	31.5659	32.8213	33.8795	35.1023	36.2307
25.3542	26.3266	27.6592	28.7104	30.2265	31.5711	32.8247	33.84	35.1211	36.2547
25.3456	26.3386	27.642	28.7156	30.2059	31.5831	32.8144	33.8452	35.128	36.2547
25.3456	26.3386	27.642	28.7156	30.2059	31.5831	32.8144	33.8452	35.128	36.2444
25.349	26.342	27.678	28.7225	30.2145	31.5539	32.8007	33.8795	35.0731	36.2805
25.349	26.342	27.678	28.7225	30.2145	31.5539	32.8007	33.8795	35.0731	36.253
25.3456	26.3317	27.6694	28.731	30.2111	31.5488	32.8007	33.8589	35.0457	36.2959
25.3593	26.33	27.6643	28.7345	30.1956	31.5694	32.8144	33.816	35.0422	36.2719
25.3593	26.33	27.6643	28.7345	30.1956	31.5694	32.8144	33.816	35.0422	36.2873
25.3525	26.318	27.6729	28.7345	30.2162	31.5676	32.7904	33.816	35.0765	36.3165
25.3731	26.3317	27.6729	28.7327	30.2196	31.5556	32.8196	33.7886	35.0577	36.325
25.3731	26.3317	27.6729	28.7327	30.2196	31.5556	32.8196	33.7886	35.0577	36.3216
25.3868	26.3215	27.6797	28.7413	30.2265	31.5968	32.7887	33.7989	35.08	36.277
25.3902	26.3043	27.6832	28.7533	30.2265	31.5711	32.787	33.8211	35.0937	36.2736
25.3902	26.3043	27.6832	28.7533	30.2265	31.5711	32.787	33.8211	35.0937	36.3028
25.3936	26.3232	27.6677	28.7207	30.2728	31.5436	32.7544	33.8469	35.0954	36.3216
25.4056	26.3386	27.654	28.7293	30.2745	31.5351	32.7716	33.8332	35.14	36.3233
25.4056	26.3386	27.654	28.7293	30.2745	31.5351	32.7716	33.8332	35.14	36.3525
25.4228	26.342	27.6763	28.767	30.2025	31.5591	32.7613	33.8486	35.0937	36.3491
25.4228	26.342	27.6763	28.767	30.2025	31.5591	32.7613	33.8486	35.0937	36.3542
25.4365	26.354	27.6866	28.7602	30.1939	31.5625	32.7424	33.8503	35.1005	36.3353
25.4451	26.342	27.6763	28.755	30.2008	31.5916	32.7304	33.8469	35.0885	36.3714
25.4451	26.342	27.6763	28.755	30.2008	31.5916	32.7304	33.8469	35.0885	36.3834
25.4554	26.3335	27.6797	28.755	30.1939	31.5522	32.7784	33.852	35.104	36.3714
25.464	26.3437	27.6626	28.7602	30.2162	31.5625	32.7836	33.8675	35.0971	36.3679
25.464	26.3437	27.6626	28.7602	30.2162	31.5625	32.7836	33.8675	35.0971	36.4005

25.4588	26.3437	27.6986	28.7619	30.1733	31.6054	32.7853	33.8349	35.0817	36.3954
25.464	26.3609	27.7089	28.7756	30.1682	31.5831	32.8076	33.8692	35.0902	36.4657
25.464	26.3609	27.7089	28.7756	30.1682	31.5831	32.8076	33.8692	35.0902	36.4708
25.4622	26.3712	27.6574	28.7533	30.1613	31.5796	32.7801	33.864	35.116	36.4417
25.4622	26.3712	27.6574	28.7533	30.1613	31.5796	32.7801	33.864	35.116	36.4159
25.464	26.3695	27.6712	28.7413	30.1802	31.5831	32.7904	33.8537	35.1091	36.4022
25.4605	26.3575	27.6866	28.7756	30.1751	31.5934	32.7527	33.8314	35.1108	36.3645
25.4605	26.3575	27.6866	28.7756	30.1751	31.5934	32.7527	33.8314	35.1108	36.3834
25.4622	26.3592	27.7175	28.7688	30.1888	31.5848	32.7561	33.8074	35.1005	36.4074
25.464	26.3575	27.6935	28.7259	30.2094	31.5934	32.7801	33.8434	35.0491	36.3834
25.464	26.3575	27.6935	28.7259	30.2094	31.5934	32.7801	33.8434	35.0491	36.3919
25.4519	26.3626	27.6729	28.767	30.1991	31.6277	32.7681	33.8657	35.0731	36.3731
25.4502	26.3558	27.6797	28.7619	30.2059	31.6242	32.7235	33.8743	35.0765	36.3765
25.4502	26.3558	27.6797	28.7619	30.2059	31.6242	32.7235	33.8743	35.0765	36.4125
25.4365	26.3472	27.6729	28.7945	30.1802	31.5951	32.7098	33.8332	35.0817	36.4074
25.4228	26.3558	27.7003	28.7911	30.2076	31.5728	32.7235	33.8537	35.0782	36.4159
25.4228	26.3558	27.7003	28.7911	30.2076	31.5728	32.7235	33.8537	35.0782	36.3902
25.4297	26.3369	27.6694	28.7962	30.2025	31.6208	32.7235	33.8297	35.0748	36.4194
25.4297	26.3369	27.6694	28.7962	30.2025	31.6208	32.7235	33.8297	35.0748	36.4194
25.4399	26.3283	27.6403	28.7825	30.1819	31.6259	32.7253	33.8486	35.0954	36.3936
25.4571	26.33	27.6712	28.7602	30.2094	31.5916	32.7235	33.8143	35.0542	36.4194
25.4571	26.33	27.6712	28.7602	30.2094	31.5916	32.7235	33.8143	35.0542	36.4434
25.4674	26.3283	27.6643	28.7893	30.2008	31.5762	32.7304	33.8246	35.1228	36.464
25.4742	26.3266	27.6746	28.7825	30.1853	31.5488	32.7184	33.8297	35.1125	36.4262
25.4742	26.3266	27.6746	28.7825	30.1853	31.5488	32.7184	33.8297	35.1125	36.4262
25.4674	26.33	27.642	28.7688	30.2059	31.5608	32.7133	33.8246	35.0937	36.4177
25.4674	26.3215	27.6523	28.7585	30.2042	31.5573	32.6995	33.8211	35.1417	36.4142
25.4674	26.3215	27.6523	28.7585	30.2042	31.5573	32.6995	33.8211	35.1417	36.4005
25.464	26.3283	27.6523	28.8305	30.1956	31.5865	32.739	33.8177	35.1057	36.3885
25.464	26.3283	27.6523	28.8305	30.1956	31.5865	32.739	33.8177	35.1057	36.3885
25.4605	26.3266	27.6454	28.8168	30.2162	31.5745	32.7527	33.8023	35.0868	36.4074
25.4708	26.3266	27.6935	28.8236	30.2111	31.5985	32.7407	33.852	35.092	36.3936
25.4708	26.3266	27.6935	28.8236	30.2111	31.5985	32.7407	33.852	35.092	36.4794
25.4537	26.3335	27.6866	28.8254	30.2282	31.5951	32.739	33.8486	35.0817	36.4228
25.4708	26.3283	27.6763	28.8236	30.2368	31.6105	32.7218	33.8554	35.128	36.4262
25.4708	26.3283	27.6763	28.8236	30.2368	31.6105	32.7218	33.8554	35.128	36.4228
25.464	26.3317	27.666	28.8065	30.2677	31.6259	32.7664	33.8554	35.1314	36.464
25.4622	26.3472	27.6506	28.8219	30.278	31.6259	32.7716	33.8572	35.1468	36.452
25.4622	26.3472	27.6506	28.8219	30.278	31.6259	32.7716	33.8572	35.1468	36.5051
25.4862	26.3317	27.7157	28.8322	30.2179	31.6294	32.8076	33.804	35.1228	36.4914
25.4914	26.3197	27.7209	28.8065	30.2299	31.6397	32.7956	33.8366	35.1486	36.5103
25.4914	26.3197	27.7209	28.8065	30.2299	31.6397	32.7956	33.8366	35.1486	36.5017
25.4828	26.306	27.7037	28.7893	30.2505	31.6122	32.7596	33.8109	35.1125	36.524

25.4828	26.306	27.7037	28.7893	30.2505	31.6122	32.7596	33.8109	35.1125	36.5034
25.4845	26.2923	27.6883	28.7979	30.2385	31.65	32.8024	33.84	35.1194	36.4914
25.4897	26.3009	27.6746	28.8476	30.2214	31.6602	32.7596	33.8366	35.1434	36.4828
25.4897	26.3009	27.6746	28.8476	30.2214	31.6602	32.7596	33.8366	35.1434	36.4914
25.4622	26.3077	27.6849	28.8254	30.2094	31.6568	32.7664	33.8623	35.1451	36.4657
25.464	26.3009	27.6986	28.8236	30.2351	31.6843	32.7733	33.8606	35.1623	36.4725
25.464	26.3009	27.6986	28.8236	30.2351	31.6843	32.7733	33.8606	35.1623	36.4863
25.4571	26.3026	27.69	28.8219	30.2196	31.6397	32.7819	33.8297	35.1674	36.4365
25.4605	26.3197	27.6952	28.8528	30.2179	31.6071	32.7887	33.8246	35.1606	36.4622
25.4605	26.3197	27.6952	28.8528	30.2179	31.6071	32.7887	33.8246	35.1606	36.4605
25.4605	26.3009	27.6917	28.8065	30.2179	31.6654	32.7664	33.8434	35.1794	36.464
25.4605	26.3009	27.6917	28.8065	30.2179	31.6654	32.7664	33.8434	35.1794	36.4794
25.4554	26.2957	27.6849	28.7996	30.2368	31.6345	32.7784	33.8572	35.1143	36.5017
25.4468	26.2906	27.69	28.8219	30.2265	31.6345	32.7699	33.8126	35.1211	36.5017
25.4468	26.2906	27.69	28.8219	30.2265	31.6345	32.7699	33.8126	35.1211	36.4794
25.4485	26.2923	27.6952	28.8322	30.2076	31.6414	32.763	33.8383	35.1023	36.4794
25.4417	26.3009	27.6986	28.8408	30.2025	31.6311	32.787	33.8417	35.0988	36.4485
25.4417	26.3009	27.6986	28.8408	30.2025	31.6311	32.787	33.8417	35.0988	36.4554
25.4434	26.3146	27.6917	28.8631	30.2317	31.6534	32.7784	33.8572	35.116	36.4502
25.4399	26.318	27.6797	28.8322	30.2282	31.6431	32.7956	33.8572	35.1417	36.4571
25.4399	26.318	27.6797	28.8322	30.2282	31.6431	32.7956	33.8572	35.1417	36.4657
25.4331	26.3232	27.678	28.8545	30.2265	31.6259	32.8024	33.8589	35.1417	36.4571
25.4279	26.33	27.678	28.8631	30.2179	31.6431	32.8316	33.8657	35.1537	36.4777
25.4279	26.33	27.678	28.8631	30.2179	31.6431	32.8316	33.8657	35.1537	36.4743
25.4245	26.3472	27.6454	28.8339	30.2145	31.6037	32.8179	33.8897	35.116	36.4382
25.4245	26.3472	27.6454	28.8339	30.2145	31.6037	32.8179	33.8897	35.116	36.4605
25.4211	26.3455	27.6489	28.8236	30.2111	31.6054	32.8213	33.8743	35.1108	36.4125
25.4331	26.3369	27.6643	28.8442	30.1613	31.6054	32.8522	33.8743	35.1606	36.4674
25.4331	26.3369	27.6643	28.8442	30.1613	31.6054	32.8522	33.8743	35.1606	36.4468
25.4262	26.3197	27.6797	28.8151	30.1802	31.6037	32.8059	33.8846	35.1537	36.4845
25.4245	26.318	27.6797	28.8288	30.139	31.6105	32.799	33.8572	35.2	36.5
25.4245	26.318	27.6797	28.8288	30.139	31.6105	32.799	33.8572	35.2	36.4605
25.4108	26.3283	27.7055	28.8236	30.1768	31.6414	32.8539	33.8486	35.1691	36.4657
25.4056	26.3266	27.7003	28.8133	30.1853	31.6311	32.8333	33.8229	35.1726	36.476
25.4056	26.3266	27.7003	28.8133	30.1853	31.6311	32.8333	33.8229	35.1726	36.452
25.4074	26.3146	27.69	28.7996	30.1939	31.6139	32.847	33.8297	35.1571	36.4708
25.4074	26.3146	27.69	28.7996	30.1939	31.6139	32.847	33.8297	35.1571	36.4571
25.4039	26.3266	27.6952	28.7636	30.1939	31.6174	32.8299	33.8366	35.1811	36.4605
25.4005	26.3077	27.6626	28.7499	30.1939	31.5985	32.8076	33.8417	35.176	36.4674
25.4005	26.3077	27.6626	28.7499	30.1939	31.5985	32.8076	33.8417	35.176	36.4914
25.3936	26.2872	27.6677	28.7499	30.2128	31.6397	32.8162	33.8452	35.1674	36.4828
25.3988	26.2854	27.6986	28.7705	30.1836	31.6071	32.8144	33.8297	35.1931	36.5017
25.3988	26.2854	27.6986	28.7705	30.1836	31.6071	32.8144	33.8297	35.1931	36.5

25.3936	26.2974	27.6849	28.7568	30.1974	31.6191	32.8024	33.8606	35.2017	36.5051
25.3816	26.2872	27.6917	28.7602	30.2231	31.6157	32.7973	33.8692	35.1966	36.5068
25.3816	26.2872	27.6917	28.7602	30.2231	31.6157	32.7973	33.8692	35.1966	36.4828
25.3851	26.2837	27.702	28.7653	30.2488	31.6174	32.8213	33.8314	35.1554	36.4605
25.3679	26.2854	27.702	28.7722	30.2505	31.6174	32.8522	33.8623	35.1709	36.4348
25.3679	26.2854	27.702	28.7722	30.2505	31.6174	32.8522	33.8623	35.1709	36.4262
25.3645	26.294	27.7089	28.7465	30.2248	31.5711	32.8127	33.8109	35.2034	36.4228
25.3645	26.294	27.7089	28.7465	30.2248	31.5711	32.8127	33.8109	35.2034	36.4194
25.3576	26.2889	27.7106	28.7413	30.2196	31.5796	32.8093	33.8349	35.1417	36.3851
25.3422	26.2889	27.6935	28.755	30.2008	31.5899	32.7733	33.8623	35.1726	36.4091
25.3422	26.2889	27.6935	28.755	30.2008	31.5899	32.7733	33.8623	35.1726	36.4245
25.3388	26.2974	27.7209	28.7499	30.2042	31.5968	32.8144	33.8657	35.1931	36.4262
25.3302	26.2957	27.7106	28.7447	30.2008	31.5985	32.8093	33.8606	35.2052	36.4348
25.3302	26.2957	27.7106	28.7447	30.2008	31.5985	32.8093	33.8606	35.2052	36.4554
25.3199	26.282	27.6592	28.7259	30.1939	31.6122	32.8144	33.8486	35.2017	36.4194
25.3182	26.2872	27.6574	28.7345	30.1853	31.6054	32.8162	33.7937	35.1811	36.4245
25.3182	26.2872	27.6574	28.7345	30.1853	31.6054	32.8162	33.7937	35.1811	36.4228
25.3062	26.2786	27.6592	28.7722	30.1751	31.5968	32.8093	33.8211	35.1966	36.4177
25.3062	26.2786	27.6592	28.7722	30.1751	31.5968	32.8093	33.8211	35.1966	36.3679
25.2959	26.2786	27.6454	28.7602	30.2059	31.5676	32.8024	33.8246	35.2274	36.4005
25.2925	26.3043	27.63	28.7482	30.1991	31.5762	32.8179	33.8383	35.236	36.3645
25.2925	26.3043	27.63	28.7482	30.1991	31.5762	32.8179	33.8383	35.236	36.3508
25.2787	26.294	27.6386	28.7756	30.2368	31.5659	32.8196	33.8297	35.2377	36.3611
25.265	26.3026	27.6437	28.7619	30.2282	31.5899	32.8402	33.8537	35.2429	36.3473
25.265	26.3026	27.6437	28.7619	30.2282	31.5899	32.8402	33.8537	35.2429	36.3834
25.2582	26.3215	27.6471	28.7482	30.2488	31.6157	32.8024	33.8537	35.2652	36.3336
25.2633	26.3146	27.666	28.767	30.2419	31.6191	32.8436	33.852	35.2497	36.3696
25.2633	26.3146	27.666	28.767	30.2419	31.6191	32.8436	33.852	35.2497	36.3816
25.253	26.3026	27.642	28.7859	30.2282	31.6105	32.8453	33.8657	35.2515	36.3473
25.2461	26.3163	27.6386	28.7636	30.2419	31.6105	32.7973	33.8537	35.2326	36.3696
25.2461	26.3163	27.6386	28.7636	30.2419	31.6105	32.7973	33.8537	35.2326	36.3902
25.2496	26.3146	27.6351	28.7602	30.2231	31.6551	32.8179	33.8623	35.2257	36.3696
25.2496	26.3146	27.6351	28.7602	30.2231	31.6551	32.8179	33.8623	35.2257	36.3696
25.2513	26.3232	27.6317	28.7859	30.2128	31.6294	32.8093	33.8829	35.2395	36.3919
25.2444	26.3215	27.6574	28.7585	30.1991	31.6294	32.835	33.9103	35.1949	36.3696
25.2444	26.3215	27.6574	28.7585	30.1991	31.6294	32.835	33.9103	35.1949	36.3885
25.241	26.3335	27.6231	28.7756	30.2196	31.5796	32.835	33.8537	35.1966	36.3748
25.2427	26.3197	27.6283	28.7773	30.2282	31.5779	32.8162	33.8709	35.1571	36.3422
25.2427	26.3197	27.6283	28.7773	30.2282	31.5779	32.8162	33.8709	35.1571	36.3542
25.2393	26.3266	27.6334	28.7859	30.1905	31.5796	32.8144	33.876	35.1777	36.3388
25.2444	26.3232	27.6403	28.7945	30.2162	31.5934	32.8093	33.8795	35.1674	36.3542
25.2444	26.3232	27.6403	28.7945	30.2162	31.5934	32.8093	33.8795	35.1674	36.3679
25.253	26.3266	27.6574	28.7602	30.2094	31.5951	32.8076	33.8709	35.1863	36.3714

25.253	26.3266	27.6574	28.7602	30.2094	31.5951	32.8076	33.8709	35.1863	36.3336
25.2582	26.3215	27.6609	28.7293	30.1991	31.6122	32.7904	33.8623	35.1914	36.3422
25.2582	26.33	27.7106	28.7173	30.2076	31.5796	32.8179	33.8469	35.1674	36.3079
25.2582	26.33	27.7106	28.7173	30.2076	31.5796	32.8179	33.8469	35.1674	36.3096
25.2564	26.3266	27.714	28.7533	30.1853	31.6088	32.8093	33.8675	35.1811	36.3148
25.2599	26.3215	27.6814	28.7585	30.1991	31.6139	32.8299	33.8503	35.2	36.3268
25.2599	26.3215	27.6814	28.7585	30.1991	31.6139	32.8299	33.8503	35.2	36.3405
25.2564	26.3197	27.6883	28.7585	30.1888	31.5831	32.8144	33.7903	35.2189	36.2667
25.2564	26.3335	27.6969	28.7585	30.2145	31.5899	32.8264	33.8486	35.1829	36.2959
25.2564	26.3335	27.6969	28.7585	30.2145	31.5899	32.8264	33.8486	35.1829	36.301
25.2513	26.3077	27.7072	28.7808	30.2299	31.6002	32.8419	33.8657	35.2172	36.3062
25.241	26.3146	27.6952	28.779	30.2402	31.6157	32.8453	33.8503	35.2154	36.301
25.241	26.3146	27.6952	28.779	30.2402	31.6157	32.8453	33.8503	35.2154	36.2873
25.241	26.3437	27.6814	28.7945	30.2471	31.6071	32.8762	33.8743	35.2137	36.2976
25.241	26.3437	27.6814	28.7945	30.2471	31.6071	32.8762	33.8743	35.2137	36.3079
25.2341	26.3523	27.666	28.7773	30.2351	31.5762	32.8282	33.8486	35.224	36.313
25.2307	26.3352	27.6694	28.7602	30.2128	31.5762	32.8247	33.8503	35.1897	36.2907
25.2307	26.3352	27.6694	28.7602	30.2128	31.5762	32.8247	33.8503	35.1897	36.2856
25.2479	26.3112	27.6866	28.7808	30.2317	31.6105	32.823	33.8829	35.164	36.2873
25.2461	26.306	27.6986	28.7893	30.2402	31.6054	32.8007	33.9138	35.1931	36.3422
25.2461	26.306	27.6986	28.7893	30.2402	31.6054	32.8007	33.9138	35.1931	36.3079
25.2547	26.2957	27.7157	28.8185	30.2334	31.6191	32.823	33.8589	35.1829	36.3233
25.241	26.2769	27.7055	28.8048	30.2505	31.5865	32.8264	33.8572	35.1588	36.3028
25.241	26.2769	27.7055	28.8048	30.2505	31.5865	32.8264	33.8572	35.1588	36.2925
25.2324	26.2957	27.7329	28.7773	30.2231	31.5762	32.8042	33.8572	35.1554	36.2753
25.2324	26.2957	27.7329	28.7773	30.2231	31.5762	32.8042	33.8572	35.1554	36.2685
25.229	26.2872	27.7106	28.7928	30.2145	31.5831	32.7973	33.8863	35.1657	36.277
25.2359	26.3112	27.7072	28.7705	30.1991	31.6294	32.7904	33.9069	35.1811	36.2907
25.2359	26.3112	27.7072	28.7705	30.1991	31.6294	32.7904	33.9069	35.1811	36.2479
25.2239	26.3026	27.6969	28.7653	30.2111	31.5762	32.7561	33.9515	35.1743	36.2376
25.2256	26.2974	27.7192	28.7808	30.2196	31.5779	32.8076	33.9052	35.1606	36.265
25.2256	26.2974	27.7192	28.7808	30.2196	31.5779	32.8076	33.9052	35.1606	36.2633
25.2256	26.3455	27.7123	28.7619	30.2368	31.5951	32.7853	33.8966	35.2189	36.2787
25.2221	26.33	27.7209	28.7825	30.2248	31.5882	32.7904	33.912	35.2309	36.289
25.2221	26.33	27.7209	28.7825	30.2248	31.5882	32.7904	33.912	35.2309	36.2942
25.2204	26.3386	27.7278	28.7859	30.2179	31.5711	32.8042	33.8709	35.2257	36.2925
25.2187	26.3352	27.702	28.7808	30.2522	31.5642	32.8007	33.8692	35.2309	36.2616
25.2187	26.3352	27.702	28.7808	30.2522	31.5642	32.8007	33.8692	35.2309	36.2959
25.2118	26.3232	27.7055	28.7413	30.2437	31.5625	32.8213	33.8486	35.1966	36.3062
25.2118	26.3232	27.7055	28.7413	30.2437	31.5625	32.8213	33.8486	35.1966	36.2976
25.2033	26.3094	27.7072	28.7465	30.2111	31.5642	32.8042	33.8897	35.1863	36.2479
25.2067	26.2974	27.6832	28.7465	30.2059	31.5745	32.7939	33.9069	35.2103	36.2805
25.2067	26.2974	27.6832	28.7465	30.2059	31.5745	32.7939	33.9069	35.2103	36.2856

25.2153	26.3094	27.6729	28.7808	30.1922	31.5762	32.7819	33.8812	35.2429	36.2702
25.2067	26.318	27.6694	28.7619	30.1939	31.6071	32.763	33.8966	35.2017	36.2633
25.2067	26.318	27.6694	28.7619	30.1939	31.6071	32.763	33.8966	35.2017	36.2582
25.2136	26.3249	27.6883	28.767	30.2248	31.5625	32.7767	33.8692	35.1949	36.2582
25.2153	26.3317	27.6746	28.7893	30.2076	31.5419	32.7613	33.8897	35.2069	36.2787
25.2153	26.3317	27.6746	28.7893	30.2076	31.5419	32.7613	33.8897	35.2069	36.2616
25.2153	26.2974	27.7003	28.7825	30.2042	31.5762	32.7853	33.9258	35.236	36.2839
25.2153	26.2974	27.7003	28.7825	30.2042	31.5762	32.7853	33.9258	35.236	36.2427
25.2084	26.294	27.7192	28.7653	30.2042	31.5573	32.7819	33.9052	35.2103	36.1981
25.229	26.3283	27.738	28.7139	30.1836	31.5402	32.7921	33.9189	35.2	36.2479
25.229	26.3283	27.738	28.7139	30.1836	31.5402	32.7921	33.9189	35.2	36.2084
25.2307	26.3215	27.714	28.7002	30.1939	31.5522	32.7681	33.9035	35.1966	36.2444
25.229	26.3112	27.6952	28.7087	30.1956	31.5556	32.7836	33.8983	35.1794	36.229
25.229	26.3112	27.6952	28.7087	30.1956	31.5556	32.7836	33.8983	35.1794	36.2376
25.2341	26.3403	27.6574	28.7242	30.2008	31.5745	32.7578	33.8863	35.1691	36.2427
25.2427	26.3283	27.678	28.7482	30.2145	31.5522	32.7801	33.9035	35.1588	36.2496
25.2427	26.3283	27.678	28.7482	30.2145	31.5522	32.7801	33.9035	35.1588	36.2359
25.2256	26.3249	27.6729	28.743	30.2025	31.511	32.7613	33.8932	35.1657	36.2496
25.2256	26.3215	27.678	28.7585	30.1974	31.5282	32.763	33.8675	35.164	36.2359
25.2256	26.3215	27.678	28.7585	30.1974	31.5282	32.763	33.8675	35.164	36.253
25.2341	26.3129	27.6969	28.743	30.1888	31.523	32.7887	33.8469	35.1263	36.2462
25.2341	26.3129	27.6969	28.743	30.1888	31.523	32.7887	33.8469	35.1263	36.265
25.2359	26.3043	27.6917	28.7327	30.1853	31.5333	32.7887	33.8469	35.1331	36.2153
25.2273	26.3043	27.7192	28.7242	30.2231	31.5951	32.7458	33.8743	35.1177	36.1964
25.2273	26.3043	27.7192	28.7242	30.2231	31.5951	32.7458	33.8743	35.1177	36.1776
25.2393	26.318	27.6969	28.7362	30.2179	31.5608	32.7801	33.8623	35.1177	36.1656
25.2273	26.3215	27.6935	28.7293	30.2505	31.5556	32.7801	33.8777	35.1023	36.1638
25.2273	26.3215	27.6935	28.7293	30.2505	31.5556	32.7801	33.8777	35.1023	36.1535
25.2273	26.3163	27.7209	28.7293	30.2094	31.5814	32.787	33.8572	35.0971	36.2067
25.2307	26.3043	27.7072	28.7242	30.2076	31.5488	32.7956	33.8743	35.1554	36.1896
25.2307	26.3043	27.7072	28.7242	30.2076	31.5488	32.7956	33.8743	35.1554	36.2239
25.2359	26.282	27.7209	28.7087	30.1991	31.5762	32.7836	33.876	35.1091	36.2101
25.2359	26.282	27.7209	28.7087	30.1991	31.5762	32.7836	33.876	35.1091	36.241
25.2307	26.2769	27.7329	28.7002	30.2162	31.5762	32.7699	33.8572	35.1417	36.2599
25.2359	26.2906	27.6883	28.7293	30.2076	31.5556	32.775	33.9	35.152	36.2239
25.2359	26.2906	27.6883	28.7293	30.2076	31.5556	32.775	33.9	35.152	36.2187
25.2273	26.2992	27.6797	28.7293	30.1785	31.5505	32.7716	33.9035	35.1417	36.2256
25.2307	26.2992	27.6746	28.7259	30.1751	31.5916	32.763	33.8777	35.1486	36.2256
25.2307	26.2992	27.6746	28.7259	30.1751	31.5916	32.763	33.8777	35.1486	36.2307
25.2427	26.2923	27.6849	28.7688	30.1922	31.5694	32.7647	33.8589	35.1143	36.2256
25.253	26.2906	27.6729	28.7533	30.1974	31.5573	32.7527	33.8469	35.1211	36.2067
25.253	26.2906	27.6729	28.7533	30.1974	31.5573	32.7527	33.8469	35.1211	36.2273
25.2582	26.294	27.6643	28.7156	30.2162	31.5916	32.7578	33.8709	35.1468	36.2479

25.2633	26.2923	27.6557	28.7225	30.2179	31.5882	32.7836	33.8812	35.116	36.2702
25.2633	26.2923	27.6557	28.7225	30.2179	31.5882	32.7836	33.8812	35.116	36.2633
25.2496	26.2992	27.6454	28.7327	30.2385	31.5848	32.7373	33.8897	35.1417	36.2702
25.2496	26.2992	27.6454	28.7327	30.2385	31.5848	32.7373	33.8897	35.1417	36.2342
25.2684	26.294	27.6677	28.719	30.2419	31.5916	32.7699	33.8932	35.14	36.2582
25.2873	26.2872	27.6729	28.7104	30.2488	31.5865	32.7578	33.9035	35.1846	36.2719
25.2873	26.2872	27.6729	28.7104	30.2488	31.5865	32.7578	33.9035	35.1846	36.2496

Improved Height Results

Table 50 shows the improved height results for the last 1,500 time steps, or 30 seconds of real time, for the ten runs.

Table 50: Improved Height Results for Spillway Comparison

Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
24.76376	26.0275	27.24873	28.34758	29.75997	31.03939	32.23049	33.29655	34.59078	35.75957
24.77502	26.04043	27.25653	28.36495	29.75367	31.05335	32.23004	33.28187	34.58787	35.7428
24.77463	26.0533	27.22447	28.35703	29.74221	31.04864	32.23168	33.28196	34.59181	35.74538
24.76974	26.0378	27.22725	28.35927	29.74504	31.00165	32.23238	33.27208	34.58794	35.7555
24.76087	26.05268	27.23097	28.33749	29.74941	31.05453	32.23217	33.26697	34.58548	35.75384
24.77786	26.05305	27.25206	28.35954	29.74662	31.05668	32.22433	33.26673	34.58164	35.75266
24.76382	26.048	27.2529	28.34975	29.71876	31.05656	32.22237	33.28704	34.58293	35.74331
24.77228	26.0247	27.25056	28.35064	29.74257	31.05475	32.21969	33.28203	34.58899	35.75765
24.77538	25.99624	27.25455	28.35645	29.74344	31.05614	32.21572	33.28827	34.56514	35.74441
24.7721	26.05522	27.25027	28.34422	29.73786	31.03787	32.21127	33.28031	34.55179	35.74194
24.77364	26.04905	27.23296	28.33652	29.74466	31.03574	32.21786	33.26476	34.57496	35.76044
24.75431	26.03941	27.25207	28.3435	29.73064	31.05094	32.22167	33.26961	34.51979	35.76585
24.76485	26.03914	27.24741	28.34514	29.73527	31.02967	32.22438	33.28905	34.57261	35.73923
24.78422	26.00033	27.2406	28.35256	29.73589	31.0011	32.21838	33.28325	34.56336	35.76775
24.78614	26.04791	27.24953	28.31075	29.74694	31.05842	32.20163	33.29181	34.53896	35.77129
24.77777	26.04231	27.23966	28.32292	29.74266	31.06244	32.20005	33.29043	34.5646	35.73108
24.77886	26.03758	27.23119	28.31695	29.74331	31.06676	32.19889	33.27196	34.55225	35.75328
24.77661	26.03411	27.2431	28.34185	29.71295	31.06073	32.22464	33.26854	34.55934	35.77424
24.77705	26.00673	27.24254	28.34271	29.75114	31.05714	32.22528	33.28317	34.54997	35.73469
24.79872	26.01543	27.2261	28.33175	29.73274	31.06631	32.22515	33.26631	34.55456	35.76558
24.80076	26.02407	27.23999	28.34381	29.74954	31.04778	32.22496	33.24983	34.5604	35.74634
24.80188	26.03001	27.24099	28.33852	29.76597	31.06852	32.2245	33.25052	34.55953	35.7732
24.79505	26.02002	27.24492	28.32944	29.73907	31.071	32.22091	33.27492	34.55846	35.76875
24.79298	26.01091	27.24404	28.33272	29.68823	31.05297	32.21802	33.27206	34.54685	35.76929
24.80941	26.03077	27.24609	28.34101	29.76612	31.04761	32.21113	33.28715	34.53808	35.78856

24.80726	26.03966	27.24302	28.35099	29.77372	31.07292	32.21209	33.28473	34.55724	35.78866
24.80502	26.044	27.20464	28.33693	29.7793	31.07233	32.21503	33.27824	34.55848	35.78298
24.82027	26.03286	27.24256	28.33353	29.76184	31.07433	32.21684	33.27783	34.54518	35.78765
24.8241	26.03535	27.24236	28.31938	29.76805	31.07483	32.21837	33.28105	34.55212	35.79255
24.8075	26.03954	27.24709	28.26844	29.77095	31.06898	32.19716	33.26657	34.5522	35.77815
24.8262	26.04154	27.21864	28.33589	29.76349	31.02768	32.19535	33.27017	34.47136	35.77648
24.82084	26.02143	27.19884	28.33925	29.76856	31.05968	32.19027	33.27113	34.5595	35.74968
24.82273	26.04674	27.24041	28.33771	29.75939	31.06455	32.19562	33.24421	34.55776	35.78309
24.81867	26.03814	27.22615	28.32517	29.74092	31.06119	32.23217	33.26999	34.554	35.78984
24.80392	26.04189	27.22132	28.31197	29.7581	31.04682	32.23301	33.25235	34.55211	35.78608
24.8029	26.04155	27.24375	28.32969	29.74992	31.0694	32.23181	33.249	34.55053	35.74821
24.81332	26.04243	27.24908	28.32269	29.72868	31.06529	32.1846	33.24767	34.55338	35.78889
24.81994	26.0221	27.20863	28.31669	29.74525	31.03606	32.21761	33.24849	34.52264	35.76741
24.82277	26.0396	27.25113	28.31827	29.73537	31.05435	32.18171	33.26177	34.53361	35.77135
24.82051	25.98614	27.25126	28.33731	29.73127	31.03913	32.20893	33.26996	34.54302	35.78357
24.82175	26.02612	27.19488	28.31537	29.74184	31.06501	32.21498	33.27806	34.50046	35.7645
24.8213	26.00001	27.25553	28.32459	29.73834	31.06077	32.21764	33.27797	34.52461	35.76351
24.81663	26.02866	27.2285	28.31298	29.7436	31.05684	32.20848	33.27886	34.53909	35.77155
24.82326	26.0299	27.25601	28.33606	29.74595	31.05452	32.20713	33.27571	34.525	35.76059
24.82935	26.00998	27.25663	28.32555	29.74772	31.06308	32.19253	33.27728	34.54133	35.75812
24.84169	25.97464	27.24719	28.33478	29.73272	31.03092	32.1626	33.27266	34.52582	35.75204
24.83428	26.02538	27.20876	28.32881	29.74416	31.06728	32.20311	33.27727	34.5414	35.75835
24.83888	26.01705	27.20546	28.31023	29.73173	31.05131	32.22624	33.2763	34.4958	35.76846
24.84028	26.02502	27.25541	28.29793	29.75673	31.07041	32.22941	33.27466	34.50933	35.70112
24.84149	26.01271	27.26065	28.32532	29.71417	31.07324	32.23018	33.27623	34.53517	35.73545
24.84246	26.01827	27.24001	28.3034	29.75305	31.07896	32.19655	33.2727	34.52922	35.76609
24.84383	26.02796	27.25224	28.32077	29.75297	31.07244	32.20653	33.26676	34.53098	35.74308
24.84363	26.03535	27.25053	28.29501	29.75625	31.04449	32.20355	33.26727	34.51141	35.76509
24.84043	26.03085	27.26161	28.31742	29.75438	31.06785	32.20527	33.27735	34.53432	35.7607
24.83532	26.0188	27.26219	28.30881	29.74941	31.06848	32.21253	33.2526	34.5265	35.75798
24.86015	26.02159	27.24514	28.32093	29.73003	31.07294	32.21167	33.26353	34.50386	35.75266
24.85977	26.03412	27.23242	28.30318	29.75389	31.07365	32.22756	33.26795	34.52507	35.75618
24.85969	26.02643	27.26291	28.29659	29.74256	31.07546	32.22235	33.25919	34.48986	35.75664
24.84828	26.03126	27.25236	28.31737	29.75336	31.06352	32.22082	33.26039	34.50928	35.75311
24.84289	26.02429	27.25632	28.32121	29.7291	31.08077	32.22176	33.26539	34.50302	35.7497
24.84606	25.99229	27.25746	28.31701	29.74848	31.07549	32.22144	33.28172	34.51921	35.75581
24.86539	26.00408	27.2535	28.32201	29.74518	31.07677	32.22112	33.27835	34.51505	35.75465
24.87452	26.02958	27.253	28.28064	29.74132	31.06071	32.21927	33.27536	34.49861	35.76199
24.87684	26.02549	27.24353	28.32727	29.72945	31.06879	32.21858	33.27165	34.52444	35.75245
24.88702	26.03095	27.25418	28.32565	29.74614	31.06717	32.20033	33.2759	34.50622	35.7303
24.88647	26.02583	27.25269	28.28316	29.72749	31.07123	32.20026	33.2747	34.51199	35.72394
24.87462	26.02535	27.25666	28.31363	29.73052	31.0605	32.22358	33.24773	34.49509	35.76177
24.85217	26.02754	27.25123	28.31672	29.74948	31.0458	32.22359	33.2674	34.51065	35.74552

24.8828	26.0285	27.25208	28.30413	29.74908	31.07756	32.22607	33.27284	34.52093	35.76412
24.88927	26.02589	27.25539	28.32609	29.73938	31.07699	32.22406	33.271	34.51458	35.76018
24.87772	25.99951	27.25591	28.30356	29.75215	31.07072	32.22537	33.26879	34.50697	35.74179
24.84628	25.98523	27.24801	28.31057	29.75051	31.06513	32.22923	33.27019	34.52053	35.73831
24.88749	26.02346	27.23552	28.31717	29.75719	31.07336	32.22858	33.27221	34.52956	35.75536
24.88721	26.03329	27.25538	28.30958	29.73407	31.065	32.22429	33.25748	34.51854	35.7593
24.87853	26.02178	27.25583	28.30875	29.75353	31.07643	32.22673	33.2649	34.51578	35.73112
24.8859	26.02988	27.26102	28.3033	29.74566	31.07763	32.22716	33.26779	34.52321	35.7588
24.87623	26.03836	27.25966	28.28986	29.7536	31.07863	32.22706	33.27367	34.52143	35.74212
24.89552	26.03971	27.22801	28.28306	29.75189	31.07967	32.21386	33.27016	34.52235	35.74981
24.88172	26.03426	27.25706	28.29443	29.74665	31.07774	32.21461	33.27015	34.51757	35.74484
24.89452	26.02035	27.26021	28.25343	29.73308	31.07993	32.20944	33.26605	34.52646	35.75093
24.90545	26.02135	27.25386	28.29428	29.71945	31.07948	32.22998	33.26112	34.52707	35.7469
24.90147	26.04478	27.2439	28.29363	29.70129	31.08162	32.23359	33.27049	34.49957	35.7109
24.89394	26.04563	27.25933	28.30281	29.74402	31.08413	32.23494	33.24964	34.52667	35.72938
24.89544	26.03627	27.24733	28.29716	29.7402	31.08653	32.23524	33.25128	34.51399	35.74736
24.8825	26.04596	27.25591	28.28406	29.72076	31.06665	32.21042	33.26721	34.53089	35.73432
24.90986	26.04573	27.25234	28.31082	29.73877	31.08434	32.20882	33.24806	34.49383	35.72638
24.91426	26.01969	27.22448	28.29815	29.74009	31.08108	32.21844	33.2417	34.50813	35.73807
24.9161	26.04129	27.24851	28.29234	29.72932	31.08018	32.21698	33.23039	34.51265	35.73357
24.90631	26.03177	27.25191	28.29583	29.73984	31.07144	32.21516	33.25067	34.52466	35.68118
24.9019	26.01671	27.24869	28.30646	29.72811	31.06789	32.19688	33.24926	34.52478	35.73086
24.91872	26.02587	27.22101	28.30356	29.73408	31.06496	32.20259	33.25694	34.52053	35.72028
24.91697	26.02591	27.21818	28.31967	29.73874	31.06587	32.20336	33.26433	34.52487	35.72309
24.91705	26.03766	27.22945	28.29789	29.7084	31.07458	32.20172	33.26989	34.51293	35.70773
24.89148	26.02689	27.23432	28.30269	29.71276	31.03199	32.19843	33.27552	34.53103	35.72258
24.89579	26.04167	27.21874	28.27988	29.74847	31.06567	32.19953	33.26084	34.51712	35.70568
24.91775	26.03405	27.24777	28.31831	29.75167	31.07466	32.18581	33.27596	34.52547	35.70484
24.91747	26.03482	27.23202	28.31397	29.74829	31.07005	32.20293	33.27598	34.51331	35.70538
24.91856	26.03426	27.21931	28.32445	29.74865	31.07402	32.20366	33.26337	34.51391	35.70925
24.92469	26.03425	27.21329	28.31918	29.73761	31.07919	32.20516	33.23572	34.53285	35.68545
24.92364	26.02222	27.22132	28.31592	29.75114	31.07903	32.21861	33.24944	34.51222	35.70166
24.9163	26.02665	27.24288	28.2999	29.75187	31.07217	32.23165	33.26964	34.52458	35.70358
24.92277	26.05168	27.23687	28.30833	29.74851	31.0502	32.23306	33.27922	34.53329	35.69758
24.92246	26.01773	27.23757	28.32429	29.73925	31.04299	32.22577	33.27798	34.51138	35.68437
24.91463	26.03423	27.24265	28.31816	29.75823	31.04376	32.22863	33.26708	34.53202	35.69111
24.91604	25.9947	27.22975	28.30094	29.75168	31.0515	32.22308	33.26412	34.53073	35.69576
24.91369	26.04653	27.24201	28.3114	29.73607	31.06174	32.22321	33.25922	34.53323	35.69787
24.91031	26.04696	27.19472	28.32063	29.74644	31.08184	32.22258	33.27003	34.53556	35.68648
24.91159	26.0463	27.22896	28.31502	29.75255	31.0786	32.18791	33.27211	34.54853	35.69036
24.90339	26.03484	27.23266	28.3148	29.73565	31.06578	32.18977	33.25981	34.51727	35.67084
24.92663	25.98892	27.22745	28.31258	29.74095	31.07112	32.22653	33.2659	34.50842	35.68998
24.90973	26.0286	27.22988	28.32641	29.73453	31.06324	32.22538	33.25735	34.54788	35.69553

24.90184	26.01277	27.23917	28.28439	29.71766	31.07671	32.20565	33.25784	34.52971	35.70371
24.92112	26.05718	27.23737	28.3189	29.70998	31.07149	32.20753	33.26313	34.55082	35.68837
24.92672	26.05683	27.19779	28.32324	29.72723	31.06951	32.20693	33.25618	34.54727	35.6825
24.92631	26.05647	27.22742	28.32291	29.73159	31.06969	32.2252	33.25841	34.54011	35.70059
24.9255	26.0327	27.21281	28.32253	29.73842	31.06959	32.22291	33.25644	34.54154	35.70144
24.91673	26.04168	27.19165	28.3171	29.72848	31.07099	32.22268	33.26591	34.54438	35.678
24.9318	26.05618	27.2199	28.32298	29.73722	31.0182	32.22253	33.26574	34.53827	35.69068
24.90874	26.0401	27.22675	28.32179	29.74618	30.99229	32.22077	33.24527	34.52429	35.69309
24.9319	26.04536	27.22247	28.31619	29.69804	31.03056	32.19396	33.26461	34.52677	35.68144
24.93351	26.00886	27.21647	28.31397	29.74106	31.06385	32.18494	33.26728	34.54294	35.69563
24.92358	26.05563	27.19774	28.325	29.73608	31.07415	32.19791	33.26945	34.53103	35.68975
24.9264	26.04551	27.22222	28.30597	29.75325	31.07791	32.19907	33.26477	34.53314	35.6956
24.92371	26.0466	27.21685	28.31461	29.75194	31.0688	32.2015	33.26452	34.54124	35.69355
24.92302	26.04119	27.22193	28.32757	29.74873	31.06468	32.20188	33.24442	34.54406	35.67187
24.93584	26.05017	27.17796	28.31945	29.74808	31.06698	32.20817	33.26218	34.53046	35.69515
24.9358	26.04746	27.23533	28.31112	29.73284	31.07249	32.20986	33.26388	34.54014	35.69184
24.9349	26.00363	27.23188	28.27412	29.71208	31.05304	32.21233	33.26446	34.51024	35.68601
24.93225	25.99407	27.21278	28.30025	29.7473	31.04209	32.23081	33.25825	34.55707	35.67644
24.92807	26.05491	27.21471	28.30599	29.74637	31.04826	32.22771	33.25818	34.55352	35.68879
24.92875	26.04722	27.21849	28.30562	29.72642	31.0673	32.21375	33.27275	34.54818	35.67014
24.92296	26.02558	27.23311	28.3008	29.73941	31.07217	32.21183	33.28134	34.56602	35.66988
24.92271	26.01555	27.22739	28.31711	29.73569	31.06625	32.19685	33.27785	34.56616	35.66082
24.9336	26.05032	27.22506	28.31581	29.72631	31.03602	32.17776	33.27725	34.55605	35.67443
24.93255	26.04875	27.22343	28.31292	29.70957	31.04551	32.18521	33.27723	34.5702	35.6645
24.90271	26.04188	27.20728	28.30048	29.72832	31.06307	32.18324	33.27328	34.54492	35.65692
24.90976	26.03036	27.2311	28.27744	29.7172	31.06084	32.21201	33.27528	34.55891	35.65734
24.93497	26.04139	27.19862	28.30911	29.72672	31.06596	32.2011	33.26917	34.56108	35.67027
24.91201	26.02599	27.22066	28.31028	29.71657	31.06394	32.19944	33.27901	34.54382	35.68103
24.92599	26.04009	27.23631	28.30139	29.72961	31.02272	32.2008	33.26567	34.54787	35.67602
24.9241	26.02829	27.22915	28.30504	29.7178	31.04936	32.19899	33.26005	34.53297	35.67642
24.9313	26.00833	27.22864	28.31111	29.72495	31.05628	32.20846	33.26809	34.55621	35.64729
24.92434	26.02121	27.23658	28.3068	29.70588	31.04155	32.20859	33.26378	34.56772	35.67768
24.93686	26.0222	27.23665	28.31458	29.7214	31.05426	32.20738	33.26181	34.57027	35.67451
24.93767	26.04007	27.23445	28.30825	29.73667	31.0648	32.22021	33.25922	34.56841	35.6779
24.90744	26.03265	27.23219	28.29047	29.73944	31.05613	32.21851	33.25633	34.54523	35.67301
24.93501	26.027	27.24839	28.30748	29.72416	31.02358	32.21801	33.19347	34.53632	35.67109
24.93863	26.04326	27.24218	28.30181	29.72765	31.05531	32.21758	33.20709	34.56413	35.64766
24.94613	26.04102	27.24037	28.31022	29.7397	31.05421	32.21556	33.26409	34.57302	35.67713
24.93493	26.03744	27.24737	28.31358	29.72332	31.04716	32.2131	33.26379	34.56134	35.68183
24.94881	26.04096	27.24721	28.30205	29.73971	31.04629	32.20991	33.26304	34.56182	35.68384
24.95042	26.04767	27.2264	28.30761	29.72036	31.03476	32.20783	33.26006	34.55403	35.68181
24.95192	26.01701	27.23005	28.30497	29.7186	31.04395	32.20574	33.25526	34.57196	35.66873
24.94066	26.01454	27.24772	28.30874	29.74698	31.04947	32.20393	33.24599	34.57384	35.68663

24.94928	26.05147	27.25197	28.29382	29.7381	31.00385	32.20214	33.25847	34.54223	35.68541
24.94586	26.03411	27.24377	28.29826	29.7435	31.05603	32.20251	33.27803	34.57582	35.6607
24.93581	26.04244	27.23868	28.30728	29.74542	31.05351	32.20094	33.28329	34.56988	35.67957
24.94962	26.05048	27.26109	28.31215	29.7308	31.05108	32.17246	33.29095	34.52582	35.6802
24.93571	26.04868	27.24791	28.30005	29.74063	31.05246	32.19622	33.29067	34.55813	35.69123
24.94923	26.04561	27.26204	28.31556	29.7397	31.042	32.19703	33.28148	34.5573	35.67729
24.95519	26.05251	27.25898	28.31803	29.72823	31.05809	32.19886	33.28025	34.57856	35.68312
24.94171	26.04347	27.25698	28.30816	29.72608	31.05654	32.19908	33.25332	34.58188	35.68178
24.93825	26.03463	27.26387	28.31435	29.74111	31.06285	32.19999	33.26922	34.58519	35.6311
24.93921	26.03642	27.26153	28.31784	29.72664	31.06263	32.19948	33.26894	34.5811	35.68271
24.94738	26.05048	27.25398	28.30924	29.69795	31.06299	32.19268	33.27622	34.57613	35.68357
24.95674	26.05004	27.26448	28.30277	29.73811	31.05799	32.19548	33.28047	34.58444	35.6899
24.95891	26.05283	27.26442	28.31717	29.72192	31.06202	32.19693	33.28068	34.57152	35.69832
24.94111	26.04101	27.2563	28.31532	29.73498	31.05414	32.20143	33.26706	34.58521	35.67642
24.96368	26.04706	27.25761	28.30556	29.74411	31.04944	32.20375	33.26415	34.5743	35.66014
24.96587	26.04229	27.26526	28.31772	29.7438	31.05326	32.20487	33.25921	34.58583	35.70562
24.96613	26.05696	27.24896	28.30113	29.74311	31.0538	32.20639	33.25954	34.57738	35.69303
24.96637	26.04098	27.26633	28.30259	29.74118	31.05197	32.20639	33.27408	34.58351	35.69641
24.9041	26.06175	27.26041	28.29897	29.75273	31.04109	32.20663	33.27553	34.58267	35.70373
24.96864	26.06385	27.24506	28.30012	29.72056	31.05812	32.2038	33.25419	34.57001	35.70934
24.93623	26.06142	27.26088	28.28494	29.75175	31.05768	32.19115	33.22563	34.57537	35.7106
24.93735	26.05299	27.26585	28.29609	29.74819	31.04643	32.18905	33.22163	34.58016	35.6971
24.97154	26.0558	27.26539	28.31239	29.74731	31.03658	32.19551	33.25713	34.57657	35.69485
24.9631	26.0418	27.26505	28.30138	29.74963	31.05977	32.19314	33.26807	34.56243	35.71475
24.96787	26.06493	27.27554	28.29502	29.72382	31.05181	32.19248	33.26633	34.57701	35.70621
24.94778	26.06174	27.23062	28.30609	29.7491	31.0245	32.18774	33.26262	34.5776	35.70785
24.96076	26.05694	27.28662	28.31536	29.72316	31.05037	32.18693	33.27008	34.56423	35.71274
24.95962	26.04461	27.27633	28.30652	29.75458	31.04537	32.18513	33.26155	34.56638	35.71649
24.95143	26.04427	27.28146	28.3103	29.7424	31.03936	32.1753	33.26433	34.58502	35.69286
24.92416	26.05975	27.23289	28.30255	29.73725	31.05639	32.175	33.27415	34.54135	35.72402
24.92302	26.06054	27.28313	28.3038	29.73408	31.05434	32.17515	33.27805	34.57996	35.72036
24.96009	26.06133	27.2871	28.21778	29.74268	31.0627	32.18308	33.27821	34.57924	35.72519
24.9707	26.05367	27.28568	28.31145	29.74142	31.05407	32.18434	33.25974	34.56769	35.70833
24.97447	26.04342	27.28133	28.31354	29.73171	31.05465	32.18861	33.26169	34.5866	35.73127
24.97613	26.05698	27.27919	28.28907	29.73263	31.05088	32.15687	33.26268	34.58348	35.72178
24.978	26.03904	27.26839	28.30311	29.73851	31.05348	32.15902	33.2771	34.58105	35.72568
24.96391	26.05446	27.28047	28.30294	29.72522	31.06283	32.20477	33.27549	34.57994	35.74222
24.97741	26.04552	27.27947	28.30128	29.73955	31.06441	32.20427	33.28344	34.57839	35.73749
24.98683	26.04553	27.27294	28.31369	29.74185	31.05385	32.20462	33.28499	34.58146	35.73769
24.99031	26.05441	27.27869	28.29296	29.74324	31.0515	32.18039	33.26631	34.59368	35.73495
24.99185	26.05771	27.27796	28.30495	29.75008	31.06322	32.1852	33.26416	34.56946	35.7269
24.99182	26.04806	27.25892	28.28204	29.71258	31.05932	32.18564	33.25438	34.59011	35.75278
24.99367	26.05676	27.26685	28.22532	29.75994	31.05944	32.18687	33.28245	34.5715	35.74366

24.95553	26.04994	27.25994	28.30628	29.75957	31.05153	32.19404	33.28198	34.57432	35.7243
24.9754	26.02827	27.26675	28.26768	29.76008	31.06719	32.19565	33.27884	34.57181	35.74424
24.98903	26.05117	27.27565	28.30671	29.74958	31.06416	32.19921	33.27745	34.57721	35.69365
24.99404	26.04461	27.26517	28.29646	29.71525	31.06263	32.20083	33.27925	34.57608	35.73864
24.99702	25.99221	27.2585	28.319	29.75647	31.03639	32.20239	33.27359	34.57853	35.75722
24.9904	26.04681	27.27187	28.29536	29.74607	31.06328	32.19658	33.25646	34.55368	35.72302
24.99645	26.03759	27.27018	28.31817	29.76374	31.06469	32.19742	33.27565	34.53693	35.75966
24.99973	26.04092	27.24198	28.31265	29.76218	31.06524	32.19929	33.27263	34.55138	35.76168
24.99641	26.04466	27.269	28.31864	29.75314	31.05728	32.19853	33.26164	34.57274	35.75982
25.00046	26.03649	27.26804	28.32459	29.75951	31.05434	32.21799	33.26769	34.56562	35.73684
24.9992	26.03719	27.25398	28.32328	29.74853	31.06465	32.21992	33.26648	34.55351	35.76108
25.00009	26.04829	27.25867	28.32026	29.74992	31.06197	32.22126	33.26972	34.56098	35.76677
24.96969	26.0498	27.2658	28.31996	29.72575	31.03534	32.21512	33.26851	34.56754	35.7751
25.00212	26.04496	27.26833	28.31891	29.75168	31.05569	32.20971	33.2575	34.58064	35.7685
24.96792	26.04834	27.24775	28.30103	29.75034	31.01366	32.20828	33.27982	34.57778	35.77514
24.99111	26.04886	27.22111	28.31991	29.72093	31.05221	32.20787	33.25371	34.58221	35.74794
24.99265	26.03781	27.2661	28.31489	29.75088	31.06021	32.20526	33.26845	34.56852	35.76265
24.98471	26.03934	27.26149	28.32827	29.73456	31.06214	32.2042	33.26897	34.57298	35.78148
24.98536	26.03445	27.26987	28.32511	29.75018	31.06461	32.20167	33.28893	34.57004	35.77101
24.98127	26.03412	27.24756	28.29999	29.74221	31.05015	32.20609	33.28457	34.58004	35.77809
24.98438	26.01342	27.22776	28.32187	29.73428	31.03972	32.20584	33.26317	34.58267	35.78462
24.9858	26.05147	27.26426	28.3277	29.75685	31.04751	32.20474	33.27366	34.55528	35.7764
24.99614	26.05146	27.26642	28.31712	29.73734	31.0562	32.20003	33.28276	34.56276	35.77552
25.00394	26.0295	27.26531	28.32059	29.7439	31.06729	32.19971	33.2887	34.56971	35.77185
24.9748	26.0422	27.26046	28.3353	29.74709	31.0706	32.20003	33.2843	34.57728	35.77932
24.97553	26.01624	27.21818	28.3376	29.76938	31.07528	32.20011	33.27369	34.55496	35.79349
24.98405	26.03289	27.25725	28.34016	29.76752	31.07752	32.20163	33.27556	34.58746	35.79635
25.00738	26.04086	27.25263	28.33244	29.77639	31.07125	32.20415	33.25334	34.59213	35.78921
25.00064	26.03866	27.23458	28.34319	29.77346	31.07761	32.20621	33.22627	34.58754	35.7644
24.99866	26.0312	27.24501	28.33969	29.74875	31.07205	32.2059	33.28242	34.5747	35.79905
25.00231	26.02	27.25989	28.34346	29.77934	31.05139	32.20682	33.27742	34.57366	35.81154
25.00158	26.05063	27.26372	28.32555	29.76419	31.07755	32.21153	33.26165	34.58242	35.81483
24.99598	26.04517	27.23742	28.33231	29.74366	31.06009	32.19196	33.27579	34.58201	35.80942
24.99398	26.03455	27.23082	28.31493	29.76305	31.06903	32.19164	33.27281	34.57993	35.81817
24.99203	26.04229	27.23407	28.33609	29.77242	31.07472	32.16104	33.27868	34.57455	35.79844
24.99029	26.04703	27.24486	28.33945	29.76752	31.07239	32.20607	33.27861	34.55735	35.81503
24.98756	26.04299	27.25514	28.34201	29.76772	31.07144	32.20643	33.2535	34.56603	35.81429
24.96467	26.03639	27.25826	28.33675	29.74012	31.07064	32.19987	33.2536	34.56265	35.81118
24.97966	26.03232	27.25342	28.34154	29.76013	31.06299	32.14443	33.27804	34.55931	35.81161
24.97422	26.04704	27.22916	28.34705	29.73695	31.04803	32.213	33.27651	34.55646	35.78798
24.94267	26.03033	27.2539	28.33274	29.75612	31.05328	32.22229	33.26081	34.56561	35.821
24.98665	26.03974	27.23976	28.33661	29.74761	31.0729	32.22288	33.26159	34.54837	35.81121
24.98564	26.04251	27.24614	28.3364	29.75987	31.075	32.22284	33.27704	34.56439	35.81564

24.98079	26.04516	27.2405	28.3238	29.75932	31.06572	32.22276	33.27674	34.56272	35.82547
24.98171	26.04076	27.23458	28.34433	29.73455	31.06948	32.22354	33.27708	34.56488	35.80899
24.9852	26.03747	27.24262	28.33497	29.7558	31.06626	32.22443	33.26969	34.56755	35.82994
24.98158	26.02816	27.23879	28.347	29.76021	31.06668	32.22424	33.27265	34.56094	35.83243
24.9653	26.02019	27.22648	28.34296	29.76623	31.06174	32.22637	33.26041	34.56955	35.79343
24.97787	26.00531	27.23748	28.32002	29.75254	31.07474	32.22684	33.26025	34.5555	35.83823
24.97803	26.02701	27.20397	28.33511	29.76639	31.06645	32.22654	33.26684	34.53367	35.83098
24.97399	26.0198	27.23291	28.31351	29.75524	31.0731	32.21567	33.26989	34.55792	35.84359
24.97352	26.01644	27.23561	28.34697	29.77614	31.06812	32.21409	33.26858	34.5337	35.84153
24.96378	26.01587	27.23599	28.3444	29.77056	31.06836	32.21223	33.24437	34.55748	35.83771
24.96991	26.01627	27.23187	28.33533	29.75996	31.07603	32.21935	33.25994	34.55731	35.84958
24.92108	25.97829	27.23246	28.33272	29.77357	31.07181	32.21694	33.25795	34.54535	35.84389
24.92748	26.01915	27.22314	28.34134	29.76165	31.07479	32.21544	33.25809	34.54261	35.84042
24.95458	26.01953	27.23343	28.33924	29.77397	31.07016	32.21298	33.26551	34.54022	35.80828
24.95266	26.01021	27.22917	28.34777	29.77526	31.06067	32.21111	33.26388	34.54687	35.85559
24.96083	26.01152	27.22662	28.33514	29.76109	31.08408	32.1935	33.25193	34.54824	35.84952
24.95695	26.01853	27.19201	28.30941	29.77323	31.08234	32.19305	33.24506	34.53841	35.84703
24.93916	25.99731	27.21548	28.34589	29.76666	31.08498	32.20283	33.25871	34.52393	35.84874
24.94785	26.00328	27.22457	28.34526	29.76361	31.07992	32.20051	33.25863	34.51852	35.86087
24.94677	26.02712	27.21331	28.34112	29.75936	31.08562	32.19663	33.2277	34.54254	35.8539
24.91916	26.02892	27.22847	28.3483	29.71786	31.08679	32.19594	33.24003	34.54033	35.83107
24.93996	26.03062	27.22231	28.3274	29.7573	31.06167	32.19608	33.26846	34.53883	35.86054
24.94656	26.01847	27.20891	28.33623	29.71299	31.08264	32.16662	33.26431	34.52707	35.86398
24.95205	26.0293	27.22933	28.34156	29.75384	31.07951	32.21102	33.26162	34.52731	35.86358
24.94703	26.02823	27.22122	28.30552	29.75209	31.02675	32.21203	33.25692	34.52344	35.8469
24.95249	26.02367	27.21082	28.34771	29.72872	31.07894	32.21371	33.26409	34.53016	35.86175
24.94636	26.02881	27.22348	28.32068	29.74389	31.07876	32.19726	33.25505	34.53629	35.86879
24.9439	26.02253	27.22245	28.34459	29.7436	31.07881	32.19784	33.24303	34.53403	35.87513
24.9297	26.02388	27.22801	28.34482	29.76078	31.07817	32.19657	33.26158	34.52727	35.87815
24.91308	26.00899	27.22023	28.33959	29.75244	31.06293	32.19926	33.2597	34.54227	35.86577
24.90641	26.01711	27.23164	28.32695	29.76472	31.06345	32.19632	33.25249	34.53667	35.87929
24.86778	26.00264	27.22579	28.33384	29.76742	31.06666	32.18588	33.25151	34.54322	35.87458
24.86699	26.02128	27.22473	28.30638	29.74205	31.06126	32.18623	33.25074	34.5477	35.87318
24.92618	26.01746	27.2358	28.34107	29.76504	31.07126	32.1868	33.25204	34.5129	35.8786
24.92468	26.0082	27.23911	28.33915	29.76032	31.06178	32.21919	33.25602	34.54014	35.87974
24.91106	26.02197	27.22437	28.32787	29.77646	31.04831	32.22139	33.25297	34.52158	35.87492
24.90942	25.99058	27.23389	28.33727	29.78068	31.06116	32.2226	33.26009	34.5465	35.87913
24.91745	26.02327	27.23411	28.32157	29.756	31.08247	32.22276	33.26164	34.5438	35.87187
24.91236	25.99615	27.22797	28.33337	29.76402	31.05296	32.22306	33.25667	34.5355	35.87282
24.8937	26.0227	27.22393	28.33145	29.77468	31.08041	32.22721	33.25271	34.5455	35.83384
24.88983	25.98636	27.23539	28.31347	29.76496	31.08728	32.22835	33.25055	34.52587	35.87681
24.89811	26.02808	27.22212	28.29985	29.76867	31.09196	32.23197	33.25354	34.53149	35.86418
24.89134	26.03635	27.23285	28.3228	29.75856	31.08349	32.23226	33.23034	34.53077	35.86621

24.88846	26.02372	27.21353	28.32882	29.76332	31.08567	32.2201	33.23963	34.53358	35.85527
24.87457	26.02736	27.21776	28.31795	29.75607	31.0707	32.22171	33.24065	34.53417	35.86691
24.88269	26.00146	27.20814	28.31225	29.75671	31.05397	32.19294	33.24613	34.51773	35.85831
24.87164	26.00543	27.22346	28.33385	29.7422	31.04924	32.19188	33.24027	34.52749	35.86744
24.88422	26.03839	27.23226	28.32253	29.74217	31.09221	32.20598	33.232	34.51832	35.8691
24.88224	26.02721	27.2108	28.29144	29.74066	31.09501	32.22865	33.23733	34.53095	35.86484
24.87911	26.01968	27.23082	28.29904	29.75493	31.08273	32.22924	33.21509	34.52427	35.86396
24.87711	26.02238	27.23261	28.33323	29.73861	31.08629	32.23344	33.24423	34.52281	35.862
24.87709	25.98732	27.23006	28.32657	29.74701	31.08505	32.23622	33.25192	34.52283	35.87651
24.87588	26.03324	27.21721	28.30693	29.75428	31.08582	32.2365	33.24731	34.51375	35.87464
24.87131	26.02768	27.23931	28.32219	29.74126	31.08702	32.23716	33.2386	34.51897	35.84047
24.86263	26.03081	27.23549	28.33346	29.75713	31.08326	32.23563	33.25163	34.50339	35.87366
24.85833	26.02229	27.2345	28.32502	29.75988	31.07042	32.19458	33.25721	34.51708	35.86993
24.85388	26.01854	27.22578	28.32908	29.75588	31.07869	32.17102	33.25805	34.5064	35.87712
24.85077	26.01122	27.20223	28.32629	29.75835	31.07942	32.22829	33.26248	34.51341	35.87006
24.84525	25.98081	27.22688	28.30442	29.76809	31.08532	32.22675	33.2485	34.50446	35.87696
24.84586	26.02695	27.20778	28.31524	29.75688	31.06055	32.22947	33.26377	34.51729	35.86949
24.84539	26.01773	27.22036	28.32354	29.72564	31.054	32.23018	33.22155	34.51804	35.86014
24.83861	26.00611	27.2453	28.32102	29.73933	31.08634	32.23297	33.22382	34.50139	35.85245
24.83132	26.01675	27.24459	28.31995	29.74162	31.08376	32.23465	33.22178	34.517	35.85814
24.81647	26.02065	27.23962	28.2792	29.7688	31.06041	32.23538	33.23828	34.51311	35.86073
24.8193	26.00949	27.23019	28.31571	29.71947	31.07729	32.23276	33.25548	34.50963	35.85814
24.81831	26.01959	27.24186	28.29159	29.75656	31.08036	32.23383	33.24474	34.51925	35.8537
24.83087	26.03176	27.22037	28.30359	29.75334	31.07968	32.23338	33.23753	34.50709	35.85295
24.82782	26.02822	27.23926	28.31248	29.7414	31.0675	32.24123	33.25005	34.5252	35.8383
24.79637	26.04168	27.24143	28.28941	29.7494	31.07774	32.24244	33.26138	34.5252	35.85477
24.78167	26.00803	27.24089	28.30901	29.74941	31.06724	32.24418	33.26023	34.52716	35.84799
24.77598	26.02424	27.24998	28.3144	29.73965	31.05913	32.24381	33.26153	34.52017	35.81148
24.81041	26.04076	27.24394	28.31056	29.7375	31.07806	32.24445	33.26163	34.52469	35.8554
24.80542	26.03984	27.2369	28.30178	29.74822	31.07595	32.2427	33.2618	34.48615	35.84652
24.80584	26.04541	27.24288	28.31703	29.72193	31.07907	32.24107	33.2661	34.52015	35.8424
24.80826	26.03722	27.23824	28.29595	29.75002	31.08266	32.22232	33.21535	34.52817	35.84711
24.79855	26.04439	27.22018	28.29137	29.7472	31.07467	32.19799	33.26588	34.50322	35.83737
24.80449	26.03264	27.23671	28.31434	29.74132	31.06979	32.19001	33.24399	34.52292	35.83256
24.80312	26.04156	27.23323	28.30644	29.73492	31.08976	32.20972	33.24938	34.52464	35.81759
24.79486	26.05065	27.22935	28.24601	29.74986	31.07305	32.21598	33.25039	34.53222	35.82407
24.79754	26.04432	27.21718	28.31439	29.74675	31.08354	32.23589	33.26259	34.53204	35.7995
24.79428	26.04785	27.23544	28.31885	29.75276	31.07978	32.23516	33.2617	34.52505	35.83411
24.79498	26.04792	27.22821	28.305	29.75122	31.07527	32.23178	33.25328	34.51189	35.82671
24.78822	26.0393	27.24509	28.30086	29.75761	31.08089	32.23215	33.25186	34.53073	35.81924
24.77672	26.03695	27.24727	28.32715	29.75021	31.01673	32.22988	33.24998	34.52661	35.82859
24.78142	26.0541	27.24084	28.32088	29.73999	31.04844	32.23379	33.23485	34.50919	35.81339
24.78161	26.05263	27.25439	28.33519	29.7585	31.05721	32.22369	33.23682	34.5083	35.8241

24.78124	26.05504	27.25897	28.32267	29.75193	31.076	32.22666	33.23168	34.50654	35.80726
24.78517	26.04098	27.23507	28.32477	29.74479	31.07723	32.22681	33.24454	34.52261	35.82778
24.78446	26.04841	27.25521	28.33331	29.73494	31.07375	32.22729	33.25234	34.51038	35.81336
24.77352	26.06027	27.25418	28.29869	29.74803	31.07533	32.22743	33.24727	34.48724	35.82478
24.77064	26.04525	27.26355	28.32182	29.73384	31.063	32.2355	33.24488	34.52833	35.81642
24.76983	26.04951	27.23511	28.32542	29.75578	31.07453	32.23515	33.25289	34.48586	35.76456
24.75358	26.05737	27.26314	28.32139	29.71097	31.07504	32.23419	33.25048	34.52588	35.81546
24.77408	26.06089	27.25765	28.30104	29.74584	31.0689	32.23171	33.24935	34.53341	35.80672
24.74389	26.06083	27.25907	28.32156	29.74815	31.07017	32.23117	33.25159	34.52741	35.80165
24.75269	26.06139	27.25534	28.30519	29.73966	31.07633	32.2292	33.25346	34.5399	35.81336
24.76985	26.05894	27.25113	28.31396	29.73824	31.07782	32.2175	33.21975	34.53346	35.80177
24.77052	26.04147	27.25687	28.32117	29.73888	31.06708	32.21888	33.25013	34.53794	35.75125
24.76992	26.04461	27.25691	28.31549	29.73694	31.06414	32.22077	33.25303	34.54524	35.77681
24.76914	26.05513	27.25608	28.31946	29.74481	31.08403	32.22199	33.22287	34.54156	35.79241
24.76951	26.0566	27.19398	28.30856	29.74003	31.07431	32.22815	33.25292	34.53865	35.78767
24.7677	26.05028	27.24706	28.27909	29.71559	31.08284	32.22703	33.25552	34.55037	35.78187
24.75755	26.0315	27.24484	28.31296	29.74174	31.05529	32.22648	33.24756	34.55156	35.78718
24.76104	26.05691	27.21921	28.31076	29.73647	31.07442	32.21948	33.2467	34.53864	35.78109
24.76218	26.05566	27.24562	28.26958	29.737	31.05809	32.22523	33.25403	34.55019	35.78363
24.7628	26.04305	27.24626	28.29226	29.73353	31.0691	32.22671	33.25155	34.5513	35.78987
24.75831	26.0513	27.24811	28.31868	29.72567	31.08437	32.22646	33.25212	34.55385	35.77395
24.75942	26.05012	27.23799	28.31613	29.71859	31.07953	32.21145	33.2424	34.55027	35.72061
24.75381	26.03714	27.24365	28.30991	29.69864	31.07996	32.21169	33.2521	34.54507	35.77473
24.73836	26.04803	27.25184	28.31152	29.72138	31.07564	32.20959	33.23644	34.54807	35.77795
24.73882	26.05781	27.23863	28.31975	29.7451	31.08948	32.21106	33.22024	34.54977	35.76753
24.75553	26.05885	27.25146	28.31477	29.74764	31.08865	32.22419	33.24501	34.53391	35.78087
24.75633	26.04827	27.25498	28.28824	29.74624	31.08653	32.22491	33.2535	34.54553	35.78247
24.75526	26.05664	27.25417	28.2965	29.74622	31.08122	32.22231	33.25104	34.54417	35.76612
24.72906	26.06257	27.23729	28.30257	29.72721	31.06884	32.22184	33.22065	34.55569	35.77193
24.72841	26.03834	27.24895	28.32252	29.7348	31.07131	32.21838	33.2211	34.54412	35.78581
24.73753	26.06391	27.2503	28.31447	29.74506	31.07849	32.21729	33.25941	34.53524	35.78118
24.75804	26.05673	27.2532	28.32214	29.73563	31.07717	32.21017	33.25743	34.5324	35.7772
24.77378	26.05132	27.25099	28.32448	29.73763	31.03395	32.21253	33.25876	34.56492	35.77928
24.77334	26.06536	27.2489	28.29847	29.70008	31.07913	32.21126	33.26133	34.56763	35.78387
24.77568	26.05085	27.26021	28.32556	29.71375	31.07081	32.2116	33.25294	34.55035	35.76053
24.76076	26.06134	27.25859	28.31421	29.72302	31.07937	32.21142	33.25328	34.571	35.77398
24.77072	26.05377	27.26608	28.31914	29.72772	31.07584	32.20739	33.25436	34.57458	35.78842
24.76973	26.06061	27.25264	28.32421	29.73723	31.06772	32.20685	33.25887	34.56001	35.77164
24.73853	26.06016	27.25888	28.30341	29.72438	31.04477	32.20729	33.2629	34.57325	35.77837
24.75772	26.04731	27.24781	28.29359	29.71583	31.0237	32.20073	33.26534	34.56888	35.75727
24.77475	26.04499	27.25781	28.32278	29.70162	31.06706	32.20219	33.26241	34.56727	35.7723
24.77581	26.05201	27.25669	28.31017	29.70677	31.06439	32.20113	33.25589	34.56509	35.76779
24.77687	26.05724	27.26146	28.30287	29.67601	31.06229	32.20202	33.23844	34.56852	35.75962

24.7647	26.05607	27.25116	28.31415	29.73436	31.05446	32.2013	33.2377	34.58803	35.77378
24.76509	26.05444	27.22349	28.28451	29.736	31.07666	32.20265	33.23775	34.57386	35.74644
24.77865	26.03281	27.2466	28.32143	29.71127	31.07685	32.20307	33.25381	34.57445	35.76651
24.78182	26.02051	27.26099	28.31683	29.74498	31.06839	32.20324	33.25866	34.57929	35.75691
24.78166	26.03991	27.25201	28.31495	29.74945	31.06503	32.21325	33.25333	34.59637	35.76147
24.78224	26.05336	27.26207	28.32048	29.73859	31.06455	32.21531	33.26117	34.5953	35.74524
24.78259	26.04954	27.23411	28.32085	29.75161	31.05966	32.21728	33.25965	34.59397	35.75114
24.77821	26.05466	27.26243	28.30786	29.73061	31.06282	32.21766	33.24745	34.55989	35.75967
24.78256	26.0467	27.26363	28.31992	29.75316	31.06076	32.21504	33.23631	34.59003	35.75371
24.79197	26.0502	27.27252	28.32562	29.75707	31.05543	32.18383	33.25851	34.592	35.74959
24.78799	26.02442	27.26144	28.3066	29.74013	31.06007	32.21162	33.23258	34.60084	35.74325
24.79832	26.04326	27.26231	28.32905	29.73141	31.0811	32.18681	33.25149	34.59567	35.75291
24.79966	26.02563	27.24624	28.32176	29.75076	31.07922	32.20848	33.24818	34.59593	35.75067
24.78538	26.04939	27.24625	28.31433	29.74096	31.0706	32.21045	33.2614	34.59892	35.75321
24.78176	26.04085	27.27372	28.32913	29.73917	31.04787	32.21001	33.2269	34.58848	35.74957
24.7836	26.04493	27.27815	28.28312	29.74043	31.08389	32.20873	33.25444	34.59894	35.75253
24.79921	26.02586	27.26356	28.31008	29.73769	31.07793	32.20854	33.26125	34.61158	35.76007
24.79785	26.04361	27.27478	28.31415	29.72002	31.04638	32.2082	33.26169	34.61743	35.76241
24.79561	26.0301	27.26764	28.32677	29.68471	31.08189	32.20838	33.26287	34.6174	35.7435
24.79375	26.04005	27.26099	28.26354	29.73755	31.07722	32.20791	33.26346	34.60917	35.74391
24.79801	26.04032	27.26418	28.31754	29.7383	31.03446	32.20833	33.25898	34.61164	35.76176
24.7611	26.02726	27.26492	28.32433	29.73048	31.04851	32.20852	33.25063	34.61706	35.74232
24.76187	26.0231	27.24941	28.32028	29.71624	31.07585	32.20426	33.25617	34.62483	35.72042
24.80075	26.01637	27.27222	28.31366	29.7348	31.07086	32.20602	33.25674	34.6129	35.76612
24.79043	26.02854	27.28081	28.30959	29.74053	31.06703	32.20623	33.24359	34.62401	35.75187
24.79869	26.03796	27.27991	28.30177	29.74716	31.07288	32.20561	33.25234	34.62069	35.72213
24.81225	26.02903	27.23548	28.31562	29.74871	31.06811	32.20489	33.2496	34.6308	35.76243
24.76983	26.03663	27.28237	28.30362	29.74052	31.06017	32.20483	33.27664	34.62193	35.76268
24.80045	25.97429	27.24826	28.30039	29.74889	31.05968	32.19326	33.27616	34.62449	35.75625
24.80179	26.04171	27.27855	28.30479	29.75466	31.0599	32.19785	33.25512	34.6351	35.75914
24.82229	26.03895	27.2814	28.25019	29.76399	31.06756	32.19944	33.27509	34.62182	35.74218
24.82541	26.02984	27.28131	28.30908	29.76365	31.03463	32.20044	33.27635	34.61999	35.73945
24.82612	26.04727	27.26738	28.29973	29.75055	31.05709	32.21842	33.27883	34.6469	35.7623
24.74934	26.02572	27.27005	28.29897	29.74592	31.0666	32.21887	33.27593	34.64529	35.71407
24.82057	26.04616	27.27615	28.30208	29.74696	31.06834	32.19459	33.27912	34.63336	35.75347
24.82112	26.02743	27.26421	28.30763	29.76361	31.06867	32.20076	33.27872	34.63247	35.74882
24.81441	26.02421	27.26269	28.27703	29.75892	31.06656	32.20228	33.27529	34.6347	35.75101
24.81259	26.03974	27.27566	28.29821	29.76076	31.07045	32.2027	33.27899	34.64292	35.75815
24.81406	26.04741	27.27006	28.30397	29.73374	31.05304	32.2017	33.23736	34.6262	35.74939
24.81251	26.04454	27.27918	28.29922	29.76103	31.05985	32.2065	33.26806	34.62508	35.74885
24.82116	26.03221	27.25335	28.30139	29.7563	31.06666	32.20637	33.267	34.62462	35.74781
24.8289	26.03408	27.26831	28.3044	29.7482	31.02647	32.20561	33.26827	34.63436	35.7567
24.83187	26.02807	27.27507	28.28963	29.74943	31.06592	32.20446	33.26867	34.63394	35.74915

24.83234	26.03411	27.27686	28.3044	29.75177	31.07109	32.21362	33.26543	34.61785	35.75114
24.83553	26.00871	27.27338	28.30174	29.75364	31.07236	32.21291	33.27135	34.6324	35.74233
24.83487	26.02835	27.27621	28.27071	29.75374	31.05683	32.19569	33.27099	34.62364	35.75073
24.8368	26.02435	27.28018	28.29299	29.74863	31.06341	32.21252	33.26975	34.6482	35.74646
24.83394	25.99992	27.25736	28.2991	29.75499	31.07168	32.21103	33.26852	34.63137	35.75064
24.83615	26.02604	27.2782	28.30052	29.73833	31.05772	32.21071	33.26693	34.62792	35.74726
24.83209	26.03205	27.28537	28.28599	29.74692	31.07633	32.20388	33.25936	34.64737	35.73772
24.83355	26.02321	27.27447	28.27054	29.73736	31.08326	32.20373	33.27437	34.65416	35.74332
24.83541	26.01585	27.28062	28.30011	29.7598	31.08152	32.20525	33.26219	34.65543	35.73575
24.83541	26.028	27.27973	28.28057	29.77075	31.07923	32.21376	33.27125	34.64648	35.72751
24.83696	25.97486	27.2772	28.27694	29.76696	31.0782	32.21604	33.27256	34.65182	35.69079
24.8186	25.9872	27.26658	28.28974	29.75764	31.0803	32.21834	33.26042	34.65663	35.74478
24.81905	26.02138	27.2598	28.29237	29.77421	31.07663	32.22341	33.26578	34.62679	35.7442
24.82375	26.01745	27.24233	28.26917	29.77472	31.02476	32.22302	33.27844	34.65945	35.71012
24.82457	26.02821	27.21385	28.29064	29.77039	31.05645	32.22217	33.27309	34.65895	35.74343
24.84668	26.00787	27.25433	28.29189	29.75706	31.06671	32.22161	33.27235	34.64734	35.73974
24.84854	26.02025	27.26194	28.28249	29.76204	31.06433	32.22095	33.26547	34.6559	35.73387
24.84769	26.00896	27.25759	28.27597	29.76791	31.06278	32.22137	33.2691	34.65933	35.72528
24.84891	26.01612	27.25032	28.27559	29.77308	31.04446	32.22202	33.25487	34.65988	35.75615
24.85465	26.02457	27.25133	28.27996	29.77129	31.06596	32.2198	33.25451	34.66149	35.75133
24.85423	26.00687	27.23198	28.28382	29.77393	31.05588	32.22086	33.2877	34.64439	35.73894
24.84993	26.02155	27.25322	28.27546	29.75337	31.05537	32.22162	33.2781	34.65817	35.74166
24.84883	26.01801	27.24657	28.27254	29.74586	31.06518	32.22235	33.28628	34.65326	35.73256
24.8463	26.01473	27.25295	28.28073	29.76774	31.06315	32.22187	33.28697	34.64677	35.75474
24.84853	26.0086	27.23857	28.28898	29.74473	31.06718	32.22088	33.27414	34.65272	35.74311
24.84794	26.00706	27.24818	28.28694	29.75922	31.06827	32.22164	33.27483	34.64307	35.75823
24.86728	25.99043	27.22786	28.27701	29.75759	31.072	32.21265	33.27678	34.65231	35.76311
24.86596	26.0123	27.24501	28.29376	29.74962	31.07394	32.19372	33.25475	34.64649	35.76452
24.86104	26.00986	27.23066	28.28971	29.7351	31.07383	32.1976	33.25817	34.63927	35.7096
24.85164	26.01695	27.24383	28.29562	29.75674	31.07246	32.19773	33.25748	34.64231	35.75385
24.86644	25.99109	27.2326	28.27466	29.76423	31.05815	32.1955	33.25662	34.60532	35.74052
24.86599	26.01931	27.23731	28.28174	29.76653	31.06379	32.19675	33.28431	34.64605	35.76648
24.86676	26.02267	27.24044	28.29354	29.74701	31.06666	32.19949	33.28187	34.6309	35.77015
24.85474	26.0179	27.23299	28.28929	29.75103	31.06381	32.20334	33.27666	34.62528	35.75241
24.86159	25.99999	27.23303	28.29437	29.76221	31.06094	32.18632	33.27479	34.64319	35.76766
24.83055	25.96946	27.22658	28.28875	29.77686	31.05946	32.22649	33.27524	34.64661	35.78939
24.85481	26.00987	27.21826	28.283	29.74946	31.04652	32.2259	33.26271	34.64824	35.76572
24.86019	26.01989	27.23471	28.29029	29.77262	31.04657	32.22597	33.27582	34.6048	35.788
24.86442	26.02554	27.21992	28.28288	29.77552	31.04462	32.21311	33.27914	34.64383	35.7995
24.865	26.0124	27.22394	28.29382	29.7695	31.04324	32.22458	33.27211	34.64015	35.77612
24.86795	26.01996	27.21421	28.28173	29.77354	31.02127	32.22726	33.27235	34.63699	35.8006
24.87589	26.00494	27.22814	28.28994	29.73886	31.04624	32.22897	33.28188	34.64709	35.79988
24.86192	26.01098	27.21928	28.27781	29.7753	31.04914	32.23083	33.2782	34.6449	35.80263

24.85947	26.01567	27.22496	28.28918	29.7724	31.05317	32.23211	33.27116	34.65023	35.79476
24.857	26.01789	27.21828	28.29004	29.74325	31.05541	32.23182	33.27771	34.65527	35.78057
24.83916	26.01627	27.21002	28.29231	29.76854	31.05898	32.23175	33.23042	34.65417	35.79591
24.87051	26.02611	27.19939	28.28255	29.7665	31.04596	32.2309	33.25296	34.63727	35.80146
24.84499	26.01458	27.21937	28.29043	29.75518	31.06126	32.22461	33.25339	34.64519	35.78676
24.84517	26.01394	27.20526	28.25801	29.7451	31.04427	32.22767	33.24354	34.65879	35.79516
24.84708	26.00498	27.21444	28.29419	29.75711	31.0502	32.22797	33.26995	34.65739	35.7868
24.85035	26.01383	27.21813	28.29981	29.75421	31.06435	32.2288	33.26602	34.64726	35.80583
24.86393	26.01515	27.22067	28.25547	29.75171	31.0639	32.23223	33.28331	34.63647	35.80883
24.87234	26.01491	27.21913	28.29843	29.72759	31.05686	32.23285	33.28549	34.64507	35.80598
24.86001	26.01648	27.21374	28.30129	29.73934	31.03074	32.23968	33.28548	34.62407	35.80608
24.86872	26.02401	27.20428	28.29435	29.70247	31.06332	32.24004	33.28674	34.65077	35.78748
24.87134	26.01923	27.21977	28.29212	29.74415	31.06025	32.23478	33.29592	34.61222	35.80309
24.88846	26.02868	27.21341	28.27304	29.75152	31.06154	32.23357	33.26676	34.63934	35.7894
24.80521	26.0298	27.20093	28.29411	29.74999	31.06254	32.2327	33.27368	34.63791	35.81466
24.89103	26.02368	27.20091	28.29921	29.74449	31.06039	32.23271	33.29031	34.63881	35.81236
24.88248	26.03065	27.22646	28.29984	29.74457	31.05291	32.2394	33.2888	34.64329	35.82437
24.88277	26.03044	27.22188	28.30219	29.73213	31.04146	32.23743	33.28234	34.64617	35.82369
24.87703	26.03456	27.21405	28.3047	29.75561	31.04678	32.23434	33.29244	34.64674	35.81007
24.88976	26.01861	27.22296	28.30863	29.73814	31.02788	32.23259	33.29446	34.64683	35.81044
24.83584	26.03366	27.20868	28.30399	29.76003	31.02259	32.22355	33.29468	34.64136	35.83427
24.893	26.0389	27.19731	28.30017	29.75954	31.04143	32.22233	33.28619	34.64235	35.82699
24.88688	26.03313	27.22178	28.3137	29.75217	31.0188	32.21935	33.28702	34.63946	35.81568
24.88816	26.0345	27.20948	28.30045	29.75159	31.05075	32.21804	33.28464	34.65016	35.83678
24.89478	26.02934	27.19861	28.29037	29.7474	31.04884	32.20884	33.27871	34.62933	35.82362
24.89605	26.02276	27.20774	28.31029	29.74754	31.04184	32.21465	33.27689	34.64136	35.82725
24.88961	26.02044	27.21614	28.30483	29.70708	31.02761	32.21768	33.27095	34.64129	35.82955
24.89176	26.0396	27.20554	28.31635	29.71764	31.02724	32.21635	33.27003	34.64226	35.83684
24.88121	26.03309	27.21556	28.32015	29.74043	31.03593	32.21407	33.27308	34.64486	35.83399
24.88069	26.02084	27.21434	28.31677	29.67397	31.03792	32.21548	33.27048	34.64458	35.82114
24.89894	26.02729	27.204	28.28983	29.70099	31.04974	32.21766	33.25566	34.63837	35.83909
24.89976	26.03505	27.21503	28.29954	29.69619	31.03379	32.21788	33.25401	34.63248	35.83432
24.89924	26.03851	27.21404	28.31714	29.71261	31.04794	32.22019	33.25325	34.5812	35.85214
24.90279	26.01664	27.21672	28.32516	29.72574	31.04572	32.23358	33.26951	34.63699	35.84807
24.90359	26.02335	27.2099	28.32611	29.72323	31.05143	32.23652	33.26224	34.63754	35.85826
24.88272	26.03965	27.20586	28.3086	29.7228	31.04784	32.23803	33.25773	34.62247	35.86
24.8955	26.04577	27.21572	28.27725	29.68483	31.04634	32.22743	33.26331	34.62614	35.85957
24.90312	26.03824	27.21827	28.31864	29.72094	31.02602	32.23186	33.25712	34.62859	35.83068
24.90435	26.023	27.20014	28.31539	29.72833	31.03997	32.23501	33.25012	34.62012	35.84937
24.89838	26.0499	27.20732	28.32622	29.71902	31.03156	32.24444	33.26536	34.61299	35.86617
24.9149	26.03948	27.21757	28.32282	29.72109	31.04331	32.24486	33.26287	34.62075	35.86517
24.9181	26.03102	27.20898	28.3197	29.71651	31.04045	32.24624	33.26286	34.61766	35.85834
24.91916	25.98691	27.21976	28.31313	29.73675	31.04939	32.23279	33.26682	34.60818	35.86707

24.88626	26.04548	27.21473	28.32235	29.73288	31.05103	32.25658	33.26028	34.60048	35.86376
24.88483	26.04502	27.21797	28.33382	29.73632	31.05987	32.25603	33.2686	34.59108	35.86588
24.91558	26.04721	27.22858	28.32724	29.73349	31.05819	32.25418	33.26581	34.60659	35.8649
24.90875	26.03788	27.23402	28.33489	29.73428	31.03812	32.2514	33.2667	34.61181	35.87326
24.91302	26.01446	27.23569	28.33022	29.71965	31.04776	32.25233	33.26698	34.60602	35.86604
24.90036	26.02729	27.23651	28.32569	29.71444	31.049	32.25125	33.2714	34.60652	35.84974
24.92637	26.04148	27.2295	28.33369	29.73211	31.05145	32.24424	33.27301	34.59837	35.87221
24.92542	26.03468	27.22409	28.33188	29.71847	31.04968	32.22973	33.26756	34.61655	35.86572
24.92738	26.03555	27.23918	28.33006	29.70191	31.04944	32.2419	33.27323	34.61477	35.86345
24.91119	26.04011	27.23236	28.32625	29.71741	31.04994	32.24694	33.27297	34.60502	35.85458
24.91198	25.96512	27.2276	28.33034	29.7124	31.04026	32.23736	33.27303	34.59809	35.87263
24.93266	26.0396	27.23316	28.32296	29.69125	31.03899	32.24468	33.2696	34.60304	35.8425
24.93442	26.04244	27.24282	28.33657	29.68254	31.04614	32.24448	33.26929	34.62184	35.86339
24.93501	26.03105	27.19795	28.31595	29.6981	31.05075	32.24534	33.26853	34.59626	35.85522
24.92987	26.02659	27.23364	28.33395	29.68215	31.05705	32.24397	33.27103	34.61974	35.84525
24.93034	26.04504	27.24015	28.32774	29.69865	31.04068	32.24642	33.27015	34.60857	35.83841
24.91314	26.02205	27.23234	28.32823	29.69188	31.037	32.23502	33.27146	34.62387	35.86456
24.94146	26.03133	27.23853	28.32958	29.68907	31.06105	32.23654	33.26577	34.61589	35.86819
24.94465	26.04886	27.20173	28.33035	29.68554	31.06251	32.23621	33.26419	34.62293	35.86833
24.94068	26.04617	27.22465	28.33516	29.67664	31.04531	32.23691	33.27005	34.61479	35.834
24.94099	26.05477	27.22373	28.33736	29.6822	31.05869	32.23821	33.26997	34.62431	35.86998
24.94161	26.04016	27.23443	28.33401	29.70462	31.05988	32.23952	33.24324	34.58648	35.84075
24.94701	26.04618	27.23576	28.33001	29.69723	31.03441	32.2421	33.23113	34.60286	35.87065
24.94883	26.05495	27.2271	28.33123	29.70383	31.05949	32.245	33.23505	34.60502	35.87638
24.94711	26.05497	27.23257	28.3163	29.69675	31.06527	32.25246	33.26	34.59908	35.85891
24.91816	26.04957	27.23639	28.33376	29.71043	31.06107	32.24077	33.26431	34.60424	35.87134
24.89218	26.05371	27.20854	28.31577	29.71458	31.05065	32.24131	33.26182	34.59423	35.88132
24.89264	26.05002	27.23552	28.34067	29.71289	31.06041	32.25257	33.26788	34.59177	35.88636
24.93229	26.034	27.24446	28.3136	29.70393	31.06207	32.25361	33.25656	34.59993	35.88643
24.94705	26.04894	27.2357	28.33032	29.70409	31.04111	32.25456	33.26092	34.59465	35.8914
24.94869	26.04327	27.24399	28.32282	29.70495	31.03963	32.23946	33.26012	34.58986	35.88386
24.95834	26.04228	27.25011	28.31693	29.71956	31.06436	32.25135	33.25872	34.58275	35.88693
24.95734	26.05403	27.24489	28.33152	29.67656	31.06346	32.25387	33.24172	34.58518	35.88699
24.94664	26.05339	27.25883	28.31814	29.70776	31.07042	32.26325	33.25056	34.5615	35.85459
24.94114	26.05125	27.26139	28.33299	29.71748	31.06956	32.26259	33.23395	34.58987	35.8736
24.94606	26.04817	27.24401	28.32895	29.71237	31.06724	32.2361	33.22969	34.58562	35.87597
24.94878	26.03432	27.25589	28.32151	29.70269	31.05325	32.26361	33.22429	34.59006	35.87758
24.96591	26.04473	27.25133	28.32931	29.70232	31.06874	32.26255	33.24564	34.58877	35.88726
24.93833	26.04226	27.25168	28.31059	29.69284	31.06386	32.26424	33.24339	34.57437	35.86815
24.96539	26.05417	27.2601	28.32136	29.66724	31.06331	32.24554	33.21543	34.5859	35.88259
24.96961	26.04829	27.25558	28.32245	29.68127	31.06245	32.24582	33.22508	34.58166	35.87611
24.93384	26.05226	27.2585	28.3064	29.6993	31.06262	32.24327	33.22949	34.59289	35.86316
24.97463	26.05642	27.26015	28.31843	29.68619	31.06198	32.24372	33.22893	34.55849	35.87427

24.97298	26.05567	27.254	28.31945	29.69058	31.05115	32.24033	33.24007	34.58419	35.8394
24.96803	26.04756	27.24999	28.31814	29.69824	31.05287	32.24949	33.24182	34.58813	35.86713
24.97569	26.05167	27.23656	28.31686	29.69295	31.05162	32.24869	33.22959	34.5645	35.82822
24.97822	26.04302	27.24599	28.31895	29.70323	31.0508	32.24578	33.23391	34.55428	35.86883
24.98214	26.03689	27.24628	28.28639	29.68895	31.06447	32.24524	33.24862	34.522	35.83375
24.9867	26.04656	27.26184	28.31997	29.70866	31.06274	32.24498	33.23158	34.5862	35.85452
24.97752	26.0497	27.24005	28.32068	29.7151	31.04481	32.24452	33.25315	34.58777	35.86086
24.96911	26.03454	27.24569	28.32504	29.68976	31.0282	32.2443	33.25808	34.57994	35.8624
24.96625	26.0214	27.26274	28.31932	29.71233	31.0611	32.24573	33.25886	34.58229	35.85999
24.98707	26.04348	27.25391	28.28872	29.67516	31.06241	32.24057	33.25966	34.57208	35.83642
24.98513	26.038	27.2539	28.31237	29.72322	31.05286	32.2259	33.212	34.57421	35.8458
24.94791	26.04521	27.26422	28.31832	29.71412	31.0676	32.23185	33.21718	34.58499	35.85268
24.9611	26.0156	27.24169	28.31082	29.72274	31.07261	32.24904	33.26429	34.58584	35.86071
24.96546	26.01581	27.25506	28.3144	29.72224	31.05235	32.24196	33.26036	34.5877	35.84573
24.9495	26.03793	27.26152	28.32073	29.7364	31.04857	32.24044	33.27449	34.57735	35.86292
24.98043	26.03328	27.24784	28.3114	29.73506	31.06598	32.23931	33.27315	34.56561	35.85235
24.98592	26.02377	27.2497	28.31774	29.73733	31.02009	32.23892	33.2702	34.58042	35.86232
24.98929	26.019	27.26373	28.30499	29.72809	31.0669	32.22845	33.27129	34.5779	35.81258
24.99511	26.03796	27.25278	28.30405	29.74063	31.0609	32.23598	33.26621	34.57901	35.86363
25.00127	26.03925	27.22651	28.30225	29.7335	31.07158	32.25029	33.26382	34.56498	35.80446
25.00295	26.04424	27.2675	28.29417	29.69237	31.07099	32.23519	33.2588	34.56927	35.85713
25.00239	26.02764	27.25936	28.30397	29.69677	31.06837	32.23526	33.24925	34.57642	35.86479
24.98819	26.03449	27.25954	28.3093	29.72338	31.06477	32.22904	33.24423	34.54357	35.85405
24.99208	26.02283	27.26542	28.30698	29.71667	31.05381	32.24762	33.24807	34.57805	35.86399
24.98398	26.04987	27.26281	28.27998	29.72289	31.0512	32.24523	33.23965	34.57822	35.85668
24.97562	26.03707	27.26007	28.31087	29.72948	31.05834	32.23824	33.24119	34.57569	35.85734
25.0024	26.03802	27.2589	28.29562	29.72922	31.07159	32.23638	33.23844	34.56963	35.85671
25.00439	26.05576	27.24846	28.29838	29.72956	31.02807	32.23942	33.22354	34.58295	35.85
25.00623	26.04506	27.2589	28.30039	29.73455	31.08466	32.24073	33.22267	34.57076	35.85311
24.98079	25.99444	27.24982	28.30256	29.72217	31.08498	32.24142	33.21888	34.59025	35.84097
25.00042	26.03837	27.25785	28.3085	29.73363	31.08014	32.2421	33.22647	34.56608	35.84612
25.01	26.04764	27.25506	28.25275	29.71492	31.08459	32.24303	33.22263	34.58803	35.84061
25.01179	26.0222	27.24022	28.29205	29.72094	31.08106	32.24349	33.22427	34.58497	35.841
24.98975	26.03743	27.25073	28.28434	29.73405	31.08964	32.24696	33.21954	34.5738	35.84107
25.01103	26.02435	27.24585	28.30076	29.74703	31.07902	32.24473	33.22461	34.57556	35.81701
25.01117	26.02957	27.24983	28.30303	29.73911	31.08344	32.24469	33.21677	34.59232	35.81912
25.00017	26.02886	27.24666	28.29279	29.75076	31.07241	32.2504	33.22876	34.57507	35.82884
25.001	26.01734	27.23519	28.2832	29.75934	31.0763	32.24918	33.23398	34.58904	35.80927
24.99509	26.02311	27.25237	28.30194	29.71533	31.07618	32.23975	33.23313	34.57807	35.81768
25.01011	26.01731	27.24547	28.29315	29.76591	31.05965	32.23952	33.22116	34.55073	35.81548
25.01217	26.02176	27.23771	28.2956	29.72193	31.06273	32.23984	33.23249	34.56352	35.81876
25.01327	26.01771	27.24509	28.27644	29.7681	31.06382	32.24072	33.23502	34.57636	35.80245
25.01386	26.03112	27.24832	28.27408	29.76815	31.08067	32.24002	33.23731	34.58439	35.81625

24.99471	26.03273	27.25051	28.29148	29.76742	31.08835	32.24458	33.24318	34.57208	35.79296
25.00933	26.03143	27.24533	28.29835	29.74577	31.08695	32.24824	33.23334	34.52843	35.81418
24.97334	26.01992	27.25071	28.27203	29.76798	31.0872	32.24685	33.23208	34.55576	35.80643
25.01265	26.00194	27.24671	28.3007	29.76592	31.0238	32.24633	33.24922	34.58211	35.79283
25.01011	26.03608	27.24353	28.25667	29.75813	31.09185	32.24316	33.24959	34.57888	35.79736
25.00953	25.99646	27.2243	28.29549	29.76659	31.08783	32.24205	33.25128	34.57738	35.79493
24.9767	26.0327	27.24785	28.30286	29.7376	31.08741	32.24241	33.22909	34.56665	35.7684
25.00752	26.02228	27.23715	28.30281	29.76965	31.07824	32.23926	33.24117	34.57576	35.79018
25.00756	26.02924	27.24016	28.29821	29.7668	31.07805	32.23862	33.22345	34.58345	35.76381
25.00404	26.00363	27.24506	28.28643	29.75108	31.06426	32.23928	33.20244	34.57238	35.76538
25.00737	26.02799	27.23245	28.29859	29.7658	31.08821	32.23834	33.24777	34.57665	35.78235
24.99152	26.02774	27.22826	28.2865	29.76569	31.08666	32.23562	33.19968	34.56582	35.77783
25.00276	26.01557	27.23477	28.29219	29.77519	31.07516	32.23699	33.25062	34.57535	35.76657
25.00161	25.99975	27.2192	28.28653	29.77097	31.07704	32.23951	33.23765	34.58795	35.76887
25.00059	26.01978	27.23382	28.28808	29.77229	31.08984	32.24157	33.23238	34.59239	35.7442
25.00211	26.00418	27.23711	28.26461	29.73821	31.08206	32.24311	33.22588	34.57289	35.77376
24.99341	26.01843	27.21619	28.28973	29.76887	31.07956	32.2421	33.24108	34.58283	35.76297
24.98492	26.01516	27.23139	28.29317	29.74389	31.08085	32.24303	33.24579	34.57746	35.75344
24.98658	26.00661	27.23576	28.29019	29.77998	31.08221	32.24966	33.23406	34.55121	35.76366
24.97678	26.01782	27.22896	28.28597	29.76531	31.07756	32.24911	33.22916	34.58415	35.77822
24.98986	26.00717	27.23846	28.29542	29.75552	31.08861	32.24856	33.23281	34.5619	35.76617
24.95836	26.02191	27.24066	28.28823	29.77479	31.08896	32.24699	33.21683	34.5883	35.76202
24.9589	26.01989	27.23756	28.2958	29.77026	31.08661	32.24	33.23387	34.59114	35.77299
24.97172	26.02283	27.21826	28.28551	29.75791	31.06495	32.23591	33.22222	34.5905	35.76395
24.98281	25.99946	27.237	28.2698	29.7782	31.08704	32.2341	33.18707	34.58069	35.75077
24.9736	26.02907	27.24126	28.30309	29.77718	31.08149	32.23434	33.22754	34.59093	35.76897
24.98373	26.02859	27.25568	28.29873	29.7706	31.0822	32.23373	33.22996	34.58169	35.76739
24.98291	26.03187	27.2313	28.29283	29.77351	31.08075	32.236	33.22711	34.58263	35.76398
24.98517	26.03088	27.24148	28.27911	29.76671	31.08636	32.23024	33.23035	34.58451	35.75915
24.98401	26.00828	27.23906	28.28118	29.76934	31.08026	32.23041	33.22188	34.59059	35.7452
24.97494	26.03869	27.245	28.2941	29.75846	31.07987	32.22856	33.23204	34.57782	35.75618
24.97364	26.03145	27.24204	28.29765	29.7312	31.06073	32.21247	33.23117	34.57448	35.75534
24.97091	26.03332	27.25393	28.30008	29.76868	31.06388	32.21921	33.20503	34.55272	35.74457
24.97076	26.03352	27.25667	28.2939	29.77417	31.06919	32.22528	33.20785	34.58795	35.75167
24.97055	26.04044	27.25697	28.30438	29.77881	31.06769	32.22415	33.22368	34.55659	35.75765
24.96428	26.03179	27.25519	28.29763	29.78448	31.07891	32.23775	33.22672	34.5801	35.75881
24.96027	26.01347	27.23849	28.27155	29.77929	31.0665	32.23805	33.2361	34.55982	35.74834
24.96201	26.02338	27.24934	28.30304	29.78056	31.06777	32.22497	33.23634	34.58113	35.75207
24.93394	26.04135	27.25403	28.29168	29.78435	31.07343	32.2187	33.23799	34.5814	35.74553
24.93205	26.01633	27.25542	28.30372	29.79	31.07938	32.21739	33.23702	34.58603	35.74427
24.94976	26.04157	27.24591	28.29443	29.79107	31.07981	32.21589	33.22216	34.57545	35.75483
24.95024	26.03008	27.22307	28.30592	29.77766	31.07654	32.20275	33.22622	34.59094	35.73947
24.89285	26.03872	27.25371	28.30301	29.7736	31.0839	32.20907	33.22977	34.59197	35.74621

24.94904	26.04603	27.25501	28.27195	29.79066	31.08818	32.20941	33.23781	34.58275	35.73397
24.93742	26.04327	27.24237	28.29417	29.78241	31.06697	32.20861	33.23719	34.56958	35.73091
24.92727	26.04172	27.24944	28.30138	29.77515	31.06787	32.18933	33.2349	34.56289	35.73601
24.89522	26.0353	27.24648	28.30465	29.79396	31.05719	32.21482	33.21059	34.56236	35.7123
24.89218	26.05163	27.21543	28.31279	29.79006	31.08557	32.22099	33.2128	34.58968	35.72818
24.87848	26.0492	27.24292	28.262	29.77273	31.08622	32.21861	33.21933	34.59188	35.71329
24.92438	26.02284	27.2343	28.28593	29.76274	31.08834	32.21384	33.23362	34.5607	35.73528
24.9289	26.05522	27.24411	28.31191	29.7851	31.08103	32.21456	33.23031	34.58759	35.73523
24.91635	26.05548	27.25155	28.28571	29.77622	31.09344	32.21308	33.21675	34.58043	35.72637
24.92655	26.04329	27.24861	28.31102	29.77684	31.08698	32.2136	33.21321	34.59013	35.73559
24.92322	26.06099	27.25092	28.30971	29.77476	31.08677	32.21136	33.22389	34.57148	35.7005
24.90211	26.06043	27.262	28.30506	29.78056	31.08232	32.21039	33.21624	34.5651	35.72055
24.89816	26.05794	27.25518	28.31312	29.76765	31.08261	32.21556	33.20746	34.58546	35.73693
24.92217	26.04415	27.25561	28.31208	29.77587	31.09178	32.21558	33.21777	34.566	35.72555
24.91828	26.06029	27.25968	28.31275	29.77343	31.07424	32.2156	33.21792	34.56871	35.72109
24.91383	26.06351	27.24762	28.31621	29.76563	31.08933	32.21724	33.22473	34.58037	35.70445
24.91099	26.05748	27.25455	28.30379	29.78197	31.07516	32.20685	33.22075	34.58006	35.74168
24.88747	26.04209	27.26104	28.30937	29.7716	31.07458	32.22867	33.22165	34.56842	35.72413
24.9005	26.03946	27.23855	28.3213	29.78279	31.07698	32.23038	33.22489	34.57659	35.71934
24.90066	26.05249	27.22697	28.32045	29.7723	31.05921	32.22931	33.22578	34.57335	35.74008
24.89701	26.02299	27.25934	28.30872	29.77964	31.07861	32.22917	33.22294	34.54372	35.7328
24.89279	26.05358	27.2329	28.33462	29.7586	31.07793	32.23077	33.2246	34.58649	35.7428
24.88383	26.05069	27.24891	28.33807	29.78168	31.08301	32.20122	33.22776	34.58585	35.72757
24.8658	26.05801	27.25842	28.32423	29.77683	31.03071	32.22277	33.21691	34.57582	35.7445
24.87247	26.02778	27.23689	28.32078	29.78345	31.0715	32.22098	33.2155	34.56936	35.73388
24.86929	26.0662	27.25997	28.31705	29.76154	31.06914	32.22118	33.17983	34.56552	35.74117
24.87159	26.06985	27.24907	28.33288	29.77563	31.06523	32.2191	33.22323	34.58336	35.73862
24.85318	26.01862	27.24824	28.31814	29.78386	31.0721	32.19627	33.17211	34.59242	35.74436
24.85185	26.07108	27.25169	28.32604	29.72103	31.05068	32.19755	33.20613	34.56917	35.70427
24.85544	26.03284	27.25671	28.32672	29.77337	31.01231	32.18817	33.22771	34.58409	35.73544
24.87165	26.07426	27.25033	28.33158	29.76721	31.04125	32.21658	33.18692	34.58364	35.73023
24.8692	26.05508	27.2492	28.31683	29.77687	31.08071	32.21885	33.22305	34.5805	35.72696
24.86825	26.05826	27.25066	28.33411	29.77103	31.08178	32.21767	33.1986	34.58319	35.73898
24.86341	26.07599	27.25388	28.30131	29.76561	31.08314	32.22193	33.19354	34.57322	35.73909
24.86347	26.07473	27.25453	28.33375	29.71401	31.08318	32.22928	33.21172	34.57879	35.70016
24.85812	26.07258	27.24664	28.3217	29.74842	31.0758	32.22853	33.21897	34.57704	35.74158
24.84678	26.06996	27.24214	28.33399	29.76449	31.08217	32.22814	33.20766	34.56476	35.74289
24.83858	26.06789	27.25131	28.32299	29.73349	31.08371	32.22782	33.23035	34.56463	35.74946
24.83654	26.07435	27.23871	28.32903	29.75905	31.08076	32.22795	33.23363	34.57921	35.72548
24.84309	26.06658	27.25125	28.32967	29.75426	31.0799	32.22685	33.23946	34.57703	35.74018
24.8449	26.07623	27.24604	28.33454	29.75663	31.06133	32.18857	33.23922	34.53681	35.74197
24.84136	26.05206	27.22287	28.33382	29.73724	31.07072	32.20562	33.24197	34.57289	35.74949
24.82868	26.05706	27.25566	28.33912	29.74662	31.03723	32.20724	33.22991	34.55655	35.74794

24.82252	26.07899	27.2495	28.33739	29.75517	31.06288	32.2074	33.24055	34.57552	35.758
24.82346	26.07484	27.24792	28.34535	29.73756	31.06131	32.1904	33.24157	34.57422	35.69342
24.81807	26.06136	27.24476	28.32388	29.76187	31.03486	32.21552	33.24422	34.56146	35.74883
24.81365	26.06735	27.25302	28.33539	29.76478	31.06414	32.22358	33.21904	34.57739	35.7461
24.81059	26.07821	27.25638	28.32254	29.7657	31.06553	32.22376	33.23697	34.56775	35.74187
24.79086	26.07522	27.26007	28.34764	29.75264	31.05969	32.22344	33.22284	34.54276	35.76297
24.79967	26.04939	27.25905	28.3335	29.76085	31.06715	32.22346	33.22569	34.5672	35.76662
24.76367	26.07257	27.25858	28.34717	29.73609	31.06543	32.22371	33.22819	34.58073	35.77809
24.8064	26.07507	27.25486	28.35187	29.76153	31.0644	32.22383	33.22707	34.55462	35.77093
24.78953	26.07507	27.24591	28.3546	29.73449	31.06508	32.22463	33.23136	34.5724	35.76828
24.80364	26.07795	27.25365	28.34248	29.76334	31.04547	32.22482	33.23142	34.57894	35.78061
24.80119	26.06819	27.2187	28.35437	29.76204	31.06268	32.22032	33.23958	34.58054	35.78513
24.79757	26.06306	27.25766	28.35004	29.75634	31.06263	32.22812	33.24079	34.57245	35.77317
24.79517	26.06984	27.25572	28.34381	29.73085	31.063	32.22975	33.22726	34.56541	35.79422
24.79365	26.07166	27.24657	28.33284	29.74144	31.03291	32.23008	33.22094	34.5632	35.79092
24.79298	26.08706	27.24316	28.3573	29.7452	31.04449	32.2214	33.23028	34.57761	35.77344
24.79083	26.05227	27.23257	28.3541	29.74784	31.06097	32.22543	33.23714	34.57379	35.79442
24.78768	26.05753	27.23573	28.33228	29.73537	31.03387	32.22558	33.24192	34.5605	35.78717
24.78427	26.07722	27.22482	28.34773	29.73258	31.0588	32.22523	33.24105	34.58105	35.75397
24.78291	26.08379	27.25005	28.36074	29.74891	31.06839	32.22523	33.21901	34.57257	35.77944
24.78287	26.08045	27.25114	28.33129	29.73671	31.06693	32.22469	33.23821	34.57593	35.78435
24.78214	26.08146	27.24066	28.35144	29.7262	31.01678	32.21652	33.23717	34.57711	35.78907
24.78267	26.07576	27.24168	28.35332	29.73268	31.05429	32.21691	33.23394	34.55862	35.77684
24.78028	26.05486	27.24878	28.35148	29.74837	31.06362	32.20436	33.20529	34.56421	35.79445
24.76229	26.08025	27.24429	28.31489	29.71949	31.06142	32.17703	33.23099	34.57371	35.79663
24.76099	26.07605	27.23301	28.35383	29.7305	31.04989	32.21659	33.22742	34.56108	35.79552
24.77167	26.08288	27.2289	28.34956	29.74052	31.06776	32.21309	33.22855	34.56558	35.78868
24.76672	26.0674	27.24432	28.33863	29.74368	31.06603	32.20735	33.22129	34.56297	35.79282
24.77072	26.08204	27.24741	28.35258	29.73424	31.06383	32.20627	33.23238	34.55534	35.78728
24.77016	26.08083	27.21976	28.34851	29.74424	31.06993	32.2049	33.23369	34.53387	35.80377
24.77574	26.05105	27.24341	28.3422	29.75246	31.06424	32.20752	33.21352	34.54727	35.77836
24.77063	26.06314	27.23481	28.34243	29.75496	31.0594	32.19644	33.21118	34.53366	35.76999
24.76492	26.07401	27.21242	28.34617	29.73267	31.07503	32.22134	33.22486	34.54696	35.81543
24.76091	26.06563	27.2409	28.32456	29.7596	31.07632	32.22174	33.23348	34.55289	35.82434
24.76074	26.06089	27.25	28.35867	29.74609	31.07386	32.22204	33.23612	34.54378	35.82076
24.76197	26.08189	27.24928	28.33514	29.74516	31.07578	32.22436	33.23511	34.55079	35.81589
24.75396	26.07871	27.2489	28.35467	29.74937	31.07423	32.23753	33.24164	34.55473	35.82017
24.75088	26.06823	27.2466	28.34394	29.75113	31.05921	32.23768	33.23357	34.53746	35.83807
24.74105	26.07264	27.23946	28.35191	29.75341	31.07548	32.23679	33.2168	34.5324	35.84693
24.73777	26.0685	27.24453	28.35191	29.75219	31.07373	32.23804	33.22041	34.56175	35.83922
24.73429	26.07421	27.25131	28.34673	29.74988	31.06416	32.23356	33.22081	34.56204	35.85288
24.73439	26.06925	27.25544	28.35077	29.74285	31.06402	32.23124	33.24762	34.5635	35.83788
24.75432	26.06679	27.24227	28.33388	29.73596	31.07177	32.23073	33.24692	34.55062	35.8529

24.75385	26.06662	27.24928	28.33847	29.72829	31.07139	32.23097	33.24063	34.53175	35.85504
24.75275	26.05692	27.22552	28.34317	29.73498	31.06408	32.22996	33.25564	34.54112	35.85714
24.75573	26.03915	27.23476	28.33472	29.73644	31.06086	32.21488	33.25635	34.5475	35.84612
24.7558	26.05958	27.2405	28.32016	29.70748	31.06447	32.21471	33.24915	34.52983	35.85895
24.75052	26.04812	27.24283	28.34255	29.73544	31.06457	32.21487	33.25976	34.54228	35.86989
24.72494	26.06155	27.25137	28.3466	29.73157	31.04084	32.21411	33.26178	34.52686	35.85998
24.72284	26.04796	27.23054	28.34722	29.73986	31.05903	32.21456	33.24533	34.54119	35.84588
24.73225	26.04777	27.24823	28.34257	29.74134	31.04696	32.21481	33.24083	34.52011	35.86961
24.7548	26.05542	27.25046	28.34418	29.7418	31.05256	32.21207	33.25706	34.53996	35.86366
24.75233	26.0501	27.23216	28.33576	29.73713	31.0567	32.21919	33.25463	34.54169	35.85501
24.74643	26.0548	27.24717	28.28445	29.73532	31.03795	32.21962	33.2516	34.53998	35.87243
24.75458	26.01953	27.24156	28.30078	29.74386	31.05647	32.22057	33.25045	34.53254	35.87136
24.74073	26.0525	27.24112	28.34215	29.75324	31.0312	32.21137	33.25209	34.50872	35.84902
24.73933	26.05309	27.24781	28.33788	29.73946	31.03676	32.22853	33.25668	34.4954	35.87474
24.73777	26.04922	27.25069	28.33736	29.73693	31.05188	32.21768	33.25487	34.49292	35.87968
24.73057	26.04544	27.25934	28.33856	29.75834	31.04194	32.18734	33.24033	34.52935	35.86677
24.73074	26.03974	27.2498	28.30521	29.76142	31.05161	32.18122	33.23264	34.53113	35.83741
24.75014	26.0448	27.25768	28.34117	29.75855	31.05104	32.21261	33.21628	34.51301	35.88789
24.75333	26.04886	27.26629	28.34421	29.75529	31.06089	32.21265	33.23364	34.52436	35.85807
24.75135	26.05115	27.23727	28.31911	29.74731	31.0599	32.21214	33.24232	34.51383	35.88806
24.75362	26.03751	27.25353	28.34608	29.75527	31.02262	32.22983	33.24384	34.52816	35.89271
24.74281	26.0464	27.26935	28.33923	29.73598	31.03545	32.23209	33.24777	34.52179	35.89536
24.73974	26.03303	27.26179	28.34729	29.74539	31.05198	32.23858	33.24926	34.51903	35.89523
24.73924	26.02471	27.25791	28.34561	29.75396	31.06216	32.23922	33.24357	34.51398	35.89321
24.7512	26.03632	27.25914	28.32537	29.74824	31.06444	32.24191	33.24567	34.52071	35.91134
24.7521	26.03843	27.26488	28.32729	29.7539	31.07296	32.24533	33.24896	34.50912	35.91597
24.75412	26.02489	27.25677	28.33693	29.70655	31.07755	32.24249	33.25014	34.52143	35.90917
24.7517	26.01527	27.24622	28.3488	29.71794	31.07582	32.23434	33.24784	34.51924	35.92092
24.73866	26.02598	27.25389	28.34708	29.74	31.06601	32.23551	33.24391	34.52176	35.92242
24.7292	26.03266	27.2555	28.33886	29.75143	31.06504	32.23447	33.21081	34.51983	35.90622
24.75151	26.02958	27.24788	28.35543	29.74751	31.06247	32.18728	33.20945	34.5043	35.93274
24.75128	26.04053	27.22057	28.33457	29.73346	31.06625	32.22743	33.21801	34.49242	35.9283
24.75285	26.02706	27.25607	28.34528	29.75102	31.06324	32.22636	33.23291	34.51124	35.91451
24.75855	26.02954	27.24639	28.35461	29.73557	31.06033	32.22903	33.2268	34.47299	35.94456
24.75839	26.04091	27.23374	28.3411	29.75352	31.05427	32.22382	33.24115	34.50862	35.94178
24.73606	26.02514	27.25628	28.33018	29.74849	31.04152	32.19565	33.20748	34.51398	35.93994
24.76166	26.00541	27.24207	28.35526	29.7526	31.04693	32.22884	33.25273	34.51006	35.94315
24.76775	26.04616	27.25906	28.34837	29.7556	31.03814	32.22972	33.24625	34.49947	35.92329
24.73486	26.04497	27.24831	28.27876	29.76807	31.02046	32.22992	33.25096	34.49596	35.94239
24.75862	26.03839	27.22923	28.34802	29.76347	31.04393	32.22037	33.23679	34.51402	35.93485
24.75543	26.03405	27.26132	28.33774	29.72475	31.05015	32.20991	33.21988	34.50124	35.91033
24.75513	26.05035	27.25968	28.34506	29.75912	31.05119	32.2252	33.24311	34.50834	35.93816
24.76387	26.03966	27.25859	28.3297	29.73272	31.05186	32.22931	33.24366	34.49654	35.94197

24.76499	26.04554	27.24444	28.33301	29.75295	31.06103	32.23109	33.22264	34.50663	35.92063
24.73586	26.04338	27.25266	28.3399	29.75764	31.06077	32.23023	33.21118	34.49356	35.94965
24.73933	26.05329	27.26476	28.33275	29.72844	31.06259	32.23186	33.2519	34.50845	35.95101
24.77143	26.04444	27.21718	28.32049	29.75031	31.05378	32.23352	33.25288	34.51671	35.95348
24.77312	26.04084	27.2657	28.33347	29.73758	31.05147	32.24272	33.24276	34.50262	35.94207
24.77494	26.05116	27.2519	28.32838	29.73606	31.06533	32.24399	33.25282	34.5017	35.95393
24.77811	26.04866	27.25595	28.3228	29.73136	31.06765	32.25672	33.25239	34.49736	35.9597
24.77664	26.0409	27.26023	28.32512	29.73506	31.04535	32.25616	33.24156	34.5029	35.96262
24.76143	26.04477	27.25822	28.31246	29.72879	31.06135	32.25378	33.24067	34.50482	35.96319
24.76544	26.03957	27.24497	28.32245	29.75099	31.06218	32.25328	33.23365	34.49903	35.96107
24.78107	26.04788	27.22242	28.32283	29.74585	31.05541	32.2534	33.23841	34.49234	35.97939
24.79045	26.05343	27.25176	28.2813	29.74774	31.04486	32.25379	33.23463	34.50994	35.97346
24.79291	26.03636	27.24653	28.31054	29.7311	31.07669	32.25494	33.24061	34.51444	35.94546
24.79321	26.05032	27.24609	28.32313	29.7115	31.07393	32.2562	33.23873	34.50907	35.98085
24.78717	26.04431	27.25298	28.30765	29.74639	31.06505	32.24731	33.23619	34.51041	35.96648
24.7775	26.05441	27.24419	28.31165	29.74336	31.05766	32.2482	33.2354	34.50968	35.98668
24.80445	26.04593	27.19312	28.32841	29.75051	31.01996	32.24824	33.22628	34.50455	35.97761
24.80387	26.05231	27.23749	28.31794	29.75042	31.06346	32.24546	33.21225	34.50508	35.96614
24.74249	26.02124	27.22671	28.31216	29.73577	31.07644	32.24445	33.23119	34.49341	35.98259
24.79313	26.05138	27.2432	28.31355	29.75499	31.05159	32.24176	33.22942	34.50126	35.99107
24.77182	26.03918	27.24221	28.30669	29.75463	31.07659	32.23735	33.22515	34.51155	35.98134
24.77619	25.94695	27.23672	28.32171	29.74156	31.04944	32.23614	33.21145	34.51094	35.98433
24.7849	26.0501	27.2365	28.32628	29.75238	31.0488	32.23504	33.21236	34.50203	35.98709
24.78648	26.04857	27.2347	28.31604	29.73062	31.05793	32.2324	33.20432	34.48557	35.99238
24.7976	26.04295	27.22178	28.31109	29.73939	31.06535	32.23318	33.22593	34.50953	35.98411
24.81512	26.02689	27.24383	28.32106	29.73867	31.04916	32.23119	33.20453	34.47859	35.97442
24.81772	26.03403	27.23496	28.32429	29.72359	31.06072	32.23259	33.2208	34.50339	35.98171
24.80808	26.04123	27.23576	28.3228	29.73855	31.05719	32.23014	33.22201	34.48908	35.96655
24.82338	25.99569	27.24113	28.32457	29.73937	31.06501	32.23186	33.22745	34.50004	35.98961
24.82499	26.03647	27.23698	28.32553	29.72647	31.07149	32.2315	33.21603	34.4934	35.98797
24.81139	26.03121	27.20337	28.32482	29.73041	31.07211	32.23233	33.21517	34.4999	35.97192
24.81172	26.04198	27.22971	28.3126	29.72768	31.0669	32.23305	33.22078	34.50002	35.97609
24.81476	26.04034	27.24317	28.29858	29.70741	31.06739	32.2341	33.21596	34.49342	35.97907
24.82227	26.01919	27.214	28.32216	29.72441	31.06766	32.23429	33.23488	34.50096	35.97835
24.82074	26.03798	27.23954	28.28355	29.71681	31.06479	32.2366	33.23122	34.48968	35.97554
24.8125	26.03156	27.2342	28.30036	29.70405	31.06266	32.23774	33.22672	34.49156	35.97966
24.81415	26.02979	27.23985	28.30221	29.72648	31.03879	32.23265	33.22279	34.48743	35.96187
24.81736	26.04137	27.19772	28.31769	29.64084	31.01831	32.25095	33.2266	34.47434	35.95139
24.84362	26.03892	27.21922	28.31767	29.68708	31.06472	32.25237	33.21623	34.49683	35.96219
24.84698	26.01613	27.23348	28.31829	29.71971	31.06667	32.25375	33.23616	34.49221	35.96493
24.83417	26.03541	27.23357	28.32107	29.73839	31.05799	32.25339	33.24058	34.50196	35.9608
24.84547	26.00248	27.24133	28.32446	29.73114	31.06754	32.25327	33.24113	34.48448	35.98082
24.84313	26.02847	27.24195	28.3239	29.71745	31.03322	32.25357	33.23583	34.49828	35.97775

24.81118	26.02106	27.23308	28.32212	29.74173	31.05675	32.25563	33.2277	34.49672	35.9652
24.83937	26.03229	27.2233	28.32378	29.73293	31.0712	32.25651	33.22835	34.4911	35.98178
24.83968	26.03551	27.19356	28.32449	29.73041	31.07485	32.25429	33.22513	34.49157	35.97077
24.85107	26.01653	27.22939	28.31163	29.73854	31.06278	32.25487	33.22933	34.48339	35.96483
24.84729	26.02059	27.23915	28.32487	29.73893	31.07079	32.25665	33.22122	34.49482	35.97031
24.84875	26.02388	27.237	28.31141	29.72547	31.0666	32.25372	33.22168	34.49422	35.9541
24.85501	26.01008	27.23827	28.33034	29.74802	31.05694	32.25611	33.23516	34.49165	35.96283
24.85484	26.01001	27.23791	28.33377	29.73809	31.07494	32.2543	33.22775	34.50825	35.9572
24.85601	26.02354	27.21448	28.33006	29.72542	31.07863	32.25409	33.21998	34.48208	35.96373
24.85626	26.01767	27.23892	28.32377	29.73389	31.08568	32.25154	33.21728	34.51059	35.95893
24.82226	26.01956	27.22722	28.32543	29.73476	31.0808	32.25111	33.22849	34.50818	35.96228
24.86101	26.02187	27.22725	28.32973	29.72879	31.07885	32.25081	33.2262	34.47524	35.94954
24.86239	26.02335	27.19927	28.32259	29.72502	31.08328	32.25187	33.19167	34.50314	35.94179
24.84397	26.02267	27.23811	28.32269	29.69182	31.06561	32.2486	33.21995	34.50954	35.93664
24.84002	26.02447	27.22939	28.31544	29.71906	31.07744	32.24764	33.21681	34.51388	35.95577
24.84367	26.01844	27.22232	28.31638	29.67615	31.06464	32.24586	33.18127	34.48202	35.95738
24.86676	26.03255	27.23851	28.30879	29.70381	31.07157	32.21836	33.21094	34.50864	35.92245
24.87314	26.03144	27.2293	28.30799	29.71684	31.00063	32.23372	33.21725	34.49188	35.94796
24.86309	26.02579	27.23363	28.31391	29.72503	31.08533	32.20695	33.21965	34.50936	35.93917
24.8491	26.02857	27.23905	28.31221	29.70232	31.058	32.23138	33.22955	34.51727	35.92262
24.86565	26.03229	27.20608	28.31492	29.70394	31.0336	32.24517	33.22687	34.51402	35.94284
24.87552	26.02102	27.22684	28.31346	29.70951	31.06326	32.24457	33.21552	34.51413	35.92327
24.87665	26.02948	27.23709	28.31605	29.72999	31.06886	32.24548	33.2129	34.49249	35.94026
24.87817	25.95517	27.23522	28.30422	29.73364	31.05772	32.24696	33.22763	34.49544	35.9309
24.87622	26.01946	27.23671	28.28333	29.73039	31.06611	32.25005	33.21654	34.49563	35.91811
24.84248	26.03038	27.23236	28.30706	29.72611	31.0562	32.25579	33.23376	34.51312	35.88402
24.84842	26.02895	27.21141	28.3041	29.73995	31.07337	32.24046	33.23485	34.519	35.92444
24.84584	26.0267	27.23788	28.31431	29.73332	31.06817	32.24118	33.2362	34.51651	35.92224
24.84834	26.00966	27.2368	28.27631	29.74162	31.06119	32.25074	33.24486	34.51036	35.91997
24.84933	26.01911	27.18272	28.31229	29.74301	31.06638	32.24797	33.23359	34.50863	35.9048
24.85119	26.01792	27.23772	28.31472	29.72827	31.06817	32.25069	33.23257	34.49915	35.89738
24.89014	26.02233	27.23805	28.27377	29.73527	31.07209	32.24841	33.23874	34.5246	35.89148
24.88575	26.01099	27.2264	28.30784	29.73979	31.06959	32.24765	33.22522	34.52489	35.90401
24.88764	25.99422	27.23848	28.27376	29.72381	31.06785	32.24823	33.21556	34.51264	35.9024
24.89018	26.01812	27.24082	28.30309	29.73578	31.06931	32.24792	33.24052	34.52024	35.89905
24.89726	26.00993	27.22821	28.30291	29.74157	31.06939	32.25096	33.22043	34.52205	35.90065
24.90196	25.99588	27.24356	28.29915	29.73909	31.07615	32.24885	33.22113	34.51531	35.8704
24.9058	26.03051	27.23824	28.2961	29.72783	31.07927	32.24998	33.22004	34.51781	35.88129
24.89296	26.03261	27.22841	28.27227	29.72856	31.08357	32.25016	33.21103	34.50099	35.88763
24.90824	26.02152	27.24675	28.29853	29.71082	31.06712	32.25435	33.21319	34.52079	35.88721
24.90799	26.0315	27.24288	28.30178	29.68253	31.07788	32.25256	33.22232	34.49617	35.87247
24.86306	26.02438	27.24464	28.30531	29.72573	31.06353	32.24462	33.23366	34.49848	35.87421
24.8973	26.03724	27.24686	28.30326	29.7233	31.07634	32.25376	33.21709	34.4805	35.87846

24.89897	26.03959	27.26073	28.29388	29.70285	31.08913	32.25448	33.21939	34.52366	35.87112
24.90846	26.03492	27.25936	28.26872	29.72586	31.07927	32.25484	33.19318	34.51064	35.87789
24.90607	26.04363	27.25313	28.30557	29.72161	31.09164	32.25371	33.22208	34.51824	35.86992
24.90462	26.03918	27.25296	28.29733	29.69971	31.09162	32.25393	33.21899	34.52695	35.86424
24.89352	26.03752	27.25721	28.29674	29.71893	31.0916	32.25361	33.22267	34.51607	35.85375
24.90514	26.03611	27.26126	28.30144	29.7167	31.08684	32.25415	33.22263	34.52582	35.86258
24.88298	26.04105	27.26137	28.28141	29.71858	31.09228	32.25278	33.20797	34.52059	35.85569
24.89419	26.01266	27.27316	28.30339	29.72174	31.08772	32.25164	33.2091	34.51941	35.83931
24.90872	26.02653	27.26263	28.28881	29.71894	31.05481	32.24893	33.20873	34.51668	35.83357
24.90964	26.0276	27.25268	28.29805	29.73018	31.06214	32.24727	33.21623	34.52672	35.82918
24.90961	26.0332	27.24663	28.29426	29.72572	31.08507	32.24782	33.22516	34.52735	35.80583
24.91461	26.04009	27.26878	28.28393	29.65314	31.06358	32.25109	33.21001	34.52325	35.83441
24.90893	26.03225	27.26217	28.29163	29.71483	31.0771	32.24885	33.22968	34.53067	35.82149
24.9116	26.03775	27.27582	28.29669	29.73327	31.08033	32.24919	33.22367	34.4986	35.80535
24.91536	26.03701	27.25845	28.29246	29.73412	31.08222	32.2502	33.22345	34.52483	35.82806
24.9227	26.04355	27.26288	28.2784	29.70957	31.08115	32.24994	33.22859	34.48856	35.81432
24.92131	26.02993	27.25945	28.27511	29.72791	31.01997	32.24866	33.23149	34.52255	35.81228
24.91534	26.04552	27.25935	28.29098	29.72847	31.03748	32.24839	33.23284	34.48333	35.81524
24.90805	26.03817	27.27455	28.27047	29.72456	31.06687	32.24814	33.24256	34.48362	35.81469
24.92416	26.04451	27.25299	28.27555	29.72424	31.07719	32.24954	33.24598	34.52446	35.81723
24.92036	26.05017	27.2664	28.29631	29.69729	31.07679	32.24984	33.24465	34.50557	35.79082
24.90298	26.02366	27.19418	28.29219	29.71643	31.06102	32.24231	33.24751	34.5232	35.80614
24.92291	26.04458	27.27781	28.29078	29.72288	31.08433	32.24141	33.24829	34.51797	35.78164
24.92289	26.04996	27.25292	28.30472	29.72982	31.08457	32.23868	33.23914	34.49467	35.80381
24.91717	26.04056	27.27542	28.26827	29.71653	31.07836	32.23687	33.24493	34.49881	35.78131
24.91835	26.05545	27.27248	28.28648	29.72504	31.06123	32.23697	33.24251	34.51921	35.77812
24.91762	26.05107	27.27082	28.27083	29.7258	31.07928	32.23611	33.24194	34.5189	35.78202
24.92334	26.04277	27.26848	28.30165	29.72822	31.07731	32.22807	33.24152	34.48979	35.78449
24.90954	26.06044	27.28063	28.30074	29.6955	31.05315	32.22591	33.24625	34.51267	35.78415
24.90729	26.06306	27.2794	28.30453	29.71962	31.08126	32.22531	33.24518	34.51172	35.76509
24.91089	26.05933	27.26136	28.26192	29.71615	31.07371	32.22645	33.23057	34.51039	35.76999
24.92844	26.06162	27.28068	28.28157	29.71099	31.07553	32.22441	33.22544	34.51423	35.76981
24.92912	26.05387	27.27682	28.30762	29.73052	31.08684	32.2308	33.22292	34.50929	35.76751
24.93011	26.04626	27.27659	28.30347	29.71963	31.08402	32.22943	33.23432	34.50391	35.7674
24.93094	26.04749	27.28125	28.3048	29.71987	31.0791	32.22804	33.20919	34.51618	35.75853
24.92483	26.04606	27.23279	28.28392	29.71308	31.08087	32.217	33.22782	34.50548	35.75921
24.92886	26.04419	27.27951	28.28858	29.71663	31.08114	32.21317	33.20094	34.52113	35.74924
24.92959	26.04651	27.28005	28.30226	29.71251	31.08015	32.21333	33.18532	34.51744	35.75749
24.9156	26.01849	27.27455	28.30706	29.72255	31.08144	32.21425	33.23502	34.52237	35.73663
24.93452	26.05721	27.25973	28.30892	29.724	31.08024	32.20678	33.23446	34.52195	35.71773
24.93562	26.03366	27.2583	28.31169	29.72779	31.08723	32.19867	33.23732	34.52253	35.73556
24.93438	26.02966	27.2805	28.30337	29.70216	31.08241	32.22965	33.23494	34.52207	35.75379
24.93595	26.02519	27.26276	28.30605	29.71102	31.07351	32.23282	33.22838	34.51791	35.73029

24.93093	26.06912	27.27503	28.30028	29.71983	31.07984	32.24206	33.23869	34.52801	35.74782
24.90753	26.05716	27.27384	28.30501	29.71134	31.07976	32.246	33.23604	34.52721	35.74341
24.9323	26.06098	27.26781	28.30542	29.71182	31.06027	32.24744	33.24585	34.52063	35.73618
24.92164	26.0712	27.27485	28.30524	29.71157	31.07012	32.24655	33.24238	34.52507	35.7282
24.93129	26.06888	27.27116	28.3083	29.70892	31.07842	32.24563	33.2128	34.51371	35.73902
24.93374	26.08239	27.27046	28.30179	29.72046	31.08019	32.24517	33.21152	34.50443	35.73583
24.93378	26.07415	27.23633	28.30674	29.70095	31.06966	32.24685	33.21345	34.51634	35.69992
24.90439	26.04509	27.27579	28.30869	29.71608	31.07006	32.24617	33.22644	34.52742	35.73015
24.90476	26.07978	27.26879	28.30013	29.716	31.06079	32.24012	33.23274	34.5249	35.72664
24.93268	26.05098	27.26074	28.29326	29.73742	31.04883	32.23953	33.2374	34.51784	35.71475
24.93905	26.07836	27.2683	28.31323	29.72741	31.02197	32.23526	33.24165	34.50907	35.70605
24.93452	26.06632	27.26886	28.28585	29.742	31.0768	32.23353	33.2382	34.52018	35.7152
24.93357	26.067	27.25274	28.30936	29.74102	31.08107	32.23055	33.24287	34.51894	35.68094
24.92662	26.05822	27.26891	28.29586	29.74593	31.08078	32.22185	33.22989	34.517	35.71065
24.93854	26.06453	27.24025	28.30123	29.73826	31.07107	32.23335	33.24619	34.50678	35.71405
24.94214	26.06132	27.27861	28.3119	29.74142	31.07877	32.23062	33.24259	34.51982	35.69742
24.92593	26.06219	27.27247	28.31866	29.74675	31.07781	32.22891	33.22966	34.52602	35.6939
24.94422	26.06171	27.26748	28.30943	29.74704	31.07963	32.22536	33.2241	34.499	35.70043
24.92408	26.03263	27.28234	28.2972	29.72659	31.07784	32.22017	33.22247	34.49826	35.6836
24.90578	26.06694	27.27997	28.31509	29.73808	31.07767	32.21944	33.23682	34.50929	35.69775
24.90256	26.03712	27.27361	28.30311	29.73959	31.07663	32.22136	33.24021	34.5089	35.70392
24.94746	26.06312	27.27672	28.31488	29.73561	31.05764	32.21356	33.23374	34.51102	35.69418
24.93841	26.06696	27.2721	28.30712	29.73938	31.04579	32.21447	33.23196	34.52358	35.6831
24.94071	26.06488	27.25836	28.31805	29.71455	31.05984	32.21519	33.24386	34.5217	35.67924
24.93519	26.06078	27.26812	28.29389	29.72729	31.05991	32.21742	33.25183	34.50328	35.64973
24.93425	26.06825	27.27151	28.31711	29.72011	31.06404	32.21925	33.24745	34.51267	35.6684
24.94663	26.05446	27.24973	28.31741	29.72291	31.07292	32.22187	33.24772	34.51354	35.67846
24.94821	26.07438	27.24446	28.32151	29.71781	31.06082	32.22372	33.25689	34.52108	35.67682
24.95119	26.04093	27.26969	28.32073	29.71053	31.04144	32.23154	33.23188	34.51161	35.67034
24.95378	26.06983	27.26951	28.29112	29.71588	31.06781	32.23371	33.25222	34.4857	35.66779
24.89485	26.06447	27.25948	28.32711	29.72519	31.06792	32.23599	33.2519	34.51469	35.66098
24.89499	26.03148	27.26042	28.31233	29.71671	31.07154	32.23726	33.24773	34.51425	35.66597
24.95586	26.06743	27.26764	28.32595	29.72886	31.0672	32.23603	33.24994	34.50983	35.66638
24.95578	26.06249	27.26564	28.32269	29.70559	31.06373	32.23848	33.24743	34.5067	35.64825
24.89585	26.02548	27.24485	28.31467	29.72152	31.03831	32.23771	33.24263	34.49988	35.66399
24.94517	26.05372	27.26501	28.31173	29.73051	31.06085	32.23228	33.23838	34.50484	35.64631
24.9453	26.06335	27.25199	28.30692	29.70766	31.06628	32.23262	33.24043	34.4768	35.66083
24.94541	26.05938	27.25921	28.29449	29.73267	31.06403	32.23869	33.25278	34.46171	35.66473
24.95679	26.06922	27.26061	28.30527	29.73363	31.06475	32.24001	33.23772	34.48933	35.65243
24.9581	26.06433	27.25724	28.3163	29.72932	31.05417	32.24087	33.23746	34.49384	35.64708
24.96058	26.07003	27.25797	28.29591	29.71747	31.05838	32.24126	33.23223	34.50649	35.66311
24.96657	26.01528	27.21558	28.29064	29.72948	31.05814	32.2394	33.23182	34.50376	35.656
24.9661	26.04869	27.25132	28.31092	29.73875	31.04838	32.23987	33.23593	34.50687	35.65761

24.93998	26.06628	27.22526	28.29755	29.7327	31.0681	32.21936	33.24447	34.50795	35.65895
24.93914	26.06317	27.25969	28.30698	29.71037	31.04748	32.21018	33.23987	34.45169	35.61244
24.95645	26.06719	27.23102	28.30445	29.71263	31.05279	32.21502	33.23933	34.48485	35.65401
24.92751	26.059	27.25573	28.29893	29.71033	31.05444	32.21152	33.24154	34.48221	35.6511
24.96237	26.05994	27.25001	28.31353	29.73031	31.0631	32.20958	33.19974	34.4934	35.65009
24.96345	26.05021	27.25523	28.31597	29.72658	31.04991	32.18429	33.21457	34.51037	35.6528
24.96463	26.06418	27.25403	28.31668	29.69839	31.06679	32.22588	33.21263	34.504	35.63238
24.9531	26.0341	27.23914	28.31792	29.73118	31.06709	32.22657	33.20681	34.49984	35.62993
24.96576	26.06472	27.24741	28.31484	29.7296	31.04612	32.22481	33.23228	34.50972	35.64873
24.96854	26.04436	27.24661	28.30349	29.70903	31.07276	32.22261	33.23655	34.50181	35.6438
24.97195	26.06942	27.24261	28.31975	29.72319	31.05845	32.22534	33.22964	34.49421	35.64546
24.9294	26.03841	27.22482	28.32063	29.72116	31.03839	32.21577	33.20803	34.47594	35.64244
24.95397	26.03636	27.23854	28.32073	29.71899	31.05482	32.20842	33.2362	34.51347	35.6459
24.95319	26.05473	27.23404	28.31797	29.72029	31.06945	32.21412	33.23542	34.49296	35.64393
24.98148	26.06071	27.23562	28.31444	29.71426	31.06888	32.21424	33.24014	34.50818	35.64329
24.97498	26.05372	27.23806	28.28829	29.72047	31.05823	32.21154	33.24027	34.50899	35.65054
24.96046	26.05251	27.24389	28.30679	29.71483	31.04172	32.21383	33.24506	34.49918	35.64829
24.96231	26.05315	27.24059	28.32702	29.72893	31.0667	32.21446	33.23979	34.50415	35.61393
24.95452	26.05148	27.23802	28.32198	29.73819	31.06882	32.21507	33.24117	34.49992	35.64744
24.97978	26.04883	27.24102	28.32003	29.73951	31.06678	32.21617	33.24398	34.47193	35.65442
24.98089	26.03394	27.24579	28.32423	29.72861	31.06567	32.21608	33.23069	34.4977	35.63776
24.9746	26.02387	27.23329	28.33043	29.74468	31.05906	32.21464	33.22838	34.48964	35.63526
24.98064	26.04444	27.21767	28.32706	29.726	31.04839	32.21892	33.25019	34.47986	35.63113
24.97958	26.04082	27.2431	28.32898	29.74647	31.04972	32.21609	33.24983	34.49703	35.64994
24.97157	26.03542	27.23368	28.33297	29.7419	31.05134	32.21603	33.21906	34.49564	35.65167
24.98904	26.01805	27.23953	28.31193	29.74487	31.06524	32.17554	33.21665	34.49127	35.64914
24.98633	26.04285	27.24896	28.33842	29.74334	31.04496	32.17635	33.23698	34.49573	35.63495
24.9842	26.05156	27.24416	28.32597	29.73729	31.06154	32.21309	33.24875	34.49063	35.65393
24.96722	26.04453	27.24761	28.33171	29.72695	31.05134	32.2144	33.24904	34.4962	35.65122
24.96573	26.05259	27.24015	28.32286	29.73466	31.05421	32.21525	33.22581	34.49248	35.63862
24.96565	26.04331	27.22772	28.3256	29.74607	31.03528	32.21642	33.24016	34.48613	35.65182
24.97544	26.0506	27.23532	28.31738	29.75082	31.05462	32.2199	33.25051	34.49515	35.65042
24.98202	26.04644	27.24253	28.3109	29.74989	31.06786	32.22348	33.24031	34.4758	35.6556
24.9835	26.03723	27.23384	28.31988	29.73726	31.06601	32.22448	33.23813	34.4937	35.65328
24.98473	26.0306	27.23992	28.3057	29.74843	31.04825	32.21385	33.25479	34.49481	35.62009
24.98784	26.02492	27.24557	28.33227	29.72382	31.04437	32.22868	33.2506	34.49604	35.63899
24.98608	26.03198	27.24636	28.32674	29.74567	31.06314	32.22763	33.23328	34.49584	35.67136
24.97252	26.01914	27.22739	28.31273	29.74191	31.04671	32.22568	33.25567	34.48768	35.66996
24.95962	26.03206	27.23319	28.32584	29.72808	31.03463	32.22303	33.24429	34.49286	35.66961
24.98781	26.03317	27.21621	28.31558	29.74858	31.04758	32.22292	33.24967	34.46041	35.66288
24.98855	26.03556	27.23112	28.32716	29.74631	31.05139	32.22052	33.25665	34.49554	35.67597
24.98707	26.03162	27.23985	28.31641	29.75144	31.04826	32.21887	33.25508	34.48702	35.68016
24.98842	26.03272	27.18724	28.32913	29.7543	31.04758	32.21648	33.24189	34.49006	35.67031

24.98818	26.03381	27.2246	28.32639	29.76232	31.04683	32.21402	33.25029	34.49238	35.68466
24.92389	26.03287	27.24043	28.33041	29.76616	31.04694	32.21605	33.24583	34.47432	35.68653
24.97649	25.99502	27.23047	28.31442	29.74858	31.04885	32.21698	33.24319	34.46196	35.67428
24.97549	25.996	27.23507	28.32266	29.7634	31.03995	32.21797	33.23676	34.47679	35.67247
24.97564	26.02796	27.24452	28.33033	29.75351	31.02736	32.21703	33.23143	34.48927	35.69507
24.97434	26.02567	27.21595	28.30662	29.75758	31.034	32.21539	33.2245	34.4927	35.66996
24.98438	26.00142	27.24589	28.32673	29.77145	31.05387	32.19862	33.24012	34.48092	35.63788
24.98414	26.02319	27.24177	28.32918	29.76696	31.05966	32.20062	33.24091	34.48968	35.68524
24.98367	26.00327	27.22682	28.33074	29.74225	31.05718	32.18603	33.22389	34.48795	35.67406
24.97643	26.03328	27.24681	28.33126	29.76124	31.05511	32.17507	33.23627	34.47949	35.68777
24.97551	26.03276	27.2495	28.31993	29.75969	31.05764	32.19483	33.23616	34.48846	35.68519
24.97429	26.03144	27.24538	28.30809	29.75807	31.04759	32.20163	33.21676	34.47543	35.67981
24.97387	26.03266	27.24733	28.32364	29.74617	31.05828	32.20144	33.23704	34.46073	35.68959
24.97373	26.02844	27.24411	28.31467	29.75817	31.05489	32.20154	33.22865	34.48991	35.66792
24.97347	26.01828	27.2417	28.32879	29.73768	31.0607	32.1596	33.20205	34.47449	35.67823
24.97538	26.03483	27.24766	28.32314	29.75468	31.04952	32.16408	33.235	34.47875	35.68173
24.97145	26.03378	27.25422	28.32541	29.72734	31.06332	32.18748	33.24126	34.46751	35.69344
24.97539	26.02742	27.25668	28.29995	29.74937	31.04608	32.19268	33.22347	34.48064	35.68137
24.92548	26.03885	27.25835	28.33023	29.75509	31.06029	32.21526	33.22426	34.45394	35.69247
24.94053	25.9969	27.23967	28.32771	29.74992	31.05996	32.21735	33.20502	34.49292	35.68131
24.93771	26.03548	27.24248	28.32115	29.73386	31.05311	32.22003	33.23918	34.49058	35.68772
24.89767	26.02894	27.24737	28.33325	29.75371	31.0408	32.22089	33.24911	34.48956	35.69443
24.92178	25.98629	27.25065	28.33047	29.74772	31.05955	32.21977	33.23712	34.49793	35.68343
24.94345	26.02302	27.24678	28.29949	29.77003	31.04731	32.22249	33.23358	34.48892	35.68495
24.94114	26.02042	27.25623	28.33128	29.75402	31.06398	32.22292	33.23307	34.49247	35.68318
24.94001	26.02295	27.25782	28.32369	29.7633	31.05936	32.22301	33.23576	34.49001	35.70484
24.95134	26.02681	27.25225	28.29414	29.77706	31.06654	32.22188	33.22622	34.47274	35.70184
24.95292	26.01712	27.22793	28.32749	29.76488	31.06689	32.2208	33.23158	34.49149	35.6655
24.94077	26.00261	27.25025	28.327	29.76837	31.04926	32.21921	33.24693	34.47351	35.7126
24.94158	26.0075	27.2553	28.30958	29.75868	31.06023	32.21712	33.2402	34.48551	35.70091
24.94462	26.0314	27.2354	28.32342	29.76698	31.07019	32.21441	33.225	34.48539	35.71881
24.94184	26.03336	27.24688	28.33168	29.78426	31.05773	32.21172	33.24694	34.48429	35.69094
24.93851	26.03383	27.24814	28.32121	29.72956	31.03827	32.20979	33.24449	34.47525	35.71306
24.93439	26.0295	27.23801	28.30741	29.76644	31.03547	32.20883	33.24657	34.48349	35.72653
24.9351	26.01917	27.24397	28.32638	29.77274	31.03724	32.2064	33.24337	34.48697	35.68563
24.92886	26.01834	27.24511	28.30922	29.77555	31.04145	32.20589	33.23793	34.48779	35.7234
24.92195	26.0261	27.24481	28.30876	29.75294	31.05117	32.20515	33.23726	34.47972	35.72695
24.90816	26.01658	27.24571	28.32949	29.75318	31.05297	32.20562	33.23906	34.48459	35.73099
24.92449	26.03127	27.23973	28.31613	29.76329	31.06288	32.20374	33.23322	34.49121	35.72308
24.92207	26.03859	27.22919	28.32127	29.73537	31.06283	32.20195	33.23314	34.48237	35.73328
24.90414	26.04198	27.24486	28.32935	29.75734	31.03635	32.18961	33.24044	34.47608	35.73441
24.89653	26.003	27.24724	28.33017	29.75404	31.05959	32.18849	33.22854	34.48819	35.73234
24.89995	26.03054	27.24667	28.32983	29.72532	31.05856	32.17893	33.22942	34.48546	35.72062

24.89735	26.03807	27.25048	28.31915	29.74437	31.05657	32.18327	33.20789	34.47794	35.71657
24.89423	26.0368	27.2324	28.32548	29.75025	31.05483	32.18346	33.21241	34.48566	35.70875
24.90351	26.01946	27.21855	28.32269	29.71622	31.05666	32.18315	33.20177	34.48521	35.72589
24.90275	26.02357	27.24123	28.32295	29.73734	31.04525	32.19748	33.24118	34.47716	35.7292
24.89994	26.01527	27.24993	28.27433	29.74845	31.02911	32.19913	33.23772	34.48433	35.73426
24.88958	26.03041	27.25216	28.32026	29.742	31.02321	32.20014	33.23354	34.48729	35.74622
24.88713	26.02277	27.24603	28.30401	29.74639	31.0321	32.20077	33.23423	34.47971	35.73852
24.87248	26.03002	27.22957	28.31597	29.7542	31.05201	32.20076	33.20763	34.48188	35.74654
24.8847	25.99046	27.25012	28.32318	29.74799	31.04402	32.19939	33.24699	34.4798	35.74896
24.88199	26.02802	27.24696	28.32058	29.73056	31.04638	32.19768	33.21733	34.49139	35.74688
24.86788	26.03356	27.24526	28.31994	29.76184	31.05008	32.194	33.23897	34.43518	35.7368
24.84676	26.03281	27.23852	28.31627	29.74425	31.05045	32.19392	33.24013	34.49321	35.72806
24.87012	26.03759	27.23783	28.31964	29.76068	31.05058	32.19434	33.23707	34.48626	35.75141
24.83855	25.99988	27.23143	28.31614	29.76707	31.04904	32.19531	33.24057	34.46292	35.74793
24.86507	26.03414	27.24284	28.31356	29.74513	31.04435	32.19729	33.22232	34.48936	35.74432
24.86286	26.0279	27.23248	28.30875	29.74884	31.03319	32.17706	33.23571	34.48153	35.72927
24.85842	26.0316	27.21013	28.31664	29.76251	31.03112	32.1767	33.23594	34.46813	35.74131
24.84903	26.04157	27.24988	28.31218	29.72107	31.03758	32.18614	33.24546	34.47856	35.75553
24.84627	26.0322	27.2374	28.31334	29.74368	31.04574	32.18443	33.23897	34.47151	35.75367
24.84324	26.04831	27.24811	28.31293	29.75408	31.02174	32.19096	33.22556	34.47003	35.74695
24.84029	26.03692	27.23354	28.30513	29.75624	31.04808	32.19146	33.23371	34.47957	35.73608
24.84263	26.04714	27.24224	28.29369	29.74133	31.04722	32.19344	33.23388	34.49126	35.74437
24.83862	26.04321	27.2386	28.30639	29.70521	31.03723	32.19226	33.23155	34.49386	35.74907
24.83454	26.01831	27.24453	28.29561	29.74397	31.04858	32.19129	33.22992	34.49189	35.75846
24.83311	26.03713	27.23608	28.29717	29.7244	31.04959	32.19143	33.23327	34.48499	35.75681
24.79238	26.03524	27.2423	28.2907	29.70909	31.03767	32.16438	33.21469	34.49231	35.75314
24.77864	26.03428	27.23217	28.29785	29.73752	31.03948	32.17941	33.19836	34.50148	35.74233
24.80584	26.02286	27.23386	28.30022	29.70926	31.0387	32.17873	33.22012	34.48477	35.74867
24.8303	26.02292	27.2418	28.30031	29.72899	31.04969	32.17744	33.22682	34.49724	35.74792
24.82773	26.00014	27.21178	28.29577	29.7203	31.03869	32.17641	33.22349	34.50799	35.74754
24.8003	26.00369	27.23856	28.28776	29.71834	31.03982	32.17521	33.22926	34.5067	35.74147
24.79672	26.02726	27.23722	28.30442	29.74067	31.05441	32.17589	33.23211	34.50432	35.747
24.80987	26.02186	27.23543	28.29921	29.73808	31.04397	32.17459	33.23156	34.50282	35.74441
24.80602	26.03257	27.24312	28.29022	29.74337	31.05287	32.17116	33.23765	34.48932	35.74981
24.81754	26.03058	27.23348	28.30235	29.74652	31.03907	32.17291	33.19686	34.49093	35.74403
24.81216	26.02427	27.23848	28.28049	29.73519	31.05144	32.16818	33.23745	34.50388	35.74497
24.80908	26.02555	27.21392	28.29348	29.7432	31.05616	32.1645	33.20477	34.49717	35.74645
24.80779	26.02329	27.2366	28.29589	29.75834	31.05199	32.17976	33.22417	34.48761	35.75496
24.79407	26.00208	27.24124	28.28973	29.74964	31.05477	32.18083	33.22102	34.50343	35.75524
24.7917	25.98943	27.23855	28.29941	29.75507	31.0563	32.17993	33.23177	34.49773	35.7506
24.78475	26.02374	27.2445	28.29566	29.742	31.05849	32.18045	33.21205	34.48648	35.75151
24.78142	26.02901	27.23354	28.28198	29.75271	31.03063	32.18067	33.22708	34.49631	35.76015
24.77953	26.00689	27.23488	28.29679	29.75104	31.04953	32.16799	33.22565	34.50785	35.73957

24.78513	26.0322	27.23643	28.29372	29.72487	31.05848	32.17622	33.22121	34.50508	35.74629
24.78428	26.03016	27.22462	28.29347	29.73198	31.05734	32.17236	33.22482	34.49157	35.75187
24.79094	26.0206	27.23979	28.29673	29.73988	31.05521	32.17094	33.22016	34.49782	35.75279
24.79149	26.03074	27.22277	28.26069	29.70267	31.05555	32.16912	33.2228	34.50724	35.75677
24.74821	26.01644	27.24096	28.27913	29.73759	31.04609	32.14701	33.22242	34.49939	35.75946
24.78007	26.02716	27.23453	28.30003	29.7271	31.05497	32.15169	33.22153	34.50654	35.74776
24.77728	26.02128	27.23789	28.29921	29.7377	31.04311	32.14758	33.20711	34.51631	35.75969
24.7754	26.02733	27.2458	28.30227	29.7167	31.04849	32.15431	33.21653	34.51725	35.73683
24.78357	25.99556	27.24356	28.30394	29.72444	31.05808	32.19005	33.21124	34.52154	35.75511
24.78059	26.02119	27.22573	28.30451	29.7188	31.05848	32.18946	33.22017	34.50295	35.75809
24.7792	26.02379	27.20239	28.28416	29.73154	31.05226	32.19019	33.20574	34.52456	35.75537
24.77775	26.02584	27.23991	28.30714	29.72813	31.0387	32.18409	33.20311	34.5162	35.74166
24.7595	26.03045	27.22157	28.29466	29.73933	31.05473	32.18959	33.21396	34.52617	35.75248
24.75775	26.03124	27.24148	28.30006	29.74184	31.02423	32.19043	33.20289	34.53746	35.74773
24.76607	26.01114	27.25625	28.30772	29.73976	31.05263	32.18854	33.21003	34.53722	35.72769
24.75449	26.03591	27.24925	28.2837	29.74177	31.06541	32.16108	33.21288	34.53803	35.74925
24.75212	26.02117	27.25205	28.30695	29.74777	31.04668	32.16169	33.22738	34.53097	35.73888
24.75008	26.03397	27.25995	28.29492	29.75452	31.05221	32.16258	33.21378	34.5211	35.73044
24.7409	26.03501	27.25025	28.29216	29.75638	31.05091	32.16367	33.21173	34.51752	35.7344
24.74057	26.03819	27.26137	28.30155	29.7506	31.04228	32.16283	33.19879	34.5403	35.71206
24.74269	26.03967	27.21443	28.30415	29.74934	31.04355	32.17658	33.20834	34.52458	35.71591
24.74201	26.03513	27.2514	28.30228	29.74106	31.05889	32.17921	33.22899	34.54713	35.73408
24.74256	26.04385	27.22327	28.29434	29.74149	31.05102	32.17539	33.23376	34.54638	35.72141
24.74243	26.0384	27.25279	28.3034	29.73896	31.05622	32.17669	33.23265	34.50552	35.7258
24.75968	26.02471	27.24505	28.31301	29.75117	31.05354	32.18408	33.22506	34.54615	35.72554
24.75049	26.04374	27.24567	28.29354	29.74596	31.05247	32.18627	33.22449	34.55042	35.71576
24.74608	26.04352	27.2483	28.31465	29.7222	31.05363	32.20547	33.23121	34.51464	35.71334
24.75785	26.03761	27.23991	28.28932	29.72271	31.03643	32.20618	33.22638	34.52264	35.6935
24.74351	26.04017	27.24712	28.30911	29.73301	31.05761	32.21003	33.22905	34.54774	35.70517
24.74528	26.01869	27.24595	28.30457	29.73492	31.02335	32.20779	33.20476	34.55273	35.70048
24.75159	26.05475	27.24912	28.2974	29.73326	31.03366	32.20874	33.23283	34.55768	35.69289
24.75278	26.04452	27.22966	28.31467	29.72574	31.06009	32.20946	33.22759	34.53384	35.69298
24.7498	26.05849	27.2495	28.31536	29.72974	31.04292	32.17029	33.23409	34.5467	35.69375
24.75529	26.03902	27.25081	28.29061	29.72281	31.05037	32.17338	33.21101	34.54593	35.69488
24.75335	26.05428	27.24564	28.31527	29.72668	31.05914	32.17983	33.21539	34.55135	35.69511
24.73459	26.04247	27.21839	28.31211	29.73312	31.05798	32.21163	33.19752	34.54939	35.69423
24.73493	26.05952	27.2566	28.31819	29.71008	31.06005	32.21111	33.22635	34.54998	35.67591
24.73641	26.02504	27.21806	28.26886	29.69813	31.05396	32.20992	33.22559	34.55926	35.68035
24.73425	26.06034	27.24136	28.29757	29.72119	31.05772	32.21009	33.21951	34.55745	35.68367
24.73726	26.06024	27.24155	28.30792	29.73074	31.05722	32.20251	33.21873	34.56368	35.67689
24.73275	26.05995	27.24125	28.27918	29.73952	31.05266	32.2047	33.21843	34.56617	35.68337
24.7334	26.0539	27.25861	28.30034	29.73272	31.04353	32.20482	33.22849	34.56694	35.67706
24.72864	26.05944	27.26608	28.31601	29.74388	31.06211	32.19131	33.22501	34.56054	35.66838

24.74622	26.03744	27.25274	28.30854	29.71967	31.0648	32.20831	33.20209	34.54283	35.67604
24.74616	26.06048	27.25106	28.30193	29.74773	31.04277	32.20849	33.2119	34.56368	35.66682
24.74748	26.05042	27.25108	28.32731	29.73236	31.04754	32.20929	33.21131	34.57456	35.68582
24.74851	26.05576	27.26542	28.33295	29.71829	31.06396	32.2063	33.22064	34.5771	35.67229
24.74813	26.05321	27.254	28.30479	29.73494	31.06132	32.20744	33.22396	34.5612	35.67322
24.73738	26.0425	27.24348	28.32797	29.74805	31.05073	32.20784	33.22562	34.57677	35.66835
24.72383	26.0534	27.26468	28.33122	29.74007	31.05909	32.2064	33.23122	34.57942	35.66937
24.72323	26.05712	27.25757	28.3357	29.74659	31.05746	32.19335	33.23118	34.57294	35.67086
24.72932	26.04824	27.24864	28.32458	29.73411	31.02921	32.19434	33.23111	34.58548	35.65227
24.7288	26.04848	27.24089	28.31757	29.70979	31.0359	32.19577	33.23259	34.58518	35.67564
24.75166	26.05201	27.23307	28.32826	29.74054	31.03621	32.19467	33.23427	34.54223	35.61014
24.75136	26.05365	27.24044	28.3352	29.71334	31.0508	32.2018	33.20695	34.58339	35.66523
24.75087	26.04566	27.24597	28.32084	29.73986	31.04418	32.20457	33.23539	34.53088	35.66954
24.75309	26.04593	27.25439	28.33408	29.73688	31.0572	32.21094	33.22926	34.5677	35.66439
24.72397	26.05711	27.24719	28.33617	29.71756	31.05746	32.21274	33.23181	34.58639	35.66187
24.73493	26.01363	27.23514	28.32126	29.7241	31.04999	32.21156	33.22811	34.58278	35.65583
24.75189	26.05621	27.21755	28.31889	29.72601	31.06139	32.1939	33.23235	34.58926	35.66136
24.75918	26.05834	27.25479	28.33804	29.72655	31.05366	32.21967	33.24075	34.59623	35.64528
24.76144	26.02426	27.25031	28.33577	29.72955	31.04618	32.22124	33.23957	34.59935	35.62696
24.76076	26.06976	27.2537	28.33518	29.72349	31.02718	32.21322	33.22343	34.60221	35.64805
24.72243	26.05866	27.25001	28.32952	29.72413	31.02806	32.21672	33.20851	34.59046	35.64047
24.76369	26.07036	27.25725	28.33808	29.73797	31.05518	32.21857	33.24151	34.57592	35.63825
24.74706	26.06704	27.24741	28.32688	29.73112	31.04987	32.22195	33.22652	34.58407	35.62675
24.75353	26.07221	27.24709	28.31948	29.72959	31.05147	32.22473	33.23329	34.59646	35.6322
24.69309	26.06449	27.2586	28.32598	29.73247	31.06293	32.23154	33.23574	34.59457	35.5812
24.76891	26.07099	27.22502	28.33634	29.74596	31.05413	32.23529	33.23541	34.5878	35.62296
24.77369	26.07102	27.26223	28.32879	29.75185	31.06509	32.23662	33.24149	34.61325	35.62179
24.77521	26.05017	27.2619	28.33558	29.7484	31.06573	32.2139	33.23392	34.60625	35.61916
24.7771	26.05675	27.24556	28.32621	29.75416	31.05959	32.21718	33.22698	34.60497	35.60478
24.78034	26.05007	27.26322	28.32979	29.75281	31.06713	32.21538	33.21191	34.61523	35.60561
24.78418	26.0583	27.24621	28.31877	29.72371	31.05806	32.24272	33.21475	34.60996	35.58125
24.78189	26.05475	27.24869	28.32642	29.75299	31.06732	32.24135	33.21216	34.6159	35.55356
24.78883	26.05359	27.25813	28.32598	29.74419	31.02658	32.2362	33.22314	34.61403	35.57188
24.79077	26.03353	27.24467	28.31946	29.74335	31.06902	32.2315	33.23097	34.61582	35.59943
24.7917	26.05382	27.25839	28.30788	29.73515	31.06969	32.23137	33.22493	34.61102	35.59052
24.77175	26.03571	27.25683	28.28847	29.73923	31.06947	32.23492	33.22591	34.60166	35.59772
24.7712	26.05209	27.24556	28.32057	29.75016	31.06716	32.23316	33.21676	34.61759	35.57078
24.7839	26.06059	27.24081	28.3137	29.74186	31.0693	32.23334	33.22196	34.60898	35.57871
24.78436	26.03931	27.25491	28.28657	29.74702	31.04657	32.22892	33.23344	34.61418	35.59287
24.78894	26.05874	27.25397	28.31678	29.73388	31.06865	32.2327	33.21537	34.61723	35.58372
24.78797	26.05643	27.24618	28.31945	29.74188	31.0484	32.23524	33.21831	34.6215	35.5575
24.78865	26.05725	27.26051	28.30694	29.7458	31.03714	32.21781	33.20468	34.62029	35.56083
24.78623	26.04077	27.25692	28.32066	29.73883	31.04696	32.21466	33.2403	34.603	35.58767

24.78935	26.05142	27.24233	28.32294	29.74853	31.07379	32.2046	33.23979	34.59076	35.53941
24.79652	26.06362	27.22429	28.25958	29.74201	31.06441	32.24442	33.24011	34.60674	35.5902
24.79971	26.0561	27.26068	28.29777	29.75847	31.07609	32.24712	33.23358	34.60747	35.56267
24.78855	26.05758	27.2618	28.31803	29.75067	31.05139	32.24943	33.23986	34.60242	35.57531
24.79231	26.05677	27.25841	28.31599	29.75199	31.07831	32.25371	33.24099	34.608	35.5608
24.78832	26.04296	27.24981	28.30973	29.75786	31.0699	32.25426	33.24153	34.5964	35.58124
24.81356	26.05991	27.25851	28.30124	29.76388	31.07495	32.25072	33.23013	34.60038	35.59227
24.8189	25.97916	27.25669	28.31043	29.76249	31.07588	32.25087	33.21661	34.61835	35.58787
24.82192	26.0574	27.2573	28.28259	29.75715	31.04154	32.25865	33.23407	34.56472	35.55885
24.79284	26.04915	27.25381	28.30849	29.73297	31.04138	32.26192	33.25167	34.62287	35.58326
24.80187	26.02717	27.23542	28.29202	29.74203	31.08077	32.26032	33.24903	34.60885	35.56948
24.80198	26.01415	27.25482	28.299	29.76236	31.06173	32.25876	33.2464	34.61455	35.55863
24.8268	26.03775	27.24766	28.3112	29.76178	31.07945	32.2585	33.22289	34.61797	35.57464
24.82157	26.04418	27.25999	28.30051	29.74858	31.07387	32.25743	33.25118	34.61702	35.53976
24.8167	26.04019	27.26836	28.3126	29.75941	31.059	32.25853	33.25334	34.60591	35.56533
24.83268	26.0332	27.25551	28.29156	29.74166	31.04963	32.26235	33.25592	34.62461	35.5583
24.83457	26.02432	27.26078	28.30284	29.75532	31.08487	32.26941	33.23958	34.6219	35.5443
24.82923	26.03788	27.26727	28.28566	29.74223	31.07909	32.27688	33.23416	34.60643	35.54878
24.81323	26.02953	27.24577	28.27801	29.75015	31.08488	32.2771	33.22709	34.62807	35.52866
24.83805	26.03953	27.26905	28.28793	29.74647	31.07716	32.27879	33.23677	34.62472	35.56664
24.83743	26.01305	27.26906	28.30726	29.75161	31.06101	32.28101	33.23756	34.61958	35.56439
24.83902	26.04486	27.26569	28.30895	29.74795	31.08423	32.2815	33.23981	34.62649	35.55793
24.82402	26.02899	27.26632	28.2791	29.75534	31.08798	32.27622	33.22184	34.62325	35.53908
24.83233	26.04056	27.2608	28.29735	29.74891	31.08182	32.27588	33.23912	34.61582	35.5293
24.83269	26.01922	27.26439	28.30441	29.74702	31.07552	32.27589	33.21888	34.62317	35.56606
24.84603	26.04674	27.26439	28.28266	29.7388	31.06849	32.27935	33.23046	34.61777	35.56001
24.84444	26.04708	27.24067	28.29677	29.76676	31.08394	32.28073	33.23544	34.62218	35.55486
24.83929	26.04062	27.26683	28.29629	29.71139	31.08704	32.28173	33.21924	34.61011	35.55033
24.84885	26.02717	27.26437	28.27914	29.75507	31.07332	32.28281	33.23556	34.61643	35.54822
24.8506	26.04201	27.24169	28.29114	29.75499	31.0739	32.28206	33.23358	34.59536	35.53593
24.85052	26.0437	27.26496	28.27996	29.76509	31.0613	32.27819	33.22911	34.60971	35.54529
24.85159	26.02505	27.24594	28.28497	29.77044	31.07043	32.27937	33.22492	34.582	35.54899
24.83484	26.02241	27.26573	28.28072	29.77736	31.07016	32.27819	33.22458	34.61115	35.55031
24.83604	25.99545	27.25684	28.20499	29.78103	31.07888	32.27677	33.24063	34.61375	35.52863
24.78076	26.03024	27.26447	28.28002	29.78244	31.08156	32.27428	33.23291	34.59341	35.53717
24.83965	26.03139	27.23062	28.26469	29.77647	31.07915	32.27274	33.21523	34.58331	35.53075
24.83707	26.02599	27.26519	28.28015	29.7604	31.07838	32.26943	33.23057	34.59726	35.54139
24.84168	26.0269	27.2566	28.28442	29.776	31.07868	32.26718	33.23668	34.60187	35.49414
24.85266	26.01861	27.24807	28.28475	29.77582	31.06271	32.26768	33.24963	34.58715	35.49471
24.86002	26.01413	27.26747	28.27461	29.76458	31.07729	32.25471	33.25032	34.58004	35.53584
24.83216	26.02635	27.26074	28.2908	29.76916	31.07474	32.25927	33.24456	34.57997	35.54132
24.84301	25.99009	27.22221	28.29658	29.77019	31.07808	32.25714	33.25367	34.59175	35.53356
24.84593	26.02583	27.25482	28.27178	29.75367	31.07546	32.2431	33.25113	34.58462	35.5227

24.85255	26.0179	27.26218	28.28755	29.76452	31.07361	32.23336	33.24419	34.58855	35.53828
24.84079	25.99801	27.25578	28.28862	29.76888	31.06709	32.23434	33.24685	34.57081	35.54909
24.86267	26.03314	27.25258	28.27031	29.75883	31.03706	32.23089	33.24498	34.55548	35.55356
24.85783	26.0005	27.25755	28.28132	29.7691	31.05187	32.2294	33.24355	34.58041	35.53684
24.84273	26.03261	27.23962	28.28251	29.75289	31.06833	32.22175	33.23933	34.5911	35.53961
24.8459	26.02123	27.24631	28.26819	29.75819	31.05822	32.25019	33.21166	34.58699	35.56382
24.84059	26.03044	27.25127	28.2391	29.76329	31.05565	32.25224	33.21689	34.57658	35.56184
24.84789	26.01031	27.23428	28.26953	29.76546	31.06774	32.25176	33.23992	34.57756	35.54373
24.85307	26.02507	27.22238	28.28511	29.76127	31.06278	32.25209	33.23639	34.57907	35.53721
24.85639	26.01653	27.243	28.28624	29.77257	31.06206	32.25258	33.24638	34.57898	35.55427
24.81646	26.03136	27.24963	28.28509	29.75411	31.04284	32.25399	33.21409	34.55603	35.56513
24.86035	26.02847	27.22332	28.28614	29.77846	31.06193	32.25639	33.24797	34.57003	35.56885
24.8598	26.03513	27.24535	28.24764	29.78268	31.06295	32.24345	33.23842	34.58002	35.57349
24.85635	26.03636	27.24533	28.28192	29.78416	31.06088	32.24428	33.24507	34.55254	35.56707
24.84984	26.03469	27.25359	28.28737	29.7841	31.05666	32.23894	33.2492	34.55442	35.58193
24.83076	26.02935	27.24974	28.29426	29.77693	31.053	32.23144	33.24623	34.55569	35.56682
24.86636	26.02049	27.21919	28.29009	29.77459	31.05016	32.24099	33.25459	34.56923	35.57896
24.8688	26.02912	27.2481	28.29058	29.79304	31.05464	32.24113	33.25629	34.56771	35.58652
24.87414	26.00539	27.23545	28.28379	29.79189	31.04421	32.24076	33.24286	34.54192	35.58514
24.87548	26.03578	27.23772	28.28277	29.77004	31.03388	32.23531	33.22631	34.57795	35.58404
24.87575	26.01797	27.24105	28.2737	29.78021	31.05144	32.23917	33.25213	34.57649	35.54982
24.85513	26.0177	27.24272	28.28906	29.77027	31.04486	32.2294	33.23533	34.57404	35.48866
24.87407	26.02119	27.22754	28.29062	29.77614	31.01188	32.23167	33.23939	34.56078	35.58598
24.87329	26.02974	27.22964	28.28113	29.73999	31.06243	32.23292	33.25417	34.55198	35.57797
24.86743	26.01412	27.22725	28.28387	29.77519	31.05378	32.2497	33.2403	34.56297	35.59109
24.85741	26.03485	27.24046	28.26357	29.77309	31.03789	32.23473	33.25187	34.55304	35.57856
24.86824	26.01464	27.23843	28.28255	29.74794	31.06749	32.23273	33.24877	34.55206	35.57076
24.86638	26.03186	27.24998	28.28429	29.77228	31.06594	32.23972	33.23378	34.53383	35.58469
24.86489	26.00957	27.25156	28.28477	29.75587	31.03588	32.26464	33.25211	34.55043	35.58825
24.85309	26.02453	27.23396	28.26273	29.76166	31.06493	32.26277	33.2462	34.55082	35.59272
24.85422	26.04032	27.24772	28.27359	29.75239	31.06626	32.24427	33.23382	34.53827	35.57194
24.86628	26.04204	27.23661	28.26507	29.76461	31.05137	32.24433	33.23126	34.51357	35.57273
24.87166	26.04025	27.24903	28.2787	29.76464	31.0473	32.24043	33.25089	34.54609	35.58853
24.87432	26.01625	27.23609	28.27657	29.74096	31.06273	32.22535	33.26055	34.54019	35.54009
24.87476	26.02737	27.25476	28.28014	29.75638	31.05988	32.22961	33.2631	34.53988	35.57303
24.87427	26.038	27.25444	28.27424	29.75851	31.06194	32.22779	33.26398	34.4739	35.59083
24.88522	26.01584	27.24171	28.28274	29.76774	31.03463	32.22523	33.25114	34.53812	35.58497
24.88561	26.0084	27.25045	28.27894	29.77043	31.05706	32.21759	33.25262	34.53912	35.58789
24.84133	26.03681	27.21719	28.25208	29.75263	31.05442	32.21714	33.24373	34.52881	35.58826
24.88318	26.02621	27.24339	28.2621	29.76867	31.04531	32.21608	33.25686	34.49845	35.59269
24.88436	26.03001	27.2334	28.27446	29.76896	31.05621	32.20647	33.26394	34.52838	35.56683
24.8912	26.03876	27.24237	28.27846	29.77117	31.03667	32.23841	33.26352	34.52129	35.59612
24.88865	26.03849	27.22921	28.27183	29.75563	31.03549	32.23943	33.23441	34.51914	35.59972

24.88944	26.01024	27.23964	28.26698	29.76956	31.05479	32.23927	33.25732	34.50417	35.59525
24.89046	26.04744	27.23861	28.27842	29.77324	31.02528	32.23904	33.26707	34.51807	35.6034
24.89628	26.04185	27.18499	28.26254	29.74316	31.05431	32.23961	33.26727	34.51309	35.60948
24.89848	26.00387	27.22968	28.28318	29.77119	31.05444	32.22831	33.25905	34.51069	35.60719
24.89652	26.04753	27.21033	28.28312	29.75959	31.04708	32.23014	33.27278	34.51828	35.58499
24.89564	26.04142	27.23307	28.236	29.76091	31.04078	32.23114	33.27337	34.51778	35.62332
24.89523	26.03097	27.2434	28.26827	29.74338	31.03964	32.23114	33.2667	34.50899	35.62229
24.88774	26.05689	27.23354	28.23859	29.76096	31.0418	32.22752	33.25866	34.50612	35.61263
24.89169	26.05686	27.22729	28.27938	29.75275	31.04792	32.24308	33.27398	34.48858	35.62905
24.88888	26.05493	27.22098	28.28346	29.75591	31.04722	32.24114	33.27549	34.49379	35.63424
24.88415	26.04866	27.2481	28.25671	29.73906	31.05293	32.23971	33.21948	34.50931	35.64501
24.89046	26.06415	27.2375	28.27223	29.75169	31.03407	32.22211	33.27936	34.51053	35.65381
24.90037	26.06696	27.24573	28.28312	29.7459	31.05451	32.22124	33.28088	34.50565	35.6462
24.90552	26.05084	27.25731	28.28457	29.7399	31.01687	32.22065	33.20682	34.46974	35.65709
24.90562	26.06701	27.24581	28.27388	29.75207	31.05006	32.23605	33.25177	34.51184	35.65998
24.89173	26.05479	27.25407	28.24753	29.74419	31.05574	32.23491	33.27812	34.51166	35.66046
24.89858	26.03771	27.25521	28.28832	29.73622	31.04582	32.23272	33.27626	34.50866	35.6752
24.89463	26.05826	27.25159	28.26235	29.75712	31.05594	32.21917	33.26624	34.50618	35.66864
24.91172	26.06036	27.2179	28.26431	29.74678	31.05429	32.22524	33.2715	34.50758	35.6425
24.89764	26.05221	27.22963	28.2935	29.7538	31.05371	32.22268	33.26972	34.51108	35.67077
24.89767	26.06085	27.24786	28.28496	29.75112	31.04663	32.21647	33.25475	34.50952	35.68662
24.90718	26.03031	27.23977	28.29253	29.75812	31.05504	32.21367	33.2585	34.51258	35.68454
24.90135	26.00489	27.24182	28.2808	29.75706	31.05275	32.21003	33.25312	34.48307	35.67594
24.86816	26.06562	27.23491	28.29498	29.74883	31.05377	32.20136	33.23368	34.49656	35.69165
24.91795	26.04911	27.23281	28.29867	29.75506	31.03963	32.19873	33.2626	34.50353	35.69167
24.91478	26.04985	27.23805	28.29623	29.74936	31.03936	32.1973	33.26143	34.49883	35.67006
24.90487	26.06064	27.18219	28.28346	29.73669	31.06455	32.16036	33.25366	34.45652	35.68452
24.90644	26.06428	27.22205	28.29102	29.72206	31.0673	32.19004	33.25655	34.49401	35.67484
24.90703	26.05556	27.23809	28.27829	29.71665	31.06792	32.19202	33.23028	34.49489	35.69318
24.89393	26.05064	27.23081	28.28806	29.73149	31.06753	32.19173	33.25217	34.49935	35.70835
24.9234	26.06149	27.23681	28.29693	29.73239	31.06148	32.18539	33.25601	34.49937	35.69497
24.92537	26.06894	27.24035	28.25053	29.72572	31.06709	32.18877	33.253	34.48194	35.71417
24.92769	26.07006	27.23825	28.27744	29.70613	31.05776	32.19092	33.25802	34.48349	35.71398
24.93054	26.07388	27.23003	28.29343	29.71983	31.05126	32.1944	33.26067	34.49378	35.72211
24.88579	26.0682	27.23152	28.28059	29.71241	31.06301	32.20322	33.26538	34.46751	35.72354
24.93305	26.03085	27.21747	28.28784	29.71962	31.01861	32.21268	33.25694	34.49978	35.72455
24.93128	26.06726	27.24085	28.28133	29.7194	31.06483	32.21374	33.25208	34.49072	35.72113
24.91419	26.06877	27.23214	28.26634	29.68708	31.06441	32.2157	33.25888	34.48706	35.69134
24.93889	26.07481	27.23434	28.2744	29.70492	31.03925	32.21843	33.25464	34.49031	35.74331
24.94389	26.06023	27.24422	28.28683	29.71822	31.05812	32.22123	33.27264	34.48527	35.73952
24.94754	26.06035	27.24354	28.28818	29.7154	31.06818	32.22138	33.26212	34.47693	35.74149
24.94889	26.07562	27.22999	28.28505	29.72425	31.06758	32.21966	33.27004	34.46844	35.74963
24.95644	26.06669	27.24184	28.29265	29.69972	31.05827	32.21949	33.24503	34.49168	35.73681

24.941	26.07033	27.24312	28.29476	29.72513	31.06291	32.20851	33.27056	34.48991	35.74341
24.94119	26.07388	27.24969	28.28221	29.70767	31.06364	32.21354	33.2704	34.48307	35.69267
24.95051	26.07067	27.24337	28.29252	29.72148	31.04507	32.20501	33.26373	34.49205	35.73082
24.92968	26.06698	27.22428	28.30128	29.73463	31.066	32.19679	33.25591	34.47269	35.75557
24.95928	26.06138	27.20432	28.30483	29.73168	31.0465	32.20076	33.21073	34.47561	35.76526
24.95281	26.06503	27.22787	28.29717	29.72187	31.04307	32.19925	33.24588	34.49574	35.75834
24.96586	26.06621	27.24538	28.30365	29.7308	31.05694	32.20389	33.25507	34.49858	35.77422
24.96551	26.07225	27.23947	28.29578	29.71837	31.06129	32.20311	33.24248	34.47523	35.76298
24.96651	26.07159	27.23464	28.30562	29.73527	31.06446	32.19653	33.25251	34.50102	35.77818
24.96385	26.06662	27.24122	28.30619	29.73331	31.05442	32.1977	33.26901	34.48435	35.75299
24.97088	26.07151	27.2439	28.30557	29.72698	31.05931	32.1991	33.26978	34.504	35.77544
24.97002	26.07447	27.22442	28.31076	29.69744	31.07041	32.19511	33.24537	34.50029	35.78725
24.94782	26.05335	27.24928	28.31133	29.71707	31.04756	32.19533	33.26649	34.48463	35.78629
24.96518	26.06807	27.22654	28.31229	29.71265	31.02488	32.17819	33.25417	34.48994	35.77762
24.97222	26.05459	27.24444	28.31068	29.70711	31.06398	32.17734	33.25499	34.481	35.79041
24.97163	26.07147	27.239	28.30922	29.70917	31.06309	32.17908	33.25836	34.49276	35.75941
24.97332	26.06944	27.24561	28.32486	29.70878	31.061	32.17496	33.25592	34.46156	35.79985
24.97982	26.02191	27.25099	28.32434	29.70543	31.0622	32.17293	33.25444	34.49549	35.78661
24.97252	26.05008	27.23391	28.26478	29.71605	31.0667	32.17513	33.25279	34.49312	35.80519
24.97628	26.06359	27.2334	28.32345	29.68814	31.06656	32.1771	33.24654	34.49223	35.81382
24.97306	26.01952	27.25433	28.32985	29.71272	31.05192	32.18588	33.24532	34.49065	35.78578
24.98554	26.06378	27.2492	28.30758	29.70972	31.06213	32.18608	33.24513	34.49314	35.81418
24.98996	26.06083	27.25132	28.32241	29.72083	31.05569	32.18242	33.242	34.48604	35.80281
24.98409	26.05846	27.25923	28.32726	29.68748	31.06899	32.19772	33.24946	34.49369	35.83005
24.98345	26.04353	27.25549	28.32332	29.72907	31.06843	32.19976	33.24894	34.4797	35.8249
24.98339	26.05879	27.19691	28.33122	29.73741	31.04475	32.19705	33.2414	34.48212	35.81843
24.98322	26.04744	27.24205	28.34022	29.71211	31.04993	32.19699	33.24009	34.48475	35.83508
24.98445	26.05642	27.25109	28.33489	29.74002	31.06984	32.20839	33.23748	34.49114	35.82956
24.98422	26.06166	27.2575	28.3142	29.74492	31.07433	32.20689	33.22965	34.49442	35.83553
24.9866	26.0486	27.24398	28.34128	29.73009	31.06672	32.20781	33.23516	34.49556	35.83664
24.98772	26.0531	27.25207	28.34543	29.73728	31.05888	32.18237	33.24077	34.49458	35.84323
25.00181	26.05155	27.25969	28.3457	29.74306	31.07885	32.18556	33.2318	34.48516	35.82468
25.00154	26.05396	27.24159	28.35007	29.75182	31.06621	32.19861	33.24345	34.49547	35.82723
25.00383	25.99711	27.25288	28.32767	29.7503	31.07665	32.20858	33.22826	34.48702	35.84015
25.00156	26.04033	27.25999	28.34287	29.75111	31.07444	32.20948	33.23584	34.46931	35.85218
24.96545	26.03795	27.26407	28.3535	29.73602	31.06571	32.20663	33.24318	34.45858	35.84303
24.9941	26.0526	27.24322	28.34125	29.71401	31.07017	32.21117	33.24679	34.49971	35.84896
24.99519	26.02396	27.26081	28.34892	29.73543	31.07836	32.21135	33.24596	34.4969	35.83721
25.00485	26.04937	27.26479	28.35874	29.72179	31.07087	32.2009	33.24085	34.50101	35.83844
25.00749	26.02673	27.25712	28.35225	29.71805	31.07074	32.20112	33.2297	34.4944	35.84654
25.00388	26.04556	27.25934	28.33656	29.74527	31.07263	32.20457	33.21283	34.48479	35.82566
25.00823	26.03804	27.25673	28.34949	29.74577	31.06749	32.20638	33.18936	34.50169	35.85979
25.013	26.03862	27.25139	28.35425	29.73076	31.03899	32.20672	33.24782	34.49648	35.86093

25.01126	26.0331	27.25866	28.35344	29.73965	31.0783	32.20571	33.24383	34.49584	35.86598
25.01235	26.03745	27.24604	28.34607	29.74294	31.07999	32.20521	33.24509	34.50985	35.86111
25.00725	26.02719	27.23077	28.35411	29.74331	31.07731	32.21246	33.24345	34.50269	35.85692
24.98501	26.03451	27.22649	28.35836	29.71908	31.08162	32.21187	33.24198	34.5124	35.85282
25.00344	26.03699	27.25743	28.35181	29.71556	31.07106	32.21179	33.24242	34.47511	35.8585
25.00198	26.02542	27.24697	28.35314	29.74703	31.083	32.21237	33.24039	34.5006	35.86342
25.00667	26.02507	27.24608	28.33956	29.74229	31.06835	32.21304	33.18941	34.48355	35.87464
25.00225	26.02239	27.26094	28.35754	29.74684	31.05579	32.21649	33.17634	34.49345	35.81827
25.00182	26.0352	27.24717	28.35965	29.74502	31.08045	32.21871	33.24209	34.51663	35.87511
24.99788	26.0314	27.23122	28.35964	29.74065	31.07306	32.21947	33.24069	34.52043	35.87726
24.99914	26.03116	27.26079	28.36069	29.76068	31.08481	32.22174	33.24359	34.5018	35.87014
24.99972	26.01692	27.26222	28.3186	29.75618	31.07379	32.21861	33.24007	34.47369	35.87169
24.98846	26.02838	27.24248	28.35384	29.75947	31.08977	32.22137	33.2316	34.51858	35.87345
24.96789	26.01844	27.25793	28.34126	29.75329	31.03461	32.22344	33.23097	34.51624	35.84625
24.95995	26.0337	27.25207	28.36017	29.74902	31.08402	32.20738	33.22662	34.49462	35.87512
24.9901	26.03358	27.23384	28.32157	29.75133	31.07753	32.20714	33.22149	34.50315	35.85914
24.99411	26.03094	27.25618	28.35223	29.76	31.05415	32.22818	33.226	34.49649	35.86011
24.98723	26.03132	27.25198	28.31507	29.75748	31.07935	32.22252	33.22879	34.51159	35.87125
24.96174	26.03295	27.25423	28.3511	29.75717	31.08666	32.22118	33.24109	34.51467	35.86354
24.92047	26.03096	27.25886	28.34863	29.75683	31.08725	32.2228	33.22829	34.53013	35.86147
24.96802	25.98397	27.23936	28.35748	29.73831	31.08836	32.21051	33.23927	34.52083	35.86759
24.98759	26.03118	27.24896	28.35329	29.75001	31.0856	32.2137	33.23869	34.51825	35.86817
24.94374	26.02452	27.25449	28.35392	29.74465	31.08104	32.1811	33.23567	34.53478	35.8267
24.97791	26.02963	27.25664	28.34949	29.74428	31.07706	32.21316	33.23773	34.52663	35.86165
24.98762	26.00147	27.24642	28.35057	29.74477	31.0813	32.2129	33.23377	34.5434	35.86327
24.98476	26.00335	27.25089	28.33936	29.74484	31.07504	32.22203	33.20967	34.54114	35.81062
24.97871	26.02069	27.24541	28.35158	29.74145	31.08829	32.22158	33.22162	34.55067	35.85635
24.98083	26.0253	27.23114	28.33951	29.72514	31.05861	32.22955	33.22328	34.54618	35.85171
24.97881	26.02258	27.24788	28.34695	29.73841	31.09644	32.23036	33.22282	34.54374	35.86386
24.97272	26.0258	27.24926	28.33801	29.72463	31.09786	32.21228	33.20127	34.52862	35.86921
24.95562	26.0086	27.23171	28.32795	29.73694	31.09368	32.2135	33.21356	34.553	35.84586
24.95385	26.02582	27.24636	28.33664	29.7383	31.09476	32.20924	33.22794	34.53133	35.85924
24.96626	26.02348	27.1933	28.33574	29.72505	31.09827	32.21071	33.2098	34.53453	35.86772
24.94719	26.02234	27.23201	28.33759	29.74124	31.09842	32.2064	33.21598	34.54628	35.86383
24.93472	26.02996	27.24064	28.34829	29.74422	31.09737	32.20594	33.19598	34.54978	35.85475
24.94662	26.01646	27.21649	28.33629	29.74439	31.08783	32.20824	33.19925	34.5489	35.83768
24.94679	26.02699	27.2351	28.33737	29.74423	31.09146	32.20788	33.22837	34.54835	35.85972
24.9414	26.00994	27.23979	28.31457	29.73818	31.08449	32.22586	33.23259	34.53697	35.76638
24.92658	26.02726	27.22823	28.34356	29.74996	31.09282	32.22759	33.23165	34.54407	35.84604
24.94958	26.02803	27.23379	28.33597	29.725	31.09393	32.22621	33.22706	34.54965	35.85603
24.9497	26.01085	27.22004	28.33111	29.73692	31.09439	32.22561	33.2134	34.55174	35.84721
24.94712	26.03122	27.21982	28.34717	29.7407	31.08687	32.24281	33.23566	34.53907	35.83158
24.94447	26.01709	27.23149	28.3445	29.75021	31.08479	32.24197	33.23142	34.52984	35.84936

24.93136	26.02847	27.20689	28.33642	29.7309	31.0868	32.24252	33.23297	34.54729	35.83912
24.93818	25.99204	27.23469	28.34483	29.71384	31.07435	32.24286	33.2392	34.55213	35.78823
24.93551	26.02136	27.23158	28.33852	29.74102	31.04841	32.24386	33.23354	34.55734	35.80871
24.93284	26.02135	27.21009	28.34538	29.75025	31.08068	32.2457	33.24132	34.55669	35.82121
24.93079	26.01067	27.22482	28.33703	29.74463	31.05093	32.23902	33.24479	34.5465	35.83018
24.92549	26.01694	27.20611	28.33834	29.74517	31.07137	32.24007	33.24309	34.55686	35.82855
24.92376	26.02109	27.23507	28.34079	29.74594	31.09364	32.24115	33.23974	34.55348	35.82863
24.92041	26.02204	27.21727	28.33501	29.72607	31.08898	32.22954	33.22382	34.56889	35.81499
24.91843	26.02057	27.22335	28.34324	29.71986	31.08593	32.23381	33.2414	34.55813	35.82994
24.90678	26.00673	27.19658	28.33437	29.73609	31.09356	32.23692	33.24042	34.53443	35.82378
24.90317	26.00999	27.21037	28.30932	29.73684	31.09472	32.23988	33.23365	34.58083	35.81864
24.8949	26.01013	27.22339	28.3348	29.74297	31.07494	32.25621	33.23372	34.58494	35.81011
24.91031	26.02052	27.19618	28.34146	29.73396	31.09878	32.25705	33.2249	34.5899	35.81164
24.90856	26.02271	27.20653	28.31804	29.66707	31.07977	32.26338	33.22301	34.5846	35.82571
24.90096	26.00311	27.21096	28.33677	29.71899	31.09543	32.26455	33.22833	34.59204	35.82052
24.89776	26.01853	27.21975	28.32564	29.74292	31.08375	32.26448	33.23151	34.57411	35.812
24.8987	26.0343	27.21488	28.3328	29.75309	31.09543	32.26365	33.24108	34.59957	35.82102
24.89605	26.01304	27.19247	28.33476	29.75022	31.06355	32.26464	33.21701	34.59319	35.814
24.89306	26.01952	27.17751	28.29723	29.75271	31.07658	32.26548	33.22167	34.57708	35.81429
24.86521	26.02881	27.21173	28.32436	29.75614	31.09262	32.26582	33.2234	34.58077	35.80969
24.88829	26.03147	27.21739	28.33219	29.75629	31.09278	32.26469	33.2198	34.5896	35.81734
24.88586	26.02459	27.20522	28.30906	29.73114	31.08488	32.26478	33.23039	34.59686	35.79982
24.85728	26.03349	27.19916	28.32121	29.75559	31.09397	32.26298	33.20999	34.60007	35.80607
24.86602	26.0397	27.19358	28.33494	29.72059	31.0944	32.25924	33.2373	34.57772	35.75092
24.86357	26.02863	27.18421	28.33113	29.73206	31.08903	32.25975	33.23271	34.59294	35.80813
24.85271	26.02068	27.22082	28.32566	29.74817	31.09064	32.25936	33.2256	34.57914	35.81262
24.87492	25.99461	27.19603	28.33429	29.74437	31.09786	32.25487	33.23543	34.57914	35.7996
24.86404	26.02978	27.20435	28.29909	29.73107	31.09294	32.25373	33.22563	34.59408	35.81529
24.86632	26.03296	27.21917	28.32359	29.69269	31.07684	32.24911	33.2384	34.57456	35.79255
24.86635	26.0249	27.22736	28.32448	29.73494	31.06845	32.24772	33.24533	34.582	35.79561
24.85504	26.00979	27.21136	28.32324	29.73189	31.08898	32.24638	33.24709	34.56454	35.79408
24.84654	26.0299	27.20082	28.28571	29.71294	31.08708	32.24754	33.24159	34.58317	35.79871
24.84984	26.02072	27.22517	28.31411	29.72876	31.07386	32.25157	33.25158	34.59082	35.79172
24.84623	26.03439	27.22343	28.32615	29.71381	31.06562	32.25358	33.24768	34.58324	35.75598
24.84934	26.02875	27.22329	28.32576	29.72461	31.07329	32.26358	33.25051	34.59159	35.74949
24.84556	26.03986	27.22952	28.30523	29.711	31.0811	32.26479	33.25374	34.58713	35.77124
24.79894	26.03762	27.20434	28.32521	29.72405	31.07709	32.2643	33.25624	34.57849	35.75919
24.79905	26.03784	27.20732	28.29138	29.69766	31.08238	32.26488	33.25436	34.60339	35.7689
24.83472	26.03752	27.22112	28.27914	29.73717	31.07666	32.26864	33.26457	34.59902	35.7685
24.82492	26.01587	27.2146	28.31218	29.73917	31.0739	32.27308	33.24949	34.58854	35.76027
24.83125	26.05247	27.22708	28.31761	29.73571	31.08824	32.27276	33.23873	34.58696	35.76053
24.82268	26.04266	27.19626	28.28732	29.70864	31.08512	32.27252	33.24777	34.58846	35.72984
24.78246	26.03703	27.22433	28.32015	29.74432	31.07923	32.27119	33.25658	34.60626	35.75945

24.79829	26.00341	27.22055	28.33233	29.72401	31.08431	32.26329	33.25359	34.60946	35.75807
24.80897	26.05326	27.22269	28.31058	29.73926	31.07644	32.26337	33.25146	34.60233	35.75972
24.81678	26.03934	27.20827	28.31591	29.75018	31.08136	32.26322	33.25772	34.60187	35.75336
24.8059	26.05586	27.21837	28.33056	29.74406	31.08592	32.19319	33.25554	34.60358	35.74155
24.80765	26.05065	27.20949	28.32924	29.73729	31.08781	32.2518	33.25717	34.61081	35.7592
24.79636	26.03854	27.20381	28.32067	29.75165	31.07209	32.23849	33.25846	34.60793	35.74671
24.79498	26.04567	27.20663	28.31751	29.73943	31.07432	32.24713	33.25735	34.59832	35.75712
24.79453	26.05754	27.22659	28.33203	29.74643	31.07165	32.23536	33.25636	34.57875	35.75616
24.79119	26.05442	27.22408	28.3106	29.7405	31.07557	32.24731	33.23654	34.59375	35.73184
24.79312	26.04792	27.23181	28.32948	29.74549	31.07523	32.25907	33.24869	34.60059	35.74442
24.79375	26.02716	27.22773	28.33208	29.73085	31.07502	32.26744	33.2538	34.60313	35.74774
24.76291	26.05063	27.23749	28.29364	29.7349	31.07627	32.26747	33.2571	34.60228	35.7321
24.78336	26.05126	27.23787	28.32703	29.70243	31.06257	32.26685	33.24809	34.59668	35.7368
24.77982	26.05123	27.23418	28.31758	29.73559	31.05383	32.2684	33.2527	34.58819	35.72819
24.77886	26.04988	27.23164	28.32313	29.70724	31.06015	32.26843	33.25704	34.58119	35.71583
24.75623	26.05076	27.24013	28.2974	29.72062	31.06261	32.25949	33.26527	34.59051	35.70324
24.7547	26.00752	27.2417	28.33268	29.72858	31.05977	32.25496	33.26613	34.59435	35.7159
24.77465	26.04848	27.23414	28.33364	29.71531	31.06499	32.24609	33.27128	34.59326	35.70983
24.74992	26.04033	27.24065	28.32274	29.72963	31.06274	32.26305	33.25842	34.56688	35.6921
24.77406	26.03898	27.20682	28.31855	29.73041	31.04488	32.26135	33.277	34.57683	35.70133
24.75441	26.05484	27.23553	28.32272	29.73325	31.05418	32.25925	33.27496	34.57784	35.66664
24.76208	26.05525	27.21316	28.33376	29.73385	31.02618	32.23681	33.25843	34.58504	35.68904
24.76211	26.05416	27.2365	28.32243	29.73033	31.03496	32.25408	33.27425	34.55594	35.68121
24.76858	26.03499	27.22354	28.32679	29.73502	31.06186	32.24731	33.27831	34.56208	35.69578
24.76382	26.05658	27.235	28.33138	29.7354	31.05836	32.24625	33.2803	34.56984	35.67739
24.75981	26.06259	27.22239	28.32491	29.71908	31.05542	32.24316	33.27798	34.57241	35.68038
24.76219	25.99743	27.23789	28.32665	29.73373	31.04168	32.24313	33.28664	34.58642	35.6718
24.76668	26.02413	27.23035	28.32099	29.7269	31.04613	32.25057	33.29041	34.57725	35.67332
24.76859	26.05913	27.23966	28.32984	29.74163	31.05461	32.25043	33.28909	34.56454	35.65533
24.75103	26.03686	27.2258	28.33603	29.74011	31.05277	32.2495	33.27715	34.57472	35.6668
24.75434	26.04327	27.22661	28.32828	29.74321	31.04457	32.24141	33.27396	34.54869	35.67259
24.74991	26.05438	27.20773	28.32451	29.74481	31.03849	32.24103	33.29216	34.55288	35.67049
24.72574	26.05225	27.23546	28.33696	29.7343	31.04348	32.23844	33.28377	34.51323	35.66417
24.73719	26.05879	27.20559	28.33719	29.70837	31.04385	32.23074	33.28007	34.57056	35.66418
24.74353	26.06847	27.24482	28.3294	29.71662	31.04124	32.22842	33.28212	34.57209	35.66683
24.73955	25.98651	27.25354	28.3377	29.73778	31.02815	32.22902	33.25473	34.57958	35.64831
24.73933	26.06823	27.25139	28.32282	29.73983	31.02894	32.22798	33.28372	34.57083	35.64752
24.74553	26.04984	27.25434	28.33491	29.71783	31.02112	32.2248	33.28115	34.57988	35.66168
24.74394	26.06143	27.25096	28.33225	29.72974	31.02667	32.22536	33.25936	34.54329	35.66682
24.74378	26.06597	27.25811	28.33652	29.73768	31.04337	32.22477	33.25649	34.57653	35.66382
24.71324	26.05402	27.25163	28.31271	29.72256	30.98961	32.23109	33.27367	34.57104	35.65596
24.74606	26.05376	27.26353	28.32402	29.73426	31.0355	32.23174	33.26145	34.56775	35.65658
24.74725	26.07268	27.257	28.32312	29.73616	31.03973	32.23179	33.27212	34.56837	35.65711

24.74564	26.07294	27.26219	28.31206	29.73893	31.0215	32.23307	33.22629	34.54365	35.63636
24.74656	26.07133	27.26813	28.31016	29.71358	31.02029	32.23308	33.23611	34.54714	35.65642
24.71576	26.06171	27.25909	28.3128	29.73197	31.01766	32.23377	33.24006	34.54496	35.65821
24.7104	26.04667	27.26632	28.30544	29.73513	31.03283	32.23288	33.28459	34.56521	35.59693
24.73412	26.06881	27.24764	28.29854	29.74442	31.02954	32.23426	33.27686	34.53719	35.64936
24.73757	26.06737	27.26033	28.30548	29.7191	31.03965	32.23496	33.27957	34.55898	35.63546
24.73618	26.04277	27.25339	28.25724	29.73309	31.02364	32.23504	33.26986	34.55405	35.64396
24.73202	26.06599	27.25723	28.29113	29.73701	31.03944	32.24149	33.28434	34.56201	35.63911
24.70828	26.0701	27.25908	28.30936	29.74398	31.04336	32.2411	33.29245	34.53214	35.64054
24.70749	26.04763	27.24388	28.30602	29.7516	31.00004	32.23869	33.29288	34.54358	35.6344
24.72581	26.03846	27.25337	28.2911	29.75058	31.02863	32.23046	33.29407	34.5362	35.62731
24.73553	26.06249	27.25051	28.29968	29.7391	31.03869	32.22518	33.27149	34.54135	35.62962
24.73497	26.05663	27.21935	28.30437	29.76051	31.04113	32.22824	33.27282	34.54753	35.61279
24.71429	26.04463	27.26605	28.30297	29.74188	31.04424	32.22592	33.24669	34.55342	35.62189
24.71526	26.0271	27.25769	28.30645	29.75179	31.04488	32.22374	33.29156	34.52197	35.62103
24.71694	26.0475	27.25258	28.31092	29.74261	31.04449	32.21483	33.27948	34.54713	35.62108
24.70144	26.05666	27.26517	28.31102	29.75732	31.03585	32.21733	33.29621	34.5473	35.61752
24.73432	26.03838	27.24952	28.29737	29.73498	31.01742	32.21677	33.2873	34.54247	35.60815
24.73574	26.05067	27.23913	28.31332	29.75589	31.036	32.21563	33.30305	34.53688	35.60341
24.73808	26.04078	27.25649	28.29293	29.74928	31.03753	32.21368	33.29994	34.53054	35.60226
24.73224	26.04333	27.24387	28.30022	29.74764	31.03983	32.21184	33.29582	34.5375	35.61861
24.74321	26.05333	27.25216	28.3131	29.74948	31.03858	32.21008	33.26722	34.49881	35.61707
24.74426	26.04136	27.27073	28.30293	29.72213	31.03892	32.20752	33.26475	34.5205	35.61945
24.75231	26.04459	27.26623	28.3077	29.74904	31.02287	32.20443	33.28576	34.52627	35.60825
24.72835	26.03592	27.26866	28.292	29.75041	31.04339	32.20264	33.28508	34.54054	35.6175
24.75851	26.07425	27.24658	28.29709	29.74231	31.04553	32.20374	33.29296	34.51018	35.62497
24.7581	26.07158	27.28173	28.29623	29.74715	31.04817	32.19751	33.28231	34.53417	35.6247
24.745	26.07308	27.27612	28.28846	29.74268	31.02802	32.19673	33.28592	34.53538	35.61499
24.74519	26.06976	27.2693	28.30202	29.7508	31.04858	32.19567	33.29753	34.53752	35.61954
24.74034	26.06501	27.21245	28.30051	29.74065	31.03277	32.19252	33.30199	34.53535	35.62636
24.7513	26.05683	27.27448	28.29545	29.73075	31.03356	32.19652	33.30051	34.53931	35.62614
24.75577	26.06732	27.28187	28.25458	29.74808	31.03573	32.19509	33.29501	34.53964	35.63164
24.75377	26.05396	27.26761	28.30366	29.71233	31.05032	32.19815	33.25943	34.53762	35.60447
24.74976	26.05409	27.24959	28.2924	29.75618	31.04675	32.18467	33.26499	34.53572	35.6316
24.75029	26.06253	27.25693	28.27563	29.75692	31.04546	32.18182	33.25845	34.53523	35.63326
24.74963	26.05093	27.26048	28.29464	29.7506	31.05844	32.19441	33.28691	34.52475	35.62328
24.74815	26.03844	27.25884	28.27814	29.7262	31.06039	32.19451	33.26338	34.52981	35.63551
24.7542	26.02745	27.27074	28.29525	29.75128	31.06217	32.19663	33.30401	34.5289	35.59074
24.75648	26.05115	27.2591	28.29646	29.70946	31.04451	32.19884	33.30663	34.49598	35.6001
24.75987	26.04455	27.25717	28.30074	29.71736	31.03764	32.20049	33.30628	34.53094	35.62984
24.74617	26.05208	27.22843	28.3022	29.7441	31.06289	32.20071	33.30571	34.52992	35.61259
24.74978	26.04826	27.26637	28.28666	29.73945	31.06425	32.20356	33.30345	34.52944	35.6296
24.75385	26.04263	27.25076	28.29402	29.73267	31.05544	32.20705	33.30519	34.53373	35.63311

24.75624	26.03632	27.25588	28.29903	29.72934	31.05456	32.2095	33.30475	34.51231	35.6103
24.77127	26.05292	27.26033	28.29147	29.71088	31.06394	32.20912	33.30097	34.50702	35.62428
24.77578	26.05509	27.24339	28.29972	29.7386	31.06131	32.19892	33.29174	34.52095	35.64265
24.77809	26.05555	27.2576	28.28872	29.73315	31.04817	32.19649	33.29055	34.51555	35.63163
24.77925	26.04982	27.26201	28.30056	29.7205	31.04709	32.20607	33.29066	34.52362	35.63535
24.754	26.05518	27.23636	28.30335	29.72082	31.06212	32.20559	33.2895	34.52459	35.63007
24.75519	26.04869	27.22004	28.29678	29.72249	31.05997	32.20442	33.29269	34.51148	35.63166
24.75636	26.04434	27.26061	28.28347	29.73012	31.05914	32.20358	33.29612	34.51455	35.64059
24.77365	26.05137	27.2283	28.2982	29.73431	31.06218	32.20187	33.3	34.52222	35.64521
24.75489	26.05767	27.27007	28.30443	29.7408	31.0561	32.19686	33.29534	34.51942	35.64916
24.77651	26.05596	27.26664	28.31238	29.73112	31.06243	32.1945	33.29108	34.52792	35.6309
24.7707	26.05133	27.25105	28.30575	29.73268	31.06241	32.18028	33.28662	34.51468	35.65006
24.77544	26.04276	27.25702	28.29515	29.71731	31.06208	32.18765	33.2785	34.51328	35.64948

Neutrino Velocity Results

Table 51 shows the x-direction velocity results for the last 1,500 time steps, or 30 seconds of real time, for the ten runs.

Table 51: Neutrino Velocity Results for Spillway Comparison

Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
0.141055	0.608251	1.12235	1.62918	2.04434	2.53537	2.94702	3.39077	3.94952	4.57886
0.14155	0.612458	1.1241	1.63581	2.07453	2.53722	2.95339	3.37462	3.93739	4.5721
0.148622	0.6221	1.12444	1.64205	2.10926	2.5328	2.97546	3.36934	3.93414	4.57565
0.16205	0.628576	1.12762	1.64172	2.14027	2.52314	2.99539	3.38138	3.93949	4.58367
0.174588	0.629132	1.13204	1.63859	2.16029	2.50934	3.00003	3.39591	3.94321	4.59888
0.180011	0.623648	1.13503	1.63351	2.17105	2.50085	2.9912	3.4142	3.94958	4.60773
0.175422	0.614389	1.13703	1.62671	2.17263	2.49734	2.97995	3.4189	3.95103	4.59997
0.164231	0.60713	1.1356	1.62249	2.16845	2.49337	2.97264	3.41082	3.9552	4.59562
0.155556	0.603317	1.13043	1.6215	2.16243	2.48638	2.96827	3.38642	3.94924	4.59287
0.152276	0.605553	1.1271	1.62222	2.15215	2.47813	2.96023	3.35497	3.9441	4.58902
0.151865	0.610921	1.12566	1.62427	2.1329	2.47334	2.95825	3.34265	3.94623	4.58527
0.152306	0.616513	1.12441	1.62237	2.11397	2.47933	2.95111	3.34045	3.95709	4.582
0.153062	0.616399	1.12314	1.6199	2.10531	2.49114	2.9432	3.35363	3.96951	4.58354
0.152383	0.608807	1.12232	1.62403	2.10181	2.50236	2.94252	3.36981	3.96716	4.57856
0.152993	0.598704	1.11981	1.63177	2.09178	2.51362	2.94619	3.38432	3.97207	4.58666
0.158325	0.592588	1.12015	1.64218	2.07734	2.52197	2.95235	3.39338	3.97298	4.57677
0.165879	0.593608	1.13128	1.64744	2.06004	2.5235	2.95741	3.40417	3.9769	4.5722
0.16913	0.603687	1.14791	1.64792	2.04268	2.51686	2.96003	3.40951	3.97742	4.58603
0.168929	0.618807	1.1579	1.64249	2.03073	2.50319	2.9559	3.41872	3.98874	4.60594

0.168801	0.630911	1.16205	1.63482	2.02681	2.48317	2.95409	3.42803	3.99579	4.61802
0.172627	0.630368	1.15238	1.62524	2.03086	2.46799	2.95242	3.43972	3.99884	4.63846
0.183393	0.616413	1.13055	1.61228	2.03651	2.46629	2.951	3.43802	3.99137	4.63522
0.200008	0.59761	1.11122	1.60331	2.04078	2.47086	2.95223	3.43031	3.98503	4.63325
0.214467	0.583377	1.10058	1.59807	2.04138	2.47883	2.94908	3.40873	3.97355	4.61449
0.218929	0.577431	1.09929	1.59684	2.0416	2.4841	2.95355	3.38673	3.96575	4.59634
0.210561	0.578905	1.11142	1.6016	2.0488	2.49065	2.96641	3.36887	3.95717	4.57078
0.194048	0.586899	1.1235	1.61074	2.0612	2.50097	2.98474	3.36802	3.96495	4.56083
0.178431	0.599849	1.13251	1.62817	2.08056	2.5083	3.00148	3.38079	3.97284	4.54367
0.167594	0.614411	1.13684	1.64903	2.10688	2.51312	3.00444	3.39362	3.97881	4.53867
0.161723	0.627647	1.13362	1.66439	2.13446	2.51226	2.99969	3.40439	3.97475	4.52662
0.160237	0.633486	1.12859	1.6686	2.15305	2.50115	2.99535	3.40196	3.96852	4.53054
0.164402	0.635699	1.12717	1.65588	2.1621	2.48855	2.98535	3.38769	3.95687	4.52978
0.17501	0.632305	1.13119	1.63737	2.16428	2.48259	2.97308	3.36344	3.9439	4.53
0.187381	0.620235	1.13778	1.62221	2.16178	2.47837	2.96635	3.34774	3.92824	4.53414
0.199434	0.60058	1.14785	1.61255	2.15897	2.47915	2.96148	3.34831	3.92252	4.54409
0.208633	0.576754	1.1552	1.61197	2.15546	2.48184	2.96483	3.36001	3.91938	4.55545
0.212603	0.556471	1.14794	1.61682	2.14922	2.48845	2.97875	3.38796	3.92936	4.56547
0.212163	0.549883	1.13187	1.6257	2.13844	2.49624	2.99077	3.41455	3.94069	4.56989
0.213915	0.564965	1.11859	1.63317	2.1183	2.50079	2.98989	3.42795	3.95418	4.59563
0.222577	0.591993	1.11369	1.6281	2.09012	2.50067	2.97961	3.42622	3.96332	4.62134
0.233959	0.61871	1.12219	1.61087	2.06166	2.49433	2.9657	3.40579	3.97194	4.62973
0.240356	0.632132	1.13534	1.59034	2.03726	2.4857	2.9483	3.38361	3.97843	4.63384
0.236666	0.628147	1.14486	1.57495	2.02234	2.4779	2.93281	3.37106	3.97804	4.6267
0.223936	0.609439	1.1507	1.57056	2.01412	2.47673	2.92288	3.37245	3.98033	4.62324
0.207636	0.591967	1.15188	1.57389	2.00877	2.48805	2.92222	3.38269	3.98799	4.62144
0.195006	0.58543	1.14724	1.58567	2.00275	2.5054	2.92357	3.39187	3.98027	4.61778
0.187821	0.589436	1.13651	1.59993	1.99811	2.52099	2.92238	3.40479	3.96248	4.6126
0.186228	0.59801	1.12396	1.61187	2.00008	2.52393	2.92345	3.41869	3.955	4.61723
0.189041	0.608667	1.11375	1.62075	2.01078	2.5197	2.93081	3.43597	3.95151	4.62983
0.194867	0.613999	1.11079	1.62978	2.0315	2.51753	2.94167	3.45027	3.94894	4.63098
0.203207	0.617311	1.11177	1.64046	2.05934	2.51958	2.96127	3.45438	3.9476	4.61776
0.214498	0.618205	1.11499	1.65003	2.08669	2.52456	2.98397	3.4473	3.94362	4.60114
0.230176	0.61889	1.1167	1.64766	2.10822	2.52573	3.00013	3.43222	3.93062	4.5763
0.245033	0.616752	1.11911	1.63591	2.11968	2.51452	3.00983	3.41536	3.91392	4.56777
0.253948	0.610773	1.12316	1.61774	2.12519	2.49016	3.0164	3.40668	3.90419	4.54196
0.25631	0.600646	1.12918	1.6006	2.13372	2.46564	3.0108	3.39594	3.91301	4.54044
0.254519	0.586313	1.13243	1.59439	2.14107	2.45174	2.99629	3.38984	3.93994	4.53491
0.252155	0.575323	1.14265	1.59969	2.14347	2.45092	2.9857	3.37863	3.96865	4.53623
0.249723	0.572503	1.15521	1.61201	2.13998	2.46221	2.97293	3.36968	3.99567	4.542
0.244952	0.575519	1.16041	1.61895	2.13253	2.47759	2.96228	3.34939	4.00917	4.54333
0.238921	0.579005	1.16314	1.62355	2.12566	2.49584	2.95508	3.32981	4.01305	4.53812
0.232333	0.586618	1.16278	1.62141	2.11784	2.51163	2.95368	3.32446	4.01874	4.53573

0.226444	0.600998	1.16195	1.61293	2.11506	2.5143	2.95467	3.32909	4.02433	4.53917
0.223312	0.617043	1.15881	1.60825	2.11601	2.50884	2.95482	3.34003	4.01887	4.55158
0.222411	0.628661	1.15003	1.60196	2.11489	2.50553	2.95531	3.35267	4.01535	4.56767
0.221702	0.632082	1.14298	1.59605	2.10557	2.5038	2.95641	3.37981	4.00379	4.59739
0.222448	0.629221	1.13391	1.59336	2.09015	2.50196	2.95478	3.41026	3.99125	4.60018
0.226035	0.622864	1.12478	1.59445	2.07036	2.49316	2.95118	3.43728	3.9818	4.60332
0.232626	0.612484	1.11727	1.60063	2.04876	2.48687	2.94457	3.44808	3.96734	4.59389
0.240989	0.599571	1.11421	1.60653	2.03295	2.49294	2.9448	3.44723	3.95683	4.58763
0.248903	0.587538	1.10832	1.60581	2.02317	2.5096	2.95508	3.43159	3.93805	4.60045
0.25703	0.579083	1.10549	1.60206	2.01346	2.52791	2.96879	3.40823	3.92783	4.61702
0.26533	0.577578	1.10822	1.59755	2.00125	2.53999	2.98056	3.38712	3.91574	4.63084
0.270435	0.585286	1.12233	1.59402	1.99366	2.54407	2.98715	3.38007	3.91176	4.63442
0.272325	0.598851	1.13912	1.59633	1.99578	2.5349	2.98702	3.37853	3.90602	4.63108
0.271523	0.612144	1.1529	1.60892	2.01199	2.51484	2.98204	3.37828	3.89927	4.61652
0.266588	0.618789	1.15945	1.62403	2.04054	2.49704	2.97206	3.37155	3.90271	4.59841
0.256897	0.616357	1.15833	1.64183	2.07683	2.48436	2.96394	3.36887	3.92622	4.57668
0.247535	0.606296	1.15088	1.65355	2.11144	2.48195	2.95767	3.36803	3.95324	4.57105
0.247342	0.594499	1.13936	1.65119	2.13973	2.4888	2.95913	3.36888	3.97629	4.57394
0.25777	0.589518	1.13022	1.63764	2.15785	2.49212	2.96415	3.38401	3.97309	4.57225
0.271227	0.591692	1.12453	1.61393	2.16377	2.48748	2.96785	3.3994	3.95732	4.57232
0.275533	0.596155	1.12417	1.59043	2.16285	2.47496	2.97021	3.41394	3.93591	4.56421
0.267858	0.600803	1.12643	1.58003	2.16404	2.4639	2.96806	3.4261	3.9171	4.56079
0.252192	0.605356	1.13611	1.5806	2.1654	2.46109	2.96091	3.43031	3.9186	4.55216
0.235421	0.612374	1.14594	1.59123	2.16041	2.46303	2.95243	3.42012	3.92817	4.54757
0.227677	0.617352	1.14643	1.60116	2.15144	2.47323	2.94117	3.40853	3.95622	4.55103
0.232969	0.61559	1.14124	1.60439	2.1388	2.49572	2.94312	3.40769	3.98404	4.55316
0.249309	0.608115	1.13142	1.60782	2.1282	2.52584	2.95056	3.40478	4.00335	4.55972
0.2685	0.602648	1.12443	1.60681	2.11726	2.54938	2.95901	3.40829	4.01682	4.57069
0.283578	0.600728	1.1204	1.60458	2.10232	2.55721	2.95702	3.40431	4.01783	4.58256
0.289704	0.601185	1.1187	1.60055	2.08027	2.55242	2.95513	3.40471	4.00908	4.60127
0.287178	0.599117	1.11958	1.59339	2.05494	2.53793	2.95739	3.39722	3.99463	4.6215
0.279859	0.596882	1.12338	1.58281	2.03377	2.51488	2.95647	3.38696	3.98884	4.63755
0.273331	0.598392	1.13056	1.57767	2.02083	2.49982	2.95703	3.37345	3.98405	4.64664
0.275992	0.60295	1.14082	1.58491	2.01519	2.49288	2.9594	3.35369	3.98431	4.64481
0.28553	0.609725	1.14757	1.60232	2.01232	2.4903	2.96321	3.3436	3.98085	4.6324
0.294373	0.619131	1.15042	1.62502	2.01043	2.48917	2.97071	3.34642	3.97767	4.62216
0.296128	0.629759	1.14643	1.64176	2.01324	2.48903	2.97844	3.36195	3.97101	4.61443
0.290723	0.635474	1.13364	1.6458	2.01754	2.48897	2.99196	3.37122	3.96938	4.6102
0.27957	0.628573	1.12052	1.64696	2.02571	2.48342	3.00982	3.38658	3.95283	4.59922
0.263688	0.613445	1.11141	1.64686	2.03597	2.47533	3.0158	3.39428	3.93447	4.5921
0.248029	0.595361	1.11205	1.64193	2.05367	2.47272	3.01218	3.39439	3.91612	4.58307
0.237643	0.585752	1.11745	1.63135	2.07889	2.47268	3.00069	3.39073	3.90241	4.57333
0.232891	0.581774	1.12218	1.611	2.10495	2.47652	2.98773	3.38793	3.90248	4.557

0.232786	0.583974	1.12529	1.5914	2.13093	2.48553	2.97145	3.3832	3.91635	4.53622
0.237415	0.589989	1.12482	1.58706	2.15153	2.49584	2.95446	3.37544	3.93559	4.52714
0.246394	0.599336	1.12484	1.58929	2.16493	2.50873	2.93934	3.36588	3.95378	4.52551
0.257767	0.610361	1.12174	1.58668	2.17236	2.5221	2.92372	3.36192	3.97318	4.54288
0.269075	0.620788	1.12173	1.58146	2.17169	2.53984	2.91514	3.37804	3.97477	4.55528
0.27835	0.628132	1.11947	1.58449	2.1651	2.55503	2.91515	3.39353	3.96778	4.5608
0.285352	0.632552	1.11262	1.59002	2.15273	2.56149	2.92241	3.41428	3.96177	4.56168
0.293234	0.638708	1.10854	1.60112	2.13384	2.55574	2.93646	3.42881	3.97195	4.55508
0.30062	0.646815	1.10231	1.61877	2.11401	2.53706	2.95239	3.43952	3.99013	4.54129
0.303569	0.656721	1.09662	1.63458	2.09868	2.51418	2.96514	3.4397	4.01567	4.5451
0.298499	0.660565	1.09278	1.6358	2.08626	2.4968	2.97186	3.44107	4.03335	4.55191
0.291898	0.651116	1.09448	1.62491	2.07731	2.48281	2.96843	3.44002	4.03751	4.5712
0.289146	0.629679	1.10536	1.61169	2.06638	2.47742	2.96428	3.43979	4.02904	4.59889
0.291475	0.596883	1.12368	1.60937	2.05506	2.48454	2.96227	3.44136	4.00934	4.61549
0.294056	0.569915	1.14374	1.61737	2.04355	2.4947	2.97553	3.42722	3.98225	4.6009
0.291884	0.558284	1.15882	1.62944	2.03717	2.49707	2.98476	3.41028	3.95816	4.58653
0.280368	0.563611	1.16538	1.63253	2.03162	2.48927	2.98339	3.38256	3.93214	4.57292
0.261243	0.575274	1.16431	1.62781	2.02508	2.47384	2.98158	3.35988	3.92381	4.57425
0.242612	0.585609	1.15711	1.61995	2.02287	2.45864	2.97906	3.33827	3.92202	4.58071
0.234374	0.593402	1.14398	1.60528	2.02565	2.45439	2.97422	3.3274	3.92704	4.6082
0.240154	0.602095	1.12729	1.59076	2.03284	2.45974	2.96321	3.32422	3.93743	4.62918
0.254344	0.615928	1.11616	1.58218	2.04108	2.47559	2.94982	3.32762	3.95198	4.63546
0.268812	0.633474	1.11375	1.5839	2.05239	2.49205	2.93557	3.337	3.95639	4.62744
0.277232	0.647721	1.11474	1.59205	2.06528	2.50348	2.92426	3.34679	3.95714	4.61467
0.282942	0.652425	1.12336	1.60315	2.08228	2.50542	2.92169	3.35696	3.95683	4.6021
0.289543	0.646939	1.13009	1.6067	2.1058	2.50589	2.92375	3.36479	3.95559	4.60216
0.295243	0.632689	1.12949	1.60626	2.13048	2.51177	2.93623	3.37406	3.95355	4.59036
0.29793	0.616635	1.12408	1.59854	2.15089	2.52162	2.95362	3.38307	3.94548	4.572
0.296228	0.603918	1.11752	1.58392	2.16507	2.53134	2.96765	3.39072	3.94445	4.55417
0.290754	0.590971	1.10973	1.57334	2.17046	2.53651	2.98368	3.40195	3.93467	4.53482
0.281232	0.586026	1.1105	1.57622	2.16704	2.53779	2.99257	3.40443	3.93449	4.50853
0.270826	0.588943	1.11747	1.59043	2.15059	2.53273	2.99826	3.39708	3.95229	4.48738
0.266007	0.594333	1.12415	1.61211	2.13047	2.52663	3.00327	3.39532	3.97425	4.48366
0.267604	0.597001	1.12217	1.62376	2.11225	2.51518	3.00418	3.39249	3.99683	4.51042
0.27102	0.604193	1.11208	1.62614	2.09814	2.50033	2.99245	3.39053	4.02096	4.53122
0.271837	0.613648	1.10765	1.61861	2.08473	2.48807	2.98165	3.40353	4.03927	4.55106
0.270306	0.618969	1.11325	1.61118	2.06615	2.47541	2.95887	3.42288	4.04582	4.5704
0.268821	0.613737	1.12479	1.60961	2.04743	2.46488	2.93982	3.4368	4.04369	4.5821
0.27054	0.603874	1.1373	1.61161	2.03363	2.45222	2.92045	3.44408	4.02539	4.58555
0.276085	0.596273	1.15091	1.61306	2.02843	2.44048	2.91581	3.43651	4.00233	4.59586
0.284896	0.596002	1.16086	1.60982	2.02548	2.43697	2.91967	3.42774	3.97994	4.59769
0.294331	0.606479	1.15888	1.60089	2.02324	2.43929	2.92748	3.4201	3.96358	4.60728
0.298869	0.622863	1.15459	1.59207	2.02371	2.44595	2.93513	3.41825	3.94992	4.61174

0.298395	0.638822	1.14897	1.58551	2.02626	2.45711	2.94637	3.41387	3.94864	4.59854
0.292941	0.644236	1.14327	1.59339	2.02457	2.47436	2.95279	3.40312	3.955	4.57007
0.284526	0.634534	1.13963	1.6069	2.02407	2.48682	2.95987	3.39622	3.94617	4.54988
0.276925	0.61415	1.13311	1.62581	2.03214	2.49397	2.96868	3.38408	3.93109	4.53089
0.275676	0.594343	1.12978	1.63936	2.05269	2.4978	2.97612	3.36632	3.91214	4.52352
0.279819	0.583051	1.13022	1.63287	2.08003	2.50847	2.98136	3.35427	3.90283	4.52275
0.286248	0.580295	1.13484	1.61608	2.10798	2.52357	2.9831	3.35671	3.913	4.53667
0.290831	0.584867	1.13832	1.59473	2.13003	2.53702	2.98314	3.36289	3.93274	4.55367
0.290515	0.593257	1.14255	1.57582	2.14259	2.54492	2.97035	3.37346	3.96322	4.57375
0.28711	0.605432	1.14417	1.56233	2.14609	2.54411	2.94876	3.3788	3.98692	4.5832
0.282875	0.618898	1.14023	1.55874	2.1464	2.53092	2.92949	3.37716	4.00423	4.58443
0.278198	0.632533	1.13326	1.56611	2.14813	2.51866	2.92468	3.36167	4.00698	4.59328
0.275626	0.643542	1.12722	1.58702	2.14856	2.5071	2.92346	3.34379	3.99531	4.58705
0.27747	0.645157	1.12667	1.61069	2.14423	2.49518	2.93171	3.33406	3.97245	4.57711
0.279492	0.63753	1.12541	1.62359	2.13178	2.48903	2.94911	3.34508	3.96483	4.55332
0.278199	0.624082	1.12581	1.62929	2.11112	2.48924	2.9726	3.36613	3.9537	4.53356
0.272587	0.609473	1.12442	1.63088	2.09144	2.48881	2.99246	3.3981	3.95959	4.52783
0.266898	0.598642	1.12414	1.62441	2.07682	2.48501	3.0002	3.42461	3.96418	4.53055
0.264608	0.594387	1.12808	1.6122	2.06263	2.47774	2.99353	3.44034	3.97692	4.53091
0.268733	0.59775	1.13619	1.59935	2.05081	2.47382	2.98252	3.43292	3.98778	4.5489
0.275847	0.604112	1.14736	1.58924	2.03977	2.47346	2.97311	3.41912	3.99566	4.55961
0.28177	0.615399	1.15958	1.5888	2.0274	2.47725	2.961	3.40606	4.00386	4.55591
0.288081	0.627384	1.17307	1.59513	2.01363	2.48018	2.95007	3.4027	3.99796	4.55125
0.294498	0.632472	1.1844	1.60147	2.0109	2.48605	2.94236	3.39484	3.99644	4.54753
0.2986	0.629641	1.18889	1.60735	2.01745	2.49292	2.94408	3.39064	3.98924	4.54679
0.298157	0.617509	1.178	1.61483	2.0278	2.50297	2.94745	3.3797	3.99171	4.55497
0.29537	0.596032	1.1575	1.62232	2.04332	2.50438	2.94956	3.36251	3.99312	4.58064
0.291884	0.578195	1.13718	1.62796	2.06079	2.49621	2.94781	3.34596	3.99968	4.59782
0.286491	0.576972	1.1238	1.63234	2.07851	2.4888	2.94296	3.33811	4.00255	4.60479
0.279699	0.59406	1.1203	1.63915	2.09579	2.48951	2.93645	3.34674	4.00406	4.61865
0.273384	0.620779	1.12062	1.63995	2.10796	2.49057	2.93123	3.35899	3.9991	4.63055
0.266874	0.644842	1.12347	1.63726	2.11791	2.48394	2.93496	3.37698	3.98786	4.64219
0.26077	0.658874	1.12722	1.62438	2.12801	2.47597	2.9384	3.39251	3.97461	4.63613
0.258905	0.656844	1.13383	1.60971	2.13377	2.48054	2.94923	3.40258	3.96245	4.6081
0.26224	0.640414	1.13952	1.6024	2.13778	2.49105	2.9628	3.40821	3.94976	4.58149
0.271364	0.624813	1.15186	1.59273	2.14424	2.4999	2.96872	3.41555	3.93808	4.56462
0.288976	0.622523	1.1676	1.583	2.15513	2.50784	2.97521	3.41874	3.92732	4.54409
0.309645	0.628801	1.17839	1.57287	2.16637	2.51709	2.97962	3.41369	3.92191	4.5175
0.323385	0.634673	1.18633	1.56741	2.17248	2.52326	2.97796	3.40509	3.92256	4.49847
0.320846	0.633686	1.18932	1.56749	2.17037	2.52519	2.98017	3.38564	3.92433	4.49733
0.304062	0.62229	1.18206	1.5767	2.15733	2.51928	2.98194	3.36483	3.93915	4.49281
0.283335	0.603757	1.16468	1.5887	2.13392	2.50846	2.97274	3.35279	3.96279	4.48785
0.265017	0.587081	1.14294	1.60013	2.10143	2.49544	2.95526	3.35344	3.97375	4.48709

0.255051	0.57808	1.12767	1.60659	2.0648	2.48129	2.9414	3.36185	3.98884	4.49276
0.256722	0.578929	1.12494	1.60643	2.03272	2.46746	2.9346	3.37398	3.99192	4.49971
0.26898	0.589198	1.12373	1.6057	2.01063	2.44877	2.93449	3.38699	3.98138	4.5107
0.280501	0.601715	1.11782	1.60766	1.99843	2.43883	2.94169	3.38776	3.96934	4.53539
0.283097	0.614317	1.10919	1.61482	1.997	2.4375	2.96445	3.38212	3.96488	4.5633
0.276797	0.627498	1.10805	1.62459	2.00794	2.44802	2.98392	3.37266	3.97554	4.58911
0.268689	0.631857	1.11579	1.63668	2.02104	2.4687	2.99321	3.37056	3.98492	4.61016
0.264777	0.628909	1.12894	1.64084	2.03326	2.49096	2.98862	3.37239	3.99382	4.62175
0.266815	0.622123	1.1482	1.63794	2.04025	2.5102	2.9721	3.37535	3.99797	4.6215
0.274618	0.613983	1.17043	1.63114	2.04635	2.52099	2.9537	3.38232	4.00044	4.60812
0.286888	0.609006	1.18143	1.62473	2.05592	2.52107	2.94324	3.38175	3.99781	4.58462
0.299781	0.60729	1.17605	1.6174	2.0759	2.51712	2.94218	3.38129	3.99688	4.5693
0.307561	0.608194	1.16229	1.60292	2.1027	2.51862	2.94696	3.38201	4.00012	4.56333
0.305209	0.610105	1.15351	1.58813	2.12766	2.5265	2.95163	3.38752	3.99178	4.55902
0.294453	0.618977	1.15289	1.57734	2.15109	2.53448	2.94974	3.4029	3.99013	4.558
0.284775	0.630127	1.16048	1.58151	2.17425	2.53595	2.94292	3.41021	3.99985	4.56574
0.282307	0.635623	1.17105	1.58986	2.18685	2.53069	2.93511	3.40414	4.00768	4.57128
0.285892	0.636942	1.18246	1.60254	2.19144	2.51608	2.93458	3.38656	4.01658	4.57759
0.289076	0.636462	1.18302	1.6102	2.1911	2.49172	2.92989	3.36801	4.01143	4.58788
0.286282	0.633761	1.16795	1.61287	2.18624	2.46273	2.92383	3.35061	3.99654	4.58118
0.277141	0.626323	1.14629	1.61264	2.17692	2.44202	2.92377	3.33309	3.97459	4.57463
0.265715	0.618175	1.12682	1.61299	2.16251	2.43724	2.92906	3.32742	3.95242	4.57061
0.256628	0.611162	1.11808	1.61781	2.14445	2.44827	2.9376	3.32726	3.92506	4.56704
0.252654	0.603168	1.1157	1.6283	2.11996	2.46867	2.96019	3.33638	3.91819	4.55454
0.256883	0.599002	1.11546	1.63867	2.09675	2.48397	2.99602	3.34196	3.9126	4.55489
0.267503	0.602032	1.11515	1.63984	2.07519	2.49171	3.02462	3.35013	3.9214	4.54712
0.278812	0.61105	1.11613	1.63638	2.05371	2.49984	3.04381	3.36818	3.92292	4.54739
0.286493	0.616773	1.12112	1.632	2.04105	2.50621	3.05439	3.38452	3.93727	4.55088
0.287681	0.614241	1.12505	1.62763	2.03182	2.50953	3.03578	3.40312	3.9436	4.5521
0.282574	0.607267	1.13246	1.62186	2.02179	2.51918	2.99973	3.41513	3.94434	4.55409
0.272875	0.60246	1.14016	1.61619	2.01004	2.53117	2.9589	3.41864	3.95603	4.54442
0.264393	0.602191	1.1431	1.60989	2.00027	2.53359	2.92922	3.42106	3.97543	4.53776
0.261375	0.604689	1.14681	1.60714	1.99915	2.52029	2.91227	3.4137	3.98851	4.53577
0.265276	0.609325	1.15031	1.59736	2.00941	2.49816	2.89969	3.40475	3.99693	4.53814
0.272194	0.620533	1.15274	1.5919	2.02809	2.47576	2.8945	3.39584	4.01058	4.5414
0.276738	0.632531	1.15671	1.59323	2.05186	2.4653	2.89912	3.38199	4.01839	4.54646
0.276021	0.640999	1.15703	1.59967	2.07851	2.46424	2.90918	3.37612	4.01324	4.55129
0.273321	0.640924	1.15163	1.61038	2.1047	2.47495	2.923	3.36962	4.01019	4.55596
0.271957	0.632788	1.1402	1.62393	2.12857	2.49623	2.93739	3.36601	4.00092	4.56027
0.269413	0.619577	1.12719	1.63761	2.14541	2.51769	2.9544	3.36089	3.99565	4.56375
0.265267	0.602861	1.12078	1.64319	2.15416	2.53193	2.97102	3.35609	3.99223	4.57161
0.260606	0.592708	1.12169	1.64275	2.15731	2.53719	2.98069	3.35027	3.99363	4.57292
0.255854	0.59449	1.13032	1.6435	2.16047	2.52865	2.98476	3.34909	3.99046	4.56947

0.252501	0.603828	1.14085	1.64338	2.16595	2.50879	2.99299	3.35327	3.9862	4.56674
0.250952	0.616925	1.14783	1.64405	2.17606	2.48342	2.99365	3.36404	3.97399	4.56502
0.251449	0.626352	1.14467	1.64944	2.19142	2.46578	2.98838	3.37907	3.96756	4.58432
0.254159	0.627216	1.13784	1.65864	2.20668	2.46219	2.98137	3.37969	3.95594	4.60685
0.258371	0.622493	1.12319	1.6672	2.20209	2.4696	2.97202	3.37154	3.94778	4.61659
0.26149	0.612042	1.11221	1.6668	2.17305	2.48479	2.95775	3.35336	3.95366	4.63063
0.265186	0.597951	1.10349	1.64964	2.13048	2.49751	2.94754	3.34692	3.95994	4.61491
0.272078	0.586897	1.10302	1.61933	2.08967	2.50563	2.94585	3.34744	3.96282	4.59463
0.27969	0.585314	1.11101	1.58923	2.06414	2.5072	2.94896	3.35742	3.95569	4.57304
0.280897	0.589012	1.12846	1.57699	2.05058	2.50639	2.9573	3.37564	3.94818	4.561
0.270142	0.591466	1.14498	1.58656	2.04151	2.50674	2.96281	3.39458	3.94762	4.56334
0.248014	0.593132	1.15261	1.61136	2.03169	2.50546	2.96723	3.40415	3.94447	4.56865
0.224312	0.593669	1.15021	1.6374	2.02448	2.50542	2.97217	3.40057	3.93865	4.56352
0.209114	0.590499	1.14161	1.65258	2.0191	2.51249	2.97365	3.38774	3.94395	4.56245
0.209674	0.589715	1.12979	1.65728	2.0157	2.51318	2.96637	3.3744	3.96095	4.54689
0.226128	0.595235	1.12568	1.65099	2.0172	2.50371	2.95442	3.36222	3.98157	4.53101
0.249659	0.604092	1.12286	1.63801	2.0301	2.4981	2.93692	3.36129	3.99109	4.53044
0.264644	0.615326	1.11733	1.62514	2.05161	2.49826	2.92604	3.37283	4.00029	4.53723
0.261218	0.628679	1.11457	1.6146	2.07809	2.50272	2.92764	3.39045	3.9945	4.55156
0.241088	0.636625	1.11646	1.60755	2.10466	2.50963	2.93757	3.40084	3.98247	4.57644
0.217333	0.633408	1.12041	1.6041	2.12945	2.5173	2.95245	3.40948	3.96363	4.58491
0.202005	0.619975	1.12493	1.60936	2.14989	2.51556	2.95769	3.41546	3.95065	4.58816
0.203294	0.603191	1.12908	1.61927	2.17103	2.50704	2.95901	3.4175	3.947	4.57881
0.217535	0.589686	1.12783	1.63066	2.18709	2.49469	2.95702	3.40619	3.95561	4.56066
0.239772	0.582932	1.1249	1.64381	2.19253	2.4836	2.94788	3.39186	3.96846	4.56319
0.259008	0.582376	1.1214	1.65909	2.18283	2.47846	2.94136	3.36929	3.9753	4.56761
0.259431	0.584514	1.11966	1.66977	2.16601	2.47602	2.93893	3.34401	3.98247	4.57845
0.240149	0.5886	1.11912	1.67269	2.14901	2.47691	2.9431	3.31749	3.99091	4.58912
0.216477	0.58995	1.118	1.67362	2.13956	2.47223	2.95325	3.29328	4.00306	4.59028
0.203642	0.590393	1.11729	1.6631	2.1312	2.45928	2.96531	3.29458	4.00501	4.58917
0.201998	0.592116	1.11717	1.64606	2.11978	2.45204	2.97454	3.30858	3.99897	4.58095
0.206736	0.59574	1.12127	1.62388	2.1072	2.45796	2.99207	3.33536	3.98742	4.57996
0.213047	0.603909	1.13022	1.60226	2.09404	2.48494	3.00866	3.36479	3.96989	4.57722
0.212626	0.612643	1.13828	1.58815	2.07825	2.52004	3.02075	3.38544	3.96136	4.56871
0.20088	0.617233	1.14045	1.58794	2.05824	2.55207	3.02465	3.40306	3.95916	4.56733
0.185414	0.614663	1.13495	1.5965	2.04217	2.56497	3.01604	3.41173	3.9608	4.56353
0.180638	0.6078	1.12629	1.61474	2.03501	2.5564	2.99404	3.41589	3.96288	4.56452
0.187507	0.600514	1.11847	1.6275	2.03861	2.53371	2.96994	3.4154	3.9705	4.55458
0.201948	0.597856	1.11389	1.6303	2.05225	2.50551	2.94415	3.41148	3.96507	4.55422
0.218975	0.600927	1.11482	1.62379	2.06886	2.48427	2.91766	3.40595	3.9609	4.53741
0.232264	0.60713	1.11605	1.61702	2.08403	2.47956	2.90072	3.40238	3.95945	4.52984
0.235986	0.610617	1.11422	1.61175	2.09065	2.49148	2.89573	3.39416	3.96341	4.51915
0.226715	0.611508	1.11281	1.61075	2.09088	2.50281	2.90047	3.38538	3.9753	4.5277

0.20916	0.612585	1.11275	1.61375	2.09509	2.50043	2.90248	3.37484	3.99103	4.54116
0.19201	0.610984	1.11407	1.62036	2.10557	2.48975	2.91371	3.36697	3.99648	4.56468
0.179948	0.607845	1.11498	1.63102	2.11792	2.47708	2.93107	3.35844	3.98218	4.59257
0.176034	0.601807	1.11855	1.64193	2.12565	2.46761	2.94581	3.3478	3.97034	4.61087
0.178958	0.60062	1.12271	1.65279	2.12831	2.46992	2.95131	3.33843	3.96364	4.62563
0.183868	0.605819	1.12325	1.65547	2.13021	2.48496	2.95076	3.32414	3.95478	4.62181
0.185003	0.611514	1.11979	1.65351	2.13617	2.51016	2.94907	3.31313	3.95613	4.60518
0.182233	0.609778	1.11863	1.65166	2.14272	2.53622	2.9442	3.32131	3.95935	4.59679
0.17965	0.599641	1.1185	1.65074	2.14723	2.54628	2.94313	3.35028	3.96393	4.59016
0.178835	0.586544	1.11689	1.65461	2.15093	2.5384	2.95127	3.38383	3.9646	4.59384
0.177517	0.574944	1.11184	1.65161	2.15126	2.51976	2.96474	3.4103	3.95936	4.59313
0.173811	0.566139	1.10675	1.63499	2.14704	2.49985	2.97859	3.41989	3.95835	4.60958
0.168009	0.569034	1.10407	1.61349	2.13631	2.47756	2.98563	3.40707	3.95986	4.60077
0.15864	0.583293	1.10673	1.59785	2.12113	2.45772	2.98458	3.37961	3.95955	4.58981
0.1477	0.604809	1.11305	1.58502	2.10452	2.44584	2.97748	3.36542	3.95436	4.56211
0.144257	0.62453	1.12882	1.58077	2.08941	2.44421	2.96973	3.36456	3.95943	4.55422
0.153599	0.635809	1.14653	1.581	2.07766	2.45651	2.97001	3.36824	3.95555	4.53025
0.169651	0.638434	1.15338	1.58391	2.06797	2.48125	2.98023	3.3701	3.96184	4.52985
0.181559	0.632143	1.14869	1.59193	2.05939	2.50609	2.99005	3.36635	3.94936	4.51282
0.185355	0.619555	1.13146	1.60053	2.04428	2.52485	2.99434	3.34648	3.95752	4.50854
0.183524	0.604066	1.10916	1.60884	2.02501	2.53341	2.98481	3.3333	3.96931	4.48541
0.17808	0.596001	1.09213	1.61693	2.00661	2.52992	2.97023	3.33144	3.97807	4.48283
0.170438	0.597099	1.08768	1.62715	1.99483	2.51611	2.95192	3.34191	3.97692	4.48901
0.161379	0.607253	1.10084	1.63377	1.99258	2.497	2.94006	3.37091	3.98298	4.50446
0.154341	0.618376	1.12104	1.63317	2.00593	2.48209	2.92444	3.3868	3.98842	4.53211
0.150367	0.623265	1.1348	1.63462	2.03533	2.47964	2.90692	3.40678	3.99404	4.55468
0.144961	0.620698	1.13359	1.63283	2.07603	2.48685	2.89756	3.41232	3.99319	4.57652
0.136423	0.612754	1.12308	1.63063	2.11704	2.49456	2.90608	3.40505	3.99835	4.59459
0.127264	0.59998	1.10987	1.62433	2.15137	2.50422	2.91813	3.39942	3.99832	4.60569
0.121278	0.584052	1.10224	1.62062	2.17455	2.51005	2.92293	3.39316	3.99251	4.60871
0.120566	0.57216	1.10535	1.62243	2.18606	2.51018	2.93096	3.38884	3.99103	4.6052
0.125071	0.569324	1.11488	1.62053	2.18628	2.50686	2.94187	3.38094	3.97534	4.60038
0.133159	0.576812	1.12709	1.61963	2.17858	2.50315	2.94898	3.37861	3.96283	4.59313
0.140883	0.591127	1.13684	1.61814	2.16862	2.50031	2.94533	3.37276	3.95305	4.59027
0.145315	0.604587	1.13666	1.62087	2.16124	2.49192	2.94313	3.36951	3.96017	4.58491
0.144315	0.615983	1.12933	1.62624	2.15626	2.4829	2.94877	3.36452	3.98749	4.58047
0.139983	0.625739	1.11651	1.62804	2.14551	2.47796	2.96147	3.35066	3.99155	4.5806
0.138224	0.630938	1.10952	1.62376	2.126	2.47318	2.97531	3.33433	3.98811	4.5845
0.142836	0.634746	1.11309	1.61519	2.10252	2.46977	2.99333	3.3214	3.9681	4.59262
0.146129	0.637184	1.12558	1.60277	2.07898	2.47141	2.99342	3.31407	3.94322	4.60287
0.144231	0.639631	1.13854	1.59374	2.0592	2.48589	2.98725	3.30844	3.91889	4.60421
0.137048	0.635854	1.14082	1.59742	2.04541	2.51266	2.97609	3.31298	3.91533	4.60884
0.128627	0.628709	1.1379	1.60769	2.03561	2.53985	2.96526	3.32048	3.93045	4.59517

0.123793	0.62171	1.13796	1.61716	2.02684	2.55039	2.95904	3.33885	3.94643	4.58963
0.121871	0.61684	1.1421	1.6241	2.01702	2.54709	2.94942	3.35606	3.9692	4.57713
0.121092	0.614571	1.14125	1.61953	2.01058	2.53002	2.94038	3.37613	3.97861	4.56578
0.117842	0.614013	1.13416	1.61101	2.00864	2.50944	2.93121	3.3845	3.9827	4.54543
0.113998	0.613634	1.12967	1.60285	2.01854	2.49047	2.92046	3.39051	3.97903	4.53455
0.112749	0.611064	1.13305	1.60065	2.03938	2.48159	2.91158	3.39765	3.97462	4.51555
0.114752	0.611231	1.14204	1.61246	2.06323	2.48054	2.91365	3.40581	3.97863	4.51931
0.12123	0.611984	1.15027	1.62642	2.08248	2.48399	2.92254	3.40215	3.97972	4.52952
0.132904	0.609019	1.15422	1.63254	2.098	2.48367	2.94047	3.39183	3.98483	4.54844
0.144872	0.60179	1.14795	1.62964	2.11382	2.47977	2.95727	3.38158	3.98264	4.56018
0.147949	0.595999	1.13278	1.62383	2.12832	2.47286	2.96902	3.37618	3.98611	4.57036
0.139729	0.599094	1.11183	1.61565	2.13692	2.45977	2.97631	3.36884	3.98943	4.56602
0.125507	0.610637	1.0983	1.61165	2.13871	2.44676	2.97676	3.3616	3.99206	4.55299
0.114825	0.624464	1.09462	1.61792	2.13292	2.44842	2.97034	3.36491	3.98896	4.54596
0.113285	0.634507	1.10441	1.62943	2.12696	2.46542	2.96022	3.36746	3.98605	4.54627
0.12092	0.642122	1.11623	1.63739	2.12718	2.48462	2.9635	3.37266	3.97898	4.54736
0.135512	0.642894	1.12096	1.63183	2.13054	2.50437	2.97172	3.38327	3.96858	4.55525
0.145564	0.633973	1.11938	1.61805	2.13036	2.52256	2.97135	3.3916	3.95277	4.55478
0.14156	0.624028	1.11654	1.60107	2.12835	2.53181	2.96263	3.40002	3.94818	4.55017
0.128383	0.620248	1.11463	1.58241	2.1227	2.52879	2.95092	3.39495	3.9523	4.54232
0.118175	0.622907	1.1168	1.5768	2.1089	2.5253	2.94335	3.38935	3.9543	4.53652
0.121701	0.629317	1.12583	1.57924	2.09244	2.51944	2.94226	3.3811	3.95079	4.54182
0.134807	0.640326	1.14079	1.58923	2.08277	2.51616	2.94305	3.36927	3.94892	4.55006
0.144288	0.65061	1.15675	1.60211	2.07832	2.51548	2.94642	3.35368	3.94849	4.5639
0.144889	0.651756	1.16951	1.61368	2.07142	2.51902	2.95404	3.33942	3.93906	4.57308
0.137312	0.641841	1.17408	1.61958	2.05785	2.51973	2.96713	3.33351	3.92912	4.58177
0.123286	0.622827	1.16927	1.61896	2.04018	2.51496	2.97322	3.33254	3.93848	4.58301
0.108443	0.606061	1.15396	1.60942	2.02738	2.50176	2.97024	3.33211	3.95192	4.58007
0.095866	0.59567	1.13152	1.60492	2.01958	2.48128	2.95819	3.33033	3.97309	4.57725
0.091594	0.592153	1.11431	1.61021	2.0165	2.46232	2.93312	3.34075	3.98464	4.58782
0.096564	0.597388	1.11499	1.62354	2.01493	2.44705	2.90573	3.36171	4.00363	4.60089
0.108319	0.607028	1.12713	1.63671	2.01967	2.43904	2.88716	3.38536	4.0234	4.6115
0.12118	0.615125	1.13833	1.64401	2.03386	2.44191	2.88564	3.4078	4.04009	4.62423
0.132415	0.620078	1.14148	1.64137	2.05419	2.45732	2.89903	3.42062	4.04105	4.62102
0.143475	0.62266	1.13758	1.62864	2.07986	2.48018	2.92461	3.42302	4.03349	4.60742
0.154699	0.619483	1.13007	1.61259	2.10783	2.50371	2.9524	3.41791	4.01579	4.58555
0.162996	0.615228	1.11991	1.59585	2.12912	2.52309	2.97467	3.40699	4.00118	4.57858
0.16974	0.616521	1.11858	1.58543	2.14222	2.53288	2.98468	3.40129	3.98053	4.56911
0.173005	0.624558	1.12304	1.57864	2.15082	2.53015	2.98581	3.39129	3.96476	4.56247
0.166513	0.63606	1.1296	1.5785	2.15912	2.51632	2.97914	3.38371	3.94897	4.54741
0.149453	0.648664	1.13849	1.5845	2.16864	2.49869	2.96365	3.37398	3.94504	4.53423
0.127932	0.657665	1.14923	1.58822	2.17213	2.48101	2.94595	3.37209	3.93282	4.51263
0.110524	0.654834	1.1492	1.59222	2.16581	2.47104	2.9385	3.36201	3.93051	4.49189

0.102345	0.641363	1.1474	1.59431	2.14924	2.46941	2.94204	3.34536	3.92189	4.48568
0.107839	0.624298	1.13697	1.59589	2.12574	2.47012	2.94624	3.32913	3.91995	4.49513
0.124195	0.609171	1.11967	1.59728	2.10035	2.47027	2.94887	3.31866	3.92526	4.50901
0.140838	0.600429	1.10522	1.59786	2.07562	2.47901	2.95188	3.30998	3.9402	4.51379
0.149052	0.599055	1.10335	1.59891	2.05194	2.49621	2.95449	3.30984	3.95768	4.52903
0.152224	0.607154	1.11984	1.60671	2.03398	2.51417	2.95669	3.32107	3.986	4.5369
0.156026	0.616143	1.14328	1.61882	2.02193	2.52756	2.96236	3.3383	4.00505	4.54393
0.16144	0.620453	1.16523	1.62788	2.01338	2.53529	2.97318	3.36078	4.01559	4.55237
0.165413	0.6189	1.17791	1.63372	2.01056	2.53648	2.96911	3.38718	4.01834	4.56333
0.1693	0.610407	1.18124	1.6352	2.01529	2.53019	2.95387	3.40894	3.99895	4.56921
0.172115	0.601447	1.17475	1.63525	2.01778	2.51395	2.93214	3.42381	3.98611	4.57107
0.171797	0.595994	1.16323	1.63522	2.0205	2.49403	2.91043	3.43376	3.97272	4.56264
0.169909	0.600186	1.14755	1.63061	2.02898	2.47939	2.89486	3.43548	3.9584	4.56971
0.169079	0.609555	1.12864	1.61891	2.04959	2.47557	2.88887	3.41866	3.95456	4.57622
0.168778	0.621328	1.11666	1.61122	2.08147	2.47935	2.89364	3.4048	3.95817	4.58607
0.163527	0.635421	1.1159	1.61334	2.11933	2.48407	2.90386	3.39318	3.96258	4.60427
0.151732	0.644763	1.12361	1.61789	2.1512	2.48761	2.91439	3.37848	3.97632	4.61297
0.140744	0.645967	1.13998	1.61918	2.17035	2.48857	2.9292	3.37493	3.9785	4.6037
0.136966	0.636839	1.15622	1.61021	2.17499	2.48507	2.9373	3.37393	3.98601	4.59247
0.142569	0.627108	1.16413	1.60066	2.17332	2.47632	2.93943	3.38257	3.98641	4.57446
0.156886	0.623255	1.15618	1.59298	2.16706	2.46485	2.93387	3.39649	3.97448	4.57775
0.174464	0.621565	1.14206	1.58645	2.15697	2.45085	2.93489	3.40582	3.96032	4.57683
0.190547	0.619215	1.13245	1.58071	2.14356	2.44903	2.93738	3.40872	3.94927	4.58121
0.198003	0.611614	1.13157	1.58042	2.13304	2.45754	2.94374	3.41143	3.96397	4.57428
0.193945	0.599722	1.1405	1.58634	2.12067	2.46912	2.94991	3.40703	3.98064	4.56837
0.181948	0.585375	1.14963	1.58976	2.10324	2.48284	2.9553	3.40076	3.99512	4.55545
0.173878	0.574758	1.15852	1.58904	2.08363	2.50123	2.95824	3.38633	3.99412	4.54554
0.178028	0.575293	1.16611	1.5936	2.07205	2.51764	2.95732	3.37468	3.97529	4.52811
0.193212	0.590076	1.17178	1.61073	2.06263	2.5328	2.95646	3.35689	3.94964	4.51793
0.213056	0.60941	1.17304	1.62486	2.05249	2.54609	2.95956	3.34544	3.92969	4.51263
0.225585	0.628618	1.16865	1.6353	2.04377	2.55641	2.9593	3.34206	3.9267	4.51763
0.221877	0.641925	1.15564	1.64244	2.04517	2.5597	2.95704	3.34403	3.93176	4.52952
0.20502	0.642748	1.14319	1.64454	2.0522	2.55524	2.95656	3.35257	3.94511	4.54617
0.180243	0.632017	1.13144	1.64523	2.05553	2.53365	2.95317	3.35839	3.95102	4.5564
0.157687	0.616256	1.12276	1.64224	2.05654	2.49846	2.94171	3.36604	3.9576	4.56668
0.147654	0.606513	1.11943	1.63647	2.05875	2.46373	2.928	3.37932	3.95992	4.56403
0.155744	0.605302	1.12013	1.62749	2.06448	2.44259	2.92236	3.3946	3.96946	4.55986
0.177225	0.613043	1.12319	1.61521	2.07463	2.43632	2.91731	3.40473	3.97453	4.55484
0.199524	0.619278	1.13287	1.59705	2.08715	2.44324	2.9156	3.40768	3.97835	4.56152
0.213748	0.618693	1.14255	1.58041	2.09722	2.45316	2.91539	3.41153	3.97431	4.57273
0.220582	0.607064	1.153	1.56982	2.10194	2.4657	2.92312	3.41089	3.97124	4.57966
0.220884	0.591814	1.16331	1.56913	2.10791	2.48242	2.93221	3.41514	3.96731	4.58876
0.218263	0.581759	1.17079	1.57447	2.12036	2.50814	2.94317	3.41528	3.97109	4.59433

0.215726	0.580992	1.17355	1.58181	2.13411	2.53033	2.94922	3.40307	3.9855	4.59198
0.217279	0.59262	1.16898	1.59033	2.14174	2.54031	2.95176	3.39648	3.98669	4.58088
0.223124	0.60493	1.16482	1.59439	2.14881	2.54004	2.94961	3.39539	3.98199	4.57125
0.230503	0.60788	1.1576	1.5915	2.15824	2.53297	2.93861	3.39558	3.97981	4.55244
0.235697	0.598359	1.1546	1.58693	2.16371	2.52197	2.92972	3.3955	3.96263	4.55007
0.23734	0.58668	1.15697	1.59168	2.16333	2.51193	2.9303	3.39893	3.95323	4.55371
0.235942	0.585653	1.16063	1.60597	2.15886	2.50471	2.93034	3.39572	3.95091	4.55661
0.231709	0.5989	1.1578	1.62272	2.14898	2.49461	2.93748	3.38517	3.95504	4.55328
0.223085	0.620203	1.15352	1.63774	2.13236	2.47704	2.94567	3.36699	3.95719	4.54938
0.209985	0.636854	1.14467	1.64697	2.11179	2.46198	2.95094	3.35571	3.9568	4.54642
0.198287	0.640093	1.13703	1.63838	2.0934	2.45817	2.95652	3.35956	3.97357	4.55293
0.194411	0.632915	1.13086	1.61624	2.07691	2.46405	2.95951	3.371	3.98459	4.56222
0.200065	0.619364	1.12791	1.59161	2.05762	2.47857	2.9556	3.38454	3.99652	4.58156
0.20941	0.610092	1.1253	1.57371	2.03868	2.49938	2.94546	3.39067	3.99234	4.59933
0.215422	0.610895	1.12429	1.57081	2.02382	2.51578	2.94063	3.39659	3.97761	4.6022
0.214571	0.61681	1.12544	1.57901	2.01354	2.52467	2.93718	3.40237	3.96403	4.5961
0.210078	0.614371	1.13105	1.58934	2.00691	2.52261	2.92934	3.40858	3.95511	4.5851
0.211956	0.606619	1.13554	1.59607	2.0124	2.51398	2.91731	3.40943	3.9587	4.57354
0.224385	0.595477	1.13186	1.60096	2.033	2.50225	2.90421	3.40623	3.9602	4.5793
0.243589	0.583309	1.12624	1.59915	2.05949	2.49477	2.89558	3.3976	3.95614	4.58332
0.265485	0.574615	1.12236	1.59331	2.08433	2.49658	2.88651	3.38855	3.94616	4.58884
0.286732	0.577323	1.12848	1.58948	2.10662	2.50446	2.89144	3.38365	3.94851	4.58553
0.299178	0.587409	1.13951	1.58944	2.12688	2.51623	2.91712	3.38531	3.94311	4.58918
0.295914	0.597417	1.1532	1.59854	2.14271	2.52569	2.95313	3.39582	3.95948	4.57882
0.275697	0.60466	1.16293	1.60611	2.14896	2.52299	2.98242	3.40352	3.96198	4.56746
0.243544	0.607035	1.16376	1.60796	2.15326	2.50838	2.99557	3.40707	3.96802	4.55933
0.211459	0.609813	1.15415	1.60557	2.16183	2.4889	2.98736	3.40449	3.96573	4.55388
0.189771	0.611234	1.14093	1.60166	2.17044	2.47238	2.9659	3.40678	3.96502	4.53559
0.181595	0.611413	1.13271	1.60096	2.17423	2.46415	2.95308	3.40477	3.95193	4.53407
0.185975	0.611633	1.12996	1.60104	2.17258	2.45878	2.95323	3.40434	3.94652	4.53671
0.201084	0.611334	1.13778	1.5967	2.16495	2.45402	2.96199	3.40552	3.95046	4.55211
0.220247	0.609781	1.15098	1.59113	2.15263	2.45401	2.97177	3.40869	3.96561	4.56982
0.23787	0.606039	1.16227	1.5929	2.13492	2.4569	2.97181	3.40012	3.99377	4.5899
0.252079	0.603492	1.16804	1.5974	2.11769	2.46608	2.95575	3.38823	4.01232	4.60497
0.262842	0.597285	1.16382	1.60405	2.09967	2.48225	2.92988	3.36524	4.02628	4.61242
0.270159	0.592015	1.15406	1.61071	2.08377	2.50279	2.90743	3.3581	4.02731	4.6088
0.273273	0.59025	1.1418	1.61187	2.07417	2.51742	2.89774	3.36329	4.02977	4.59806
0.271174	0.591404	1.12532	1.60415	2.06399	2.52316	2.89551	3.37502	4.0206	4.57559
0.268467	0.595211	1.10702	1.59243	2.05018	2.52437	2.89413	3.39179	4.01581	4.55377
0.270775	0.603812	1.09175	1.58287	2.03688	2.52512	2.89945	3.40691	4.00545	4.5315
0.277151	0.6158	1.08059	1.57818	2.02799	2.53001	2.91263	3.40889	3.98877	4.52412
0.280987	0.626547	1.07602	1.58287	2.02258	2.52969	2.93058	3.39419	3.97521	4.52755
0.277172	0.629362	1.08019	1.59038	2.0251	2.52279	2.94366	3.38598	3.94687	4.5351

0.266495	0.621265	1.09069	1.59583	2.03688	2.51756	2.94648	3.37774	3.92324	4.53981
0.252323	0.606578	1.10409	1.59336	2.05422	2.51805	2.9414	3.37911	3.90608	4.55212
0.240352	0.587422	1.11714	1.59187	2.07165	2.5191	2.94063	3.38868	3.89328	4.57229
0.234036	0.568296	1.12872	1.59307	2.08639	2.51767	2.94062	3.39329	3.89362	4.5745
0.233784	0.556219	1.14053	1.60299	2.09605	2.50895	2.94496	3.39941	3.90587	4.57387
0.23872	0.552413	1.15213	1.61976	2.10199	2.49726	2.94949	3.40187	3.92534	4.56023
0.246069	0.558619	1.16508	1.63559	2.11222	2.47858	2.9557	3.40373	3.94254	4.55494
0.254007	0.574678	1.17298	1.6413	2.12377	2.45928	2.96129	3.41451	3.95661	4.55317
0.262429	0.595648	1.17649	1.639	2.13382	2.44563	2.97194	3.42415	3.96634	4.54947
0.266661	0.615988	1.16797	1.63265	2.14818	2.44372	2.9841	3.42657	3.98409	4.56143
0.263336	0.629365	1.15143	1.61658	2.16945	2.45755	2.99376	3.42657	3.99587	4.57098
0.25616	0.631907	1.12813	1.60156	2.19117	2.48036	2.99321	3.41816	4.01455	4.58294
0.253637	0.623858	1.10938	1.58706	2.19805	2.50482	2.98332	3.4074	4.02469	4.58611
0.257697	0.612527	1.09782	1.57054	2.18732	2.51958	2.96412	3.40288	4.02951	4.59283
0.268125	0.601263	1.09719	1.56046	2.16499	2.52177	2.94178	3.39616	4.03171	4.59385
0.279109	0.591424	1.10207	1.56049	2.13801	2.51346	2.92774	3.3927	4.023	4.5951
0.28412	0.586372	1.10704	1.56785	2.11266	2.50359	2.92787	3.39215	4.01469	4.59839
0.282335	0.59045	1.10931	1.57358	2.08854	2.49129	2.94264	3.3951	3.99356	4.60151
0.277557	0.598815	1.10902	1.58528	2.06739	2.48491	2.95614	3.3891	3.9685	4.59237
0.272187	0.607981	1.10716	1.59886	2.05022	2.49186	2.96079	3.38605	3.93602	4.587
0.267998	0.612994	1.11196	1.61302	2.04221	2.50206	2.95549	3.37661	3.90365	4.58856
0.268395	0.612521	1.12195	1.6258	2.04222	2.5043	2.93561	3.37079	3.89549	4.59894
0.274538	0.61001	1.13173	1.62636	2.04077	2.50248	2.91785	3.36304	3.9033	4.59275
0.283251	0.599699	1.13628	1.61924	2.03881	2.50074	2.90605	3.35663	3.93465	4.59118
0.28566	0.586551	1.1392	1.61469	2.04063	2.49769	2.89859	3.35519	3.97394	4.57208
0.276329	0.576435	1.13361	1.61252	2.0487	2.49224	2.89095	3.36945	4.00861	4.56518
0.260399	0.572649	1.12115	1.61398	2.0594	2.4858	2.89202	3.39188	4.02608	4.56386
0.248353	0.577217	1.10778	1.61697	2.07323	2.48833	2.90413	3.41044	4.02102	4.56274
0.244145	0.588784	1.1025	1.62244	2.08864	2.5015	2.92492	3.42057	4.01002	4.56287
0.247302	0.602357	1.10832	1.62553	2.10128	2.51698	2.94643	3.42145	3.99314	4.55564
0.254298	0.612283	1.12184	1.62456	2.10405	2.53251	2.95846	3.41756	3.97565	4.56302
0.261794	0.621821	1.14033	1.61998	2.10262	2.53246	2.96213	3.42123	3.96981	4.5621
0.266119	0.628589	1.14814	1.61678	2.10512	2.51506	2.95529	3.42826	3.96817	4.55444
0.26984	0.633542	1.13753	1.61519	2.10957	2.48727	2.9508	3.43608	3.96954	4.5483
0.278175	0.633992	1.11943	1.61468	2.11735	2.46583	2.95414	3.43384	3.96872	4.53331
0.288717	0.631528	1.10609	1.61425	2.12585	2.46026	2.97981	3.42564	3.96381	4.53059
0.293121	0.621188	1.09906	1.61144	2.13922	2.4647	3.00956	3.41607	3.95383	4.53794
0.290082	0.604274	1.09367	1.60549	2.15323	2.47716	3.0288	3.40153	3.93585	4.54973
0.28428	0.58461	1.09512	1.60032	2.15953	2.49317	3.03058	3.38922	3.92296	4.56635
0.278108	0.569717	1.10692	1.59665	2.15376	2.50873	3.00993	3.38549	3.91835	4.58666
0.274255	0.565756	1.12088	1.59511	2.14425	2.52146	2.97684	3.38156	3.92592	4.60418
0.274042	0.572227	1.13072	1.59737	2.13174	2.52537	2.94651	3.37555	3.93375	4.62275
0.272032	0.588911	1.13539	1.59913	2.11935	2.52301	2.92109	3.36111	3.94624	4.62984

0.263524	0.606262	1.1386	1.5986	2.10364	2.52142	2.906	3.34716	3.95048	4.61889
0.251743	0.616511	1.13979	1.6012	2.07844	2.51817	2.90227	3.33709	3.9507	4.60968
0.245819	0.618868	1.14152	1.60985	2.04968	2.51475	2.9038	3.34246	3.94155	4.5895
0.249943	0.620404	1.14145	1.6197	2.02466	2.5115	2.90642	3.35509	3.94113	4.57831
0.263474	0.624374	1.13826	1.62812	2.00525	2.50461	2.90801	3.38189	3.94261	4.56687
0.280278	0.627044	1.13159	1.63433	1.99679	2.50373	2.91144	3.41059	3.94696	4.54415
0.296587	0.622353	1.1272	1.63294	2.00115	2.5048	2.92221	3.41971	3.95839	4.53678
0.306378	0.610103	1.12566	1.62568	2.01461	2.50685	2.94238	3.41145	3.96245	4.52714
0.302659	0.592538	1.12916	1.61939	2.03042	2.50589	2.96292	3.39743	3.9672	4.53176
0.288258	0.576762	1.13392	1.61793	2.041	2.50731	2.97698	3.40151	3.97224	4.53342
0.274704	0.575672	1.13813	1.62525	2.05118	2.50737	2.98075	3.41718	3.98797	4.53664
0.269087	0.588465	1.13606	1.63991	2.0695	2.50644	2.97416	3.42686	4.00353	4.54002
0.27281	0.609046	1.12897	1.65481	2.09398	2.50736	2.95997	3.43547	4.00901	4.54474
0.281952	0.629254	1.11943	1.66003	2.119	2.50415	2.95329	3.43449	4.00728	4.54941
0.2922	0.640156	1.11169	1.65137	2.13724	2.49756	2.95353	3.42105	3.98851	4.56988
0.297933	0.640396	1.10416	1.63149	2.14836	2.49546	2.95002	3.41622	3.96164	4.59042
0.295772	0.628557	1.10067	1.60406	2.15361	2.50055	2.95816	3.41709	3.94039	4.61271
0.286602	0.612302	1.10461	1.58414	2.14942	2.51118	2.96828	3.42295	3.93135	4.6237
0.278751	0.599472	1.1141	1.57279	2.13838	2.51881	2.97772	3.42211	3.93517	4.61956
0.273533	0.59636	1.12408	1.57159	2.13042	2.51659	2.98108	3.41227	3.95324	4.602
0.270391	0.600204	1.13196	1.57931	2.12287	2.50447	2.9724	3.40249	3.96371	4.59903
0.270124	0.606348	1.1371	1.59688	2.11279	2.4913	2.95027	3.3958	3.97184	4.60347
0.273162	0.608228	1.1405	1.61504	2.09947	2.47853	2.93035	3.39386	3.95735	4.61417
0.275521	0.607079	1.14567	1.62909	2.08314	2.47172	2.91639	3.39755	3.94601	4.61905
0.277211	0.607399	1.14815	1.6392	2.06799	2.48346	2.92059	3.40113	3.94449	4.61928
0.280547	0.606384	1.14765	1.64883	2.05491	2.51152	2.93592	3.39757	3.9374	4.62212
0.287834	0.600963	1.14404	1.65431	2.04517	2.53466	2.95111	3.38919	3.93309	4.61265
0.295748	0.591761	1.14296	1.65803	2.03904	2.5485	2.95771	3.36502	3.93486	4.60343
0.298837	0.584985	1.14189	1.65686	2.03993	2.55325	2.95257	3.34144	3.9332	4.58314
0.294797	0.585398	1.14641	1.65182	2.04406	2.54869	2.94011	3.32635	3.93601	4.56907
0.288289	0.59323	1.15987	1.65128	2.0435	2.53369	2.92494	3.31815	3.94408	4.54685
0.281776	0.605159	1.17341	1.65432	2.03448	2.51577	2.91973	3.33323	3.94208	4.52922
0.27787	0.619565	1.17456	1.65239	2.01796	2.4957	2.9223	3.35506	3.93349	4.50329
0.277148	0.629522	1.16518	1.63438	2.00906	2.48036	2.93696	3.38131	3.92639	4.50598
0.277954	0.630236	1.15108	1.61371	2.01777	2.47737	2.94816	3.40244	3.92673	4.522
0.279643	0.628582	1.13741	1.5951	2.04184	2.48239	2.96379	3.41437	3.9388	4.53562
0.282213	0.62734	1.13277	1.58989	2.06887	2.49422	2.9818	3.41424	3.95187	4.54673
0.28541	0.627071	1.13723	1.59861	2.09128	2.51267	2.98898	3.41004	3.96917	4.5578
0.288413	0.627478	1.14241	1.60858	2.10581	2.52393	2.99001	3.40617	3.97801	4.56818
0.292041	0.63037	1.14075	1.61267	2.11072	2.52261	2.98128	3.41181	3.97919	4.56883
0.294866	0.635106	1.13294	1.6053	2.10921	2.51713	2.972	3.42555	3.97638	4.57602
0.294727	0.632428	1.11966	1.59519	2.10738	2.50733	2.96369	3.43372	3.96589	4.56096
0.290877	0.617034	1.10827	1.58692	2.10914	2.49613	2.96396	3.43011	3.96092	4.55782

0.286165	0.593375	1.1066	1.58726	2.11827	2.48664	2.97221	3.42569	3.96096	4.56084
0.284882	0.574224	1.11535	1.60034	2.12924	2.48914	2.97979	3.42166	3.96081	4.5694
0.288899	0.565469	1.12566	1.61543	2.13162	2.50846	2.98385	3.41801	3.9632	4.57921
0.294716	0.569892	1.13447	1.63068	2.12646	2.53391	2.97728	3.41448	3.96011	4.57915
0.297074	0.583429	1.14192	1.63775	2.11517	2.54855	2.96587	3.40494	3.95663	4.5803
0.291246	0.602337	1.14896	1.64101	2.0974	2.54901	2.94923	3.39863	3.94761	4.57202
0.282606	0.618854	1.15876	1.63656	2.08301	2.53656	2.93461	3.39141	3.94473	4.57704
0.275878	0.628885	1.16604	1.63131	2.07534	2.52112	2.9229	3.38531	3.94396	4.59121
0.274864	0.629585	1.16963	1.62944	2.07331	2.50341	2.92039	3.38167	3.95316	4.59485
0.283508	0.623341	1.16829	1.6299	2.07448	2.48945	2.92561	3.38791	3.95334	4.60507
0.297762	0.614506	1.16194	1.63332	2.06767	2.48005	2.92927	3.39342	3.96038	4.61057
0.307396	0.604516	1.15509	1.63658	2.04347	2.48179	2.93954	3.40418	3.95399	4.60016
0.306542	0.600014	1.14731	1.6374	2.01278	2.48767	2.95655	3.41252	3.95507	4.58939
0.298981	0.602672	1.13648	1.63217	1.98744	2.4905	2.9654	3.41231	3.95429	4.58107
0.288799	0.608187	1.13209	1.62184	1.97622	2.48986	2.97521	3.40647	3.95737	4.57996
0.279333	0.613337	1.1328	1.60675	1.98197	2.49149	2.9784	3.39418	3.94837	4.58996
0.27671	0.615707	1.1381	1.59446	2.00488	2.49158	2.97005	3.38826	3.95068	4.58522
0.283242	0.613949	1.14305	1.58439	2.04491	2.49414	2.96494	3.37806	3.94897	4.58011
0.293028	0.61139	1.14221	1.57963	2.08568	2.49897	2.96134	3.37223	3.95589	4.56406
0.297837	0.611126	1.13866	1.58253	2.11524	2.51515	2.96123	3.36406	3.95983	4.54574
0.296268	0.611726	1.13692	1.59169	2.13447	2.53585	2.96537	3.34946	3.96658	4.53398
0.290774	0.610711	1.13692	1.59921	2.14413	2.55127	2.96949	3.34008	3.96768	4.53274
0.284077	0.605097	1.1378	1.59729	2.14716	2.56037	2.97008	3.3406	3.96615	4.52794
0.284265	0.598957	1.13855	1.58542	2.14658	2.55382	2.97214	3.34687	3.96175	4.52815
0.291391	0.600347	1.13959	1.5751	2.14679	2.54055	2.96972	3.35289	3.95969	4.53166
0.298698	0.60874	1.1472	1.5755	2.14599	2.53394	2.95884	3.36849	3.9597	4.54712
0.304324	0.621544	1.15373	1.58566	2.14107	2.53046	2.94342	3.38499	3.95783	4.55532
0.308269	0.630299	1.15435	1.59993	2.12985	2.52342	2.93838	3.39385	3.94643	4.5634
0.30606	0.630186	1.1504	1.61984	2.11926	2.50754	2.9425	3.40186	3.93381	4.56239
0.295591	0.614411	1.14458	1.63765	2.10986	2.48797	2.9534	3.39899	3.91735	4.55654
0.283069	0.588721	1.13968	1.65133	2.09681	2.47305	2.97282	3.39821	3.89734	4.55859
0.274741	0.566135	1.13876	1.65556	2.08208	2.46795	2.98748	3.39066	3.89061	4.5476
0.272703	0.560318	1.13484	1.64686	2.07208	2.47411	3.00083	3.38829	3.89534	4.54726
0.274688	0.567691	1.12923	1.6293	2.06512	2.48755	3.00714	3.3925	3.91777	4.54691
0.280393	0.581987	1.13143	1.61159	2.0601	2.49903	3.00609	3.39538	3.94329	4.55837
0.289098	0.597787	1.14072	1.59335	2.05687	2.5073	2.99957	3.39154	3.96517	4.57137
0.298298	0.613543	1.14766	1.58287	2.05568	2.50713	2.98227	3.38293	3.98164	4.58117
0.302351	0.628358	1.14796	1.58221	2.05353	2.49994	2.95158	3.37807	3.99001	4.57612
0.301907	0.642655	1.14903	1.59217	2.04842	2.48608	2.92198	3.37824	3.97849	4.56468
0.300539	0.649729	1.15011	1.59987	2.04447	2.47737	2.90539	3.38367	3.96984	4.56094
0.298379	0.645319	1.14867	1.60276	2.04664	2.47824	2.90459	3.39621	3.9544	4.56263
0.292657	0.633766	1.14205	1.59532	2.05806	2.49357	2.91972	3.41052	3.94408	4.57931
0.285844	0.622826	1.13246	1.5854	2.0764	2.51642	2.93617	3.42461	3.93458	4.58158

0.280114	0.614018	1.12487	1.57537	2.09559	2.53944	2.94697	3.43241	3.9343	4.56984
0.274591	0.608986	1.12612	1.57339	2.11071	2.55887	2.95831	3.43653	3.95206	4.56038
0.269015	0.604219	1.13356	1.581	2.11904	2.57064	2.95439	3.42822	3.97279	4.56038
0.266364	0.597916	1.14687	1.59016	2.12635	2.56921	2.94403	3.40781	4.00422	4.5549
0.268093	0.589882	1.15599	1.59824	2.13767	2.55316	2.93296	3.39612	4.0193	4.55673
0.274572	0.583851	1.15898	1.60197	2.14945	2.52789	2.9269	3.38778	4.02618	4.55747
0.281538	0.580595	1.16044	1.60211	2.15396	2.50593	2.92512	3.38125	4.01362	4.55063
0.285524	0.582567	1.1584	1.60395	2.1491	2.49524	2.92884	3.37818	3.99643	4.55746
0.283453	0.591104	1.15412	1.6066	2.142	2.49405	2.93409	3.3768	3.97093	4.56048
0.274788	0.602427	1.14727	1.61182	2.13734	2.49613	2.94692	3.37069	3.94039	4.55044
0.266282	0.610088	1.14524	1.61569	2.1341	2.49589	2.96743	3.36009	3.91962	4.54544
0.263822	0.610462	1.15077	1.61678	2.12659	2.49327	2.98463	3.35324	3.90353	4.54412
0.266286	0.606171	1.14739	1.61474	2.11304	2.49252	2.99768	3.35131	3.9001	4.53002
0.265645	0.600185	1.13458	1.60731	2.10074	2.49408	3.00391	3.35527	3.9091	4.5207
0.260614	0.596231	1.11795	1.60378	2.0947	2.49644	3.0065	3.35271	3.91707	4.53025
0.25759	0.595167	1.10593	1.60467	2.08892	2.49872	2.9981	3.35664	3.92881	4.54745
0.261269	0.599128	1.10106	1.60613	2.07972	2.49428	2.98758	3.36448	3.94141	4.56567
0.270131	0.606979	1.10581	1.60665	2.07106	2.4904	2.97896	3.37725	3.95458	4.57688
0.276795	0.613345	1.11576	1.60528	2.06331	2.49097	2.96762	3.38582	3.96384	4.58346
0.278462	0.617594	1.12793	1.59566	2.0567	2.50135	2.95359	3.39688	3.96345	4.59447
0.275019	0.616717	1.13695	1.58292	2.0495	2.51089	2.93697	3.40815	3.94928	4.60516
0.266722	0.610614	1.13821	1.56961	2.0431	2.51818	2.91913	3.41231	3.94556	4.60017
0.254287	0.602881	1.13423	1.56443	2.04157	2.52493	2.91401	3.40832	3.93722	4.59881
0.244783	0.595554	1.13612	1.56747	2.04354	2.52773	2.91579	3.40617	3.94198	4.59908
0.243122	0.591231	1.13818	1.57547	2.05155	2.52426	2.92056	3.39353	3.95369	4.59674
0.247681	0.589452	1.14283	1.5861	2.06537	2.51677	2.92063	3.38727	3.95136	4.59102
0.250644	0.591299	1.14821	1.59996	2.09043	2.50423	2.91464	3.38189	3.95234	4.58285
0.250747	0.595262	1.14802	1.61056	2.12116	2.49619	2.9088	3.37317	3.94765	4.5688
0.251224	0.603896	1.14171	1.61257	2.14742	2.49643	2.9135	3.3714	3.95118	4.54606
0.253642	0.6174	1.13777	1.60633	2.16199	2.50152	2.93183	3.37506	3.96076	4.5205
0.258976	0.628668	1.13824	1.59616	2.16751	2.5163	2.95035	3.38384	3.97046	4.51001
0.26499	0.632175	1.14395	1.59428	2.16403	2.53565	2.95711	3.39177	3.97851	4.4995
0.266787	0.622281	1.15217	1.59685	2.15647	2.54469	2.96034	3.39976	3.97553	4.50154
0.260174	0.604491	1.15943	1.60477	2.14663	2.54099	2.96354	3.41375	3.97725	4.51549
0.245063	0.58484	1.15366	1.61602	2.13719	2.52681	2.96613	3.42839	3.9774	4.53084
0.227498	0.57006	1.13686	1.62755	2.12828	2.50538	2.97068	3.43643	3.97772	4.54626
0.215162	0.565839	1.11372	1.63218	2.12103	2.48701	2.97966	3.43458	3.974	4.5547
0.213612	0.572395	1.09365	1.62703	2.11813	2.48008	2.99165	3.42902	3.96953	4.55965
0.220458	0.583536	1.08505	1.61577	2.11782	2.47836	2.99893	3.41857	3.96957	4.56338
0.234036	0.595344	1.08982	1.60641	2.11587	2.4854	2.98669	3.40318	3.96795	4.57126
0.25093	0.605319	1.10761	1.60265	2.108	2.49652	2.9712	3.38999	3.97327	4.58223
0.264451	0.611062	1.13114	1.59772	2.09424	2.50919	2.94922	3.3778	3.9753	4.5836
0.267002	0.613654	1.15289	1.59186	2.07823	2.51747	2.93157	3.36884	3.97381	4.59118

0.258664	0.618474	1.16478	1.58535	2.065	2.5163	2.92114	3.36153	3.96548	4.57442
0.241977	0.629318	1.16396	1.58315	2.05666	2.5094	2.91999	3.35821	3.95186	4.56737
0.224532	0.638137	1.15191	1.58221	2.05311	2.50184	2.91904	3.36236	3.93409	4.5633
0.212405	0.639123	1.13286	1.586	2.05111	2.49505	2.91855	3.36577	3.91814	4.56285
0.204982	0.632897	1.11732	1.59568	2.05199	2.49653	2.91281	3.35899	3.91412	4.56943
0.199917	0.620958	1.11582	1.6068	2.05487	2.51145	2.91304	3.35515	3.91352	4.57696
0.197963	0.60575	1.13045	1.62115	2.06377	2.52808	2.92041	3.35618	3.91643	4.57232
0.200773	0.593364	1.15381	1.62484	2.08085	2.53657	2.92318	3.35224	3.92277	4.56912
0.208525	0.589318	1.17367	1.61658	2.10056	2.53267	2.92273	3.35745	3.93054	4.56397
0.219705	0.593021	1.17246	1.6006	2.12215	2.52255	2.92995	3.37207	3.93408	4.57074
0.231287	0.600695	1.15601	1.59365	2.14417	2.51234	2.94425	3.39476	3.95051	4.56656
0.237041	0.605261	1.12988	1.59472	2.1603	2.51096	2.96228	3.41322	3.96469	4.58167
0.231883	0.610496	1.10566	1.60072	2.16861	2.51222	2.97553	3.41921	3.98695	4.58358
0.216315	0.617504	1.09116	1.61043	2.17003	2.51413	2.97952	3.42041	3.99509	4.59767
0.198808	0.617263	1.09433	1.62255	2.16211	2.51402	2.98409	3.42061	3.99057	4.61376
0.188223	0.608452	1.11036	1.63306	2.14986	2.51376	2.98978	3.42279	3.98546	4.61255
0.184092	0.596102	1.13348	1.63676	2.13656	2.50774	2.98425	3.41482	3.9782	4.60575
0.187538	0.590161	1.15071	1.63759	2.1297	2.49799	2.98023	3.416	3.97074	4.59277
0.195263	0.592004	1.15826	1.63924	2.12753	2.49037	2.9742	3.41978	3.96753	4.56953
0.200724	0.601609	1.1545	1.63577	2.12792	2.47797	2.96103	3.41924	3.97024	4.55046
0.200821	0.616118	1.13999	1.629	2.12228	2.46425	2.94454	3.41359	3.96626	4.53866
0.200983	0.626773	1.12177	1.61694	2.11245	2.45922	2.92917	3.39532	3.96077	4.52285
0.202607	0.633791	1.11459	1.60546	2.10773	2.46041	2.92101	3.37437	3.95481	4.49938
0.201128	0.638383	1.11929	1.5972	2.10569	2.46771	2.91854	3.36412	3.95144	4.4809
0.193682	0.644698	1.12854	1.58874	2.1059	2.48133	2.91587	3.37185	3.95265	4.47003
0.181689	0.651321	1.13351	1.59289	2.10173	2.50174	2.91918	3.38889	3.94086	4.47884
0.170777	0.653662	1.13061	1.60925	2.08837	2.51604	2.92408	3.41123	3.93294	4.49415
0.165932	0.650093	1.12412	1.62748	2.06613	2.52589	2.92655	3.42112	3.92451	4.51351
0.167489	0.63851	1.11737	1.63691	2.04141	2.5322	2.9279	3.42628	3.9145	4.54558
0.173362	0.622288	1.11319	1.63742	2.02458	2.53869	2.92408	3.42044	3.92242	4.57348
0.179045	0.60664	1.11131	1.63653	2.02444	2.54308	2.92277	3.41127	3.92888	4.5854
0.181275	0.596215	1.11551	1.63549	2.03791	2.54327	2.92105	3.40234	3.94414	4.6011
0.178014	0.594638	1.11974	1.63349	2.05658	2.53813	2.91954	3.39513	3.95532	4.6049
0.171556	0.600605	1.12115	1.62742	2.07702	2.52347	2.92114	3.39409	3.96705	4.61144
0.169217	0.611252	1.11793	1.61801	2.09613	2.5023	2.92604	3.39456	3.97149	4.61381
0.174261	0.622072	1.11556	1.60634	2.11009	2.48046	2.93327	3.38132	3.97378	4.60564
0.18156	0.62748	1.11894	1.59584	2.12337	2.46729	2.94684	3.37039	3.97436	4.59127
0.184414	0.625493	1.11967	1.59679	2.13915	2.46603	2.9611	3.35234	3.98275	4.57636
0.178306	0.621187	1.12321	1.608	2.15549	2.47106	2.97755	3.3425	3.99578	4.56777
0.165186	0.617385	1.13305	1.62713	2.16985	2.47672	2.98519	3.33891	4.00303	4.55208
0.150218	0.61901	1.14613	1.64562	2.17531	2.47962	2.9846	3.34692	3.99312	4.54946
0.138158	0.626482	1.15507	1.65513	2.1714	2.47379	2.98165	3.36385	3.97644	4.56273
0.132711	0.637235	1.15699	1.6582	2.1615	2.46671	2.97418	3.37823	3.95997	4.57847

0.137165	0.645281	1.15512	1.65181	2.15014	2.46073	2.96511	3.39064	3.93959	4.58987
0.14935	0.650441	1.14795	1.64429	2.143	2.45862	2.94792	3.39845	3.9292	4.59574
0.1619	0.652894	1.1387	1.63172	2.13235	2.47177	2.93427	3.39333	3.92375	4.60313
0.16961	0.655428	1.13032	1.6217	2.11272	2.50105	2.92873	3.39926	3.93896	4.6061
0.171271	0.656461	1.12385	1.61961	2.08658	2.53342	2.92118	3.40246	3.95599	4.5964
0.166083	0.651787	1.1206	1.62317	2.06483	2.55777	2.91367	3.39803	3.96242	4.59136
0.155738	0.641463	1.11834	1.62335	2.04834	2.56496	2.90825	3.39876	3.96395	4.57991
0.144227	0.627888	1.11217	1.61888	2.03715	2.55947	2.91212	3.40373	3.97479	4.57272
0.137227	0.614987	1.10365	1.60933	2.02487	2.54634	2.92245	3.41189	3.97501	4.56277
0.135321	0.607318	1.09423	1.59814	2.01217	2.5309	2.93393	3.4107	3.97578	4.55671
0.138294	0.609046	1.09112	1.58986	2.00325	2.51785	2.93695	3.41224	3.97597	4.55252
0.14161	0.619138	1.09283	1.58934	2.00527	2.50684	2.93805	3.40923	3.97556	4.54243
0.141649	0.633108	1.1055	1.60443	2.0228	2.4947	2.93945	3.41098	3.95834	4.52856
0.140824	0.644611	1.12197	1.63174	2.05047	2.48346	2.94194	3.40386	3.94565	4.51738
0.141251	0.650496	1.13695	1.65756	2.08218	2.4679	2.94646	3.39795	3.9369	4.51531
0.142933	0.65123	1.14593	1.67002	2.11451	2.45125	2.94892	3.39506	3.94057	4.51973
0.146523	0.64876	1.1491	1.66827	2.14516	2.44497	2.94371	3.39979	3.94988	4.53018
0.149854	0.642056	1.14504	1.65742	2.16555	2.45073	2.94271	3.41458	3.97405	4.53439
0.15036	0.634589	1.13701	1.64778	2.16896	2.47326	2.93584	3.42247	4.00157	4.53869
0.143535	0.630545	1.12766	1.64211	2.16495	2.4938	2.93112	3.42794	4.02186	4.54168
0.132694	0.629869	1.12348	1.63809	2.15796	2.50539	2.93873	3.42644	4.01832	4.53692
0.123908	0.630955	1.12536	1.63522	2.14871	2.50302	2.96653	3.41677	4.00428	4.5359
0.117818	0.62721	1.12812	1.63638	2.13093	2.49562	2.99553	3.40607	3.98104	4.54525
0.111771	0.621993	1.13209	1.63699	2.1071	2.49176	3.0072	3.39542	3.95472	4.5621
0.109305	0.617486	1.13125	1.63555	2.0828	2.49201	2.99979	3.38598	3.93398	4.58164
0.114606	0.616365	1.12498	1.63708	2.06666	2.4964	2.97177	3.37809	3.9218	4.59634
0.126271	0.620976	1.12095	1.6348	2.06045	2.50406	2.93555	3.37177	3.92472	4.59725
0.137597	0.630281	1.11907	1.62209	2.06463	2.5155	2.90074	3.36608	3.94076	4.60463
0.144717	0.638018	1.12559	1.6038	2.0798	2.52372	2.88167	3.35931	3.94276	4.61076
0.145313	0.64515	1.13061	1.58198	2.09693	2.52799	2.87594	3.35662	3.94808	4.60794
0.13556	0.653426	1.13566	1.57267	2.1057	2.5266	2.8799	3.35394	3.95789	4.60589
0.11719	0.657544	1.13713	1.58146	2.10379	2.52221	2.8886	3.36288	3.96768	4.59634
0.102922	0.655263	1.13201	1.60289	2.10104	2.5115	2.90865	3.37388	3.97927	4.5858
0.102199	0.646599	1.12788	1.63616	2.097	2.49858	2.92211	3.39515	3.98419	4.58403
0.115042	0.635615	1.12153	1.66175	2.09043	2.48786	2.93513	3.41804	3.98194	4.58318
0.132398	0.625882	1.11436	1.67396	2.07764	2.48091	2.95724	3.42889	3.97998	4.59461
0.144642	0.619756	1.11416	1.67387	2.06168	2.47869	2.97878	3.42885	3.98376	4.59525
0.145073	0.617892	1.11684	1.67126	2.04262	2.48306	2.99442	3.42056	3.97918	4.59567
0.134809	0.618058	1.12298	1.6634	2.02793	2.48946	3.00152	3.40811	3.9713	4.58818
0.120761	0.618874	1.1275	1.655	2.01801	2.4945	3.00093	3.39555	3.96993	4.58767
0.108435	0.621165	1.12845	1.64376	2.0201	2.49534	2.98846	3.38918	3.95715	4.59053
0.100369	0.628606	1.12682	1.63416	2.03713	2.49695	2.97234	3.40149	3.94726	4.59489
0.098445	0.633904	1.12752	1.63676	2.06747	2.49677	2.95565	3.41452	3.93943	4.59011

0.104482	0.637392	1.12765	1.64112	2.09904	2.49556	2.94605	3.42339	3.9465	4.57483
0.116512	0.638622	1.12864	1.63688	2.12437	2.49292	2.94553	3.4162	3.96035	4.56265
0.126387	0.633692	1.12624	1.62965	2.14675	2.49183	2.95008	3.39909	3.96769	4.54212
0.128935	0.625611	1.12495	1.61785	2.16307	2.49162	2.94927	3.38282	3.969	4.53324
0.125982	0.618376	1.12125	1.60978	2.1701	2.48694	2.93693	3.37236	3.96121	4.52778
0.12132	0.614632	1.11177	1.60938	2.16113	2.4762	2.91605	3.37137	3.96089	4.52292
0.119082	0.612862	1.0984	1.61589	2.14639	2.4618	2.90469	3.37979	3.95945	4.53013
0.122064	0.613728	1.08785	1.6208	2.13273	2.45713	2.90698	3.38638	3.96322	4.53537
0.129347	0.616934	1.08877	1.61973	2.11868	2.4575	2.91416	3.39369	3.95677	4.53984
0.138022	0.625209	1.10251	1.61558	2.10778	2.46747	2.92806	3.39762	3.96069	4.54797
0.14283	0.636574	1.12637	1.61018	2.0996	2.49281	2.95272	3.40248	3.96251	4.55436
0.138337	0.648077	1.14403	1.61105	2.09022	2.52518	2.97115	3.39845	3.94256	4.56994
0.124824	0.657059	1.15385	1.62179	2.08066	2.54921	2.97634	3.4013	3.93384	4.56408
0.109941	0.661639	1.15969	1.63808	2.06848	2.56234	2.9774	3.41026	3.93475	4.57299
0.102931	0.653369	1.15981	1.65453	2.05005	2.56269	2.96895	3.41121	3.94104	4.58346
0.105922	0.633419	1.15133	1.66027	2.02656	2.55115	2.96384	3.40639	3.95456	4.60308
0.11054	0.609622	1.14073	1.65631	2.00474	2.53555	2.96188	3.39489	3.97354	4.61158
0.11228	0.594487	1.13394	1.65229	1.99528	2.51968	2.96105	3.37957	3.98539	4.62627
0.112707	0.59369	1.13034	1.64681	1.99692	2.50528	2.96101	3.37741	3.9866	4.63403
0.115132	0.60004	1.13125	1.64612	2.00663	2.48927	2.96347	3.37699	3.9811	4.64058
0.120457	0.611524	1.13656	1.64264	2.02471	2.46612	2.96506	3.38191	3.97494	4.63461
0.126931	0.623647	1.14595	1.63676	2.04744	2.44521	2.96675	3.38652	3.96835	4.62361
0.133196	0.63199	1.14898	1.63095	2.07152	2.43025	2.96961	3.39356	3.96705	4.62553
0.13802	0.635551	1.13923	1.62013	2.09818	2.42454	2.9676	3.39038	3.96802	4.61006
0.140174	0.628952	1.12784	1.60656	2.12904	2.43214	2.96478	3.37543	3.97602	4.60303
0.138848	0.615574	1.11522	1.59554	2.15992	2.45753	2.9526	3.35358	3.99167	4.58726
0.136059	0.598069	1.10017	1.5974	2.18558	2.49198	2.92536	3.34887	3.99109	4.56218
0.132425	0.586927	1.08463	1.60264	2.19511	2.52001	2.90338	3.35391	3.9937	4.54
0.130194	0.587483	1.07601	1.61242	2.18524	2.53573	2.89413	3.35991	3.98505	4.52738
0.129696	0.603502	1.07821	1.6224	2.16597	2.53285	2.90159	3.36636	3.97823	4.5148
0.1324	0.626274	1.08431	1.63432	2.14231	2.51795	2.91645	3.36236	3.96758	4.51103
0.140183	0.643319	1.0931	1.64874	2.11464	2.50382	2.93246	3.35839	3.95889	4.51994
0.151092	0.650919	1.10353	1.6566	2.09078	2.49787	2.94527	3.35398	3.95331	4.53903
0.162032	0.650613	1.12007	1.6531	2.07248	2.49766	2.95517	3.36356	3.94533	4.55694
0.169485	0.6447	1.1416	1.64062	2.05896	2.50108	2.96492	3.37837	3.94258	4.57181
0.171605	0.631901	1.16122	1.62615	2.05317	2.5045	2.96722	3.40428	3.95272	4.57753
0.165929	0.616739	1.17788	1.61529	2.05592	2.50055	2.96584	3.42914	3.95639	4.57789
0.153294	0.605953	1.18565	1.60906	2.06416	2.48603	2.96032	3.43731	3.95883	4.57392
0.138818	0.604514	1.18278	1.60958	2.0723	2.46834	2.95175	3.42809	3.95739	4.57833
0.127042	0.609223	1.16842	1.61158	2.07491	2.46614	2.94612	3.42234	3.96749	4.57718
0.12217	0.612131	1.15506	1.62008	2.06553	2.47751	2.95082	3.42524	3.97881	4.5864
0.128595	0.611405	1.14636	1.63498	2.05081	2.49274	2.95903	3.41314	3.9884	4.5964
0.142439	0.605257	1.13489	1.64551	2.04062	2.50259	2.96849	3.39742	3.98854	4.60998

0.152875	0.593307	1.1228	1.64806	2.03492	2.51094	2.97879	3.37626	3.99317	4.60246
0.155426	0.5815	1.10625	1.64242	2.03355	2.51032	2.98815	3.35687	3.98191	4.58969
0.151735	0.580528	1.09312	1.62424	2.03795	2.49819	2.98319	3.35402	3.97678	4.58972
0.14551	0.58776	1.08358	1.60193	2.0468	2.48539	2.97266	3.36152	3.97308	4.59128
0.142904	0.599739	1.08262	1.58923	2.06418	2.48562	2.96311	3.38224	3.97593	4.59463
0.148486	0.611658	1.09717	1.59269	2.08527	2.4941	2.95593	3.40477	3.97292	4.58769
0.164403	0.622307	1.11876	1.60024	2.10174	2.50319	2.94729	3.4111	3.9621	4.58653
0.185964	0.630532	1.13796	1.61109	2.11626	2.50127	2.94325	3.40837	3.95197	4.56952
0.202248	0.635629	1.14997	1.61921	2.13349	2.48758	2.93965	3.39561	3.92118	4.55014
0.202707	0.635967	1.15353	1.62092	2.15016	2.47381	2.93466	3.37187	3.90697	4.53918
0.188041	0.634373	1.15088	1.62152	2.15937	2.472	2.92985	3.34228	3.9005	4.53926
0.167919	0.629173	1.13836	1.62451	2.1598	2.47797	2.92723	3.33309	3.90969	4.55786
0.153571	0.623068	1.11639	1.63054	2.15682	2.48797	2.91919	3.33804	3.92448	4.57996
0.154905	0.621066	1.09806	1.63836	2.15325	2.49893	2.91643	3.35587	3.95811	4.60892
0.169689	0.618256	1.09326	1.64257	2.14456	2.50694	2.92088	3.36989	3.99936	4.61591
0.186036	0.615392	1.10515	1.63968	2.12525	2.50949	2.93264	3.37593	4.01802	4.6171
0.191701	0.609225	1.12848	1.6266	2.0984	2.51003	2.9508	3.37277	4.01943	4.59876
0.185005	0.599384	1.15053	1.61694	2.06763	2.50861	2.96118	3.36666	4.01337	4.59073
0.173685	0.589766	1.15985	1.61298	2.04104	2.50177	2.96678	3.36486	3.99669	4.57504
0.165876	0.584308	1.15795	1.61215	2.02777	2.49109	2.96812	3.36429	3.98365	4.57893
0.162715	0.582403	1.14973	1.61159	2.02594	2.47731	2.97364	3.37269	3.97778	4.58899
0.163883	0.580433	1.13699	1.60545	2.02495	2.46472	2.96849	3.38213	3.97199	4.5923
0.16964	0.577874	1.12716	1.60164	2.02119	2.45564	2.96577	3.38495	3.9816	4.59209
0.177457	0.576827	1.12303	1.60342	2.01327	2.45525	2.96257	3.38899	3.98541	4.58727
0.187615	0.575124	1.11781	1.60768	2.01055	2.45704	2.95777	3.39109	3.983	4.58401
0.200965	0.577765	1.1136	1.6071	2.0191	2.46313	2.94921	3.39547	3.9783	4.572
0.217256	0.588995	1.11003	1.60171	2.03604	2.47453	2.94082	3.39543	3.97054	4.56532
0.229914	0.606809	1.1025	1.59687	2.0585	2.48472	2.93562	3.39987	3.96498	4.54246
0.231019	0.624085	1.09564	1.59109	2.08321	2.48779	2.93691	3.41047	3.95094	4.52886
0.222451	0.633111	1.0943	1.58946	2.10344	2.4874	2.95212	3.42575	3.95136	4.54365
0.212876	0.637489	1.10443	1.59373	2.11264	2.49426	2.96844	3.43698	3.95824	4.56084
0.207572	0.634439	1.11882	1.60647	2.11735	2.50356	2.97836	3.44729	3.9635	4.57818
0.207694	0.627362	1.13107	1.6246	2.1253	2.51444	2.97347	3.44415	3.97355	4.57432
0.21017	0.619567	1.14198	1.63169	2.13537	2.52508	2.96244	3.42721	3.9736	4.56666
0.210624	0.613196	1.14627	1.63135	2.14296	2.52757	2.94722	3.41218	3.96077	4.5447
0.204269	0.607923	1.13719	1.63228	2.15024	2.52348	2.93589	3.39078	3.95934	4.54088
0.194177	0.599311	1.12513	1.63488	2.16051	2.51243	2.92942	3.36979	3.95591	4.5481
0.189768	0.589259	1.12102	1.62953	2.16438	2.49518	2.93023	3.35431	3.96622	4.56523
0.193001	0.583513	1.11973	1.619	2.15127	2.47251	2.93173	3.34228	3.97623	4.56779
0.202208	0.58264	1.12332	1.60844	2.12481	2.45785	2.93076	3.33113	3.96344	4.5744
0.214405	0.584888	1.12262	1.59683	2.0934	2.44873	2.92622	3.33164	3.95105	4.56478
0.227126	0.590341	1.11691	1.59224	2.06342	2.45045	2.92663	3.33802	3.9371	4.56988
0.237874	0.598942	1.1101	1.58793	2.04181	2.46057	2.93408	3.35638	3.94555	4.56954

0.243634	0.606447	1.10351	1.58415	2.03014	2.47393	2.95027	3.36952	3.96394	4.59107
0.243774	0.606235	1.10106	1.58328	2.02942	2.48335	2.96534	3.38252	3.98546	4.60592
0.241352	0.600003	1.09816	1.58337	2.03828	2.48703	2.97267	3.3899	4.00021	4.61806
0.241015	0.58939	1.09495	1.57914	2.05095	2.49027	2.97918	3.39405	4.00723	4.62958
0.244551	0.576223	1.09463	1.57911	2.05559	2.50078	2.98659	3.38933	4.01346	4.62566
0.25143	0.568042	1.10001	1.5864	2.04799	2.52042	2.99633	3.38476	4.00701	4.61014
0.254725	0.570223	1.11156	1.60071	2.03606	2.53458	3.00368	3.37523	4.00412	4.5961
0.248059	0.583164	1.12133	1.61771	2.0266	2.54044	3.00002	3.3776	3.99225	4.57708
0.231896	0.602391	1.12949	1.62789	2.02699	2.53537	2.98772	3.3828	3.97213	4.5591
0.215326	0.616582	1.13391	1.63299	2.03551	2.51845	2.96953	3.39069	3.97198	4.55351
0.207135	0.620283	1.13286	1.63088	2.05074	2.49389	2.94291	3.39421	3.96523	4.5516
0.210522	0.614344	1.13166	1.62684	2.06787	2.47126	2.91271	3.40267	3.96268	4.55758
0.222907	0.601715	1.13191	1.61984	2.08297	2.46118	2.90378	3.40194	3.96589	4.55819
0.239293	0.592818	1.13268	1.61374	2.09395	2.45528	2.90276	3.39898	3.97702	4.55722
0.254411	0.59568	1.12773	1.61414	2.1064	2.44794	2.91041	3.38828	3.98064	4.55682
0.263491	0.608	1.12055	1.61469	2.12089	2.44221	2.9202	3.36638	3.97192	4.55486
0.26653	0.61955	1.11557	1.61119	2.13313	2.43708	2.92721	3.35264	3.9664	4.56029
0.269023	0.623593	1.11698	1.60745	2.13564	2.43601	2.93881	3.34218	3.95702	4.55906
0.274915	0.619463	1.11688	1.59736	2.12825	2.4452	2.94622	3.35006	3.94847	4.57306
0.284007	0.611061	1.11036	1.57936	2.11042	2.46976	2.94965	3.36864	3.94601	4.56795
0.289811	0.602029	1.10106	1.56801	2.09142	2.50284	2.95602	3.39263	3.96361	4.55982
0.284498	0.591102	1.10121	1.57102	2.08001	2.5301	2.96993	3.41048	3.98687	4.55188
0.268314	0.578885	1.10767	1.58562	2.07532	2.54403	2.97907	3.42593	4.00758	4.54955
0.247864	0.568232	1.11257	1.60346	2.07519	2.54513	2.97905	3.44221	4.02433	4.54745
0.231752	0.561166	1.11616	1.61928	2.07593	2.52937	2.97601	3.44365	4.02426	4.54393
0.224217	0.554829	1.11683	1.62695	2.07611	2.51091	2.97516	3.42985	4.00775	4.54605
0.223669	0.552469	1.10785	1.6227	2.07148	2.49806	2.97138	3.41241	3.98262	4.54112
0.231802	0.557637	1.09666	1.61453	2.06383	2.48974	2.97025	3.3993	3.96464	4.53403
0.24861	0.574943	1.08985	1.61114	2.04938	2.47675	2.9687	3.39182	3.95189	4.54357
0.26985	0.597144	1.09504	1.61556	2.03229	2.45977	2.96352	3.39012	3.957	4.54923
0.290238	0.616662	1.11227	1.62048	2.02072	2.44949	2.96242	3.39521	3.97214	4.55766
0.300759	0.628362	1.1285	1.62027	2.02077	2.45548	2.96335	3.40294	3.97729	4.573
0.300991	0.632722	1.13785	1.60866	2.03055	2.47034	2.95796	3.40049	3.96983	4.57586
0.296179	0.631356	1.13906	1.59501	2.04513	2.48797	2.95592	3.39609	3.95529	4.56393
0.289514	0.62673	1.1266	1.5786	2.05815	2.51207	2.95131	3.37805	3.94962	4.54719
0.283913	0.62284	1.11017	1.57003	2.0686	2.53197	2.94743	3.35739	3.94757	4.53332
0.28221	0.618123	1.10648	1.57492	2.08581	2.53831	2.94018	3.33235	3.9556	4.53888
0.28528	0.610012	1.11138	1.58598	2.10618	2.52716	2.93246	3.31711	3.96205	4.55791
0.288106	0.595122	1.12653	1.59376	2.12143	2.50302	2.9243	3.31884	3.96721	4.57174
0.285167	0.580368	1.14462	1.59516	2.13026	2.48115	2.92926	3.33279	3.97498	4.5918
0.275113	0.572909	1.15251	1.5969	2.13072	2.46904	2.93725	3.34577	3.97817	4.58625
0.260073	0.572681	1.15255	1.60154	2.12695	2.46855	2.93961	3.36342	3.97011	4.57824
0.24613	0.575526	1.14346	1.60966	2.12676	2.47486	2.93219	3.37635	3.96356	4.57864

0.240806	0.577755	1.13182	1.61671	2.12558	2.48543	2.92307	3.38639	3.95292	4.58079
0.247675	0.57722	1.11531	1.61718	2.12119	2.49482	2.91458	3.39375	3.94805	4.57654
0.261404	0.572473	1.1042	1.61361	2.11077	2.49724	2.91721	3.39483	3.94495	4.58737
0.276009	0.571988	1.09875	1.60585	2.09171	2.49387	2.93223	3.39099	3.95309	4.5713
0.288103	0.583962	1.1027	1.59413	2.07018	2.48677	2.94812	3.38531	3.95681	4.54866
0.298403	0.605443	1.10789	1.58394	2.05195	2.48257	2.95658	3.36991	3.96755	4.5199
0.305027	0.624777	1.1125	1.58172	2.03607	2.487	2.96023	3.35463	3.97653	4.50824
0.305432	0.63314	1.11306	1.5863	2.03203	2.49022	2.96463	3.35218	3.98768	4.50198
0.30128	0.634524	1.1161	1.59595	2.03535	2.48478	2.97301	3.35722	4.01103	4.52621
0.29508	0.630188	1.12694	1.60266	2.03762	2.47258	2.98457	3.3736	4.02664	4.54359
0.286714	0.621285	1.13953	1.60258	2.0367	2.46387	2.99856	3.39934	4.04433	4.56486
0.280912	0.608328	1.14518	1.59768	2.03502	2.46107	3.01359	3.42602	4.0368	4.56458
0.280771	0.5915	1.14326	1.58912	2.03365	2.46629	3.01687	3.44765	4.01733	4.54848
0.283119	0.576548	1.13637	1.58577	2.03584	2.48658	3.00944	3.44942	3.98953	4.53456
0.284013	0.568419	1.12385	1.58239	2.03628	2.50644	2.9901	3.4448	3.96375	4.53856
0.284834	0.572532	1.11105	1.58459	2.02943	2.52827	2.96099	3.44411	3.956	4.55639
0.286343	0.586909	1.10578	1.58756	2.0269	2.54057	2.94093	3.43645	3.9569	4.57291
0.286393	0.605786	1.11459	1.58728	2.03743	2.54264	2.92419	3.43357	3.9602	4.57409
0.283744	0.621575	1.12552	1.58962	2.05865	2.54065	2.91912	3.42373	3.95825	4.57327
0.279465	0.629871	1.13167	1.59645	2.08384	2.53502	2.91582	3.40499	3.94512	4.5533
0.27686	0.626153	1.13161	1.60412	2.10901	2.52712	2.90809	3.38624	3.93398	4.52614
0.277355	0.611584	1.12772	1.61341	2.13085	2.51507	2.89589	3.36441	3.93012	4.49649
0.279709	0.596133	1.12411	1.61661	2.14249	2.50123	2.88824	3.33775	3.93363	4.48069
0.281917	0.589358	1.12868	1.61789	2.14512	2.48749	2.88904	3.31911	3.95096	4.48462
0.28198	0.594138	1.14362	1.6153	2.14505	2.47544	2.89872	3.2983	3.95879	4.50666
0.280553	0.608628	1.1595	1.61315	2.14606	2.4669	2.91412	3.28934	3.97363	4.52392
0.27993	0.626647	1.16626	1.60714	2.14419	2.46968	2.92757	3.29521	3.98018	4.53213
0.279338	0.637588	1.16217	1.60282	2.13649	2.48017	2.94005	3.31746	3.9787	4.53609
0.278015	0.638961	1.15484	1.60348	2.12031	2.49017	2.9438	3.35251	3.97233	4.5369
0.278463	0.630889	1.14333	1.60385	2.09103	2.49615	2.94519	3.38627	3.97202	4.54027
0.283417	0.615456	1.13571	1.60142	2.05878	2.50087	2.94132	3.40576	3.97407	4.53619
0.29215	0.598977	1.13322	1.59812	2.03287	2.50513	2.94092	3.40867	3.98752	4.55332
0.30051	0.589956	1.1348	1.59192	2.02165	2.49937	2.95536	3.3978	3.99696	4.5567
0.304452	0.587756	1.13533	1.59057	2.02289	2.48725	2.97527	3.39037	4.01098	4.56522
0.302538	0.585355	1.133	1.59189	2.02031	2.47895	2.98901	3.38916	4.01888	4.55701
0.296704	0.585611	1.13073	1.59687	2.01613	2.48346	2.98975	3.39357	4.01779	4.55477
0.290454	0.593509	1.13479	1.60115	2.01527	2.49337	2.98436	3.3958	4.0071	4.55151
0.28602	0.6096	1.14162	1.60705	2.01734	2.50556	2.97488	3.40183	3.99832	4.55917
0.284644	0.625586	1.14715	1.60915	2.02334	2.51156	2.97173	3.40224	3.98347	4.57143
0.28478	0.635192	1.14464	1.60635	2.03296	2.50654	2.98107	3.4041	3.97072	4.58869
0.282768	0.635552	1.13372	1.60567	2.04356	2.49648	2.99062	3.40281	3.96585	4.59943
0.276264	0.628548	1.12216	1.61109	2.05189	2.48981	2.98992	3.39162	3.95622	4.59599
0.26697	0.621597	1.1182	1.61514	2.06062	2.48985	2.97962	3.38414	3.95378	4.5766

0.259208	0.617916	1.12648	1.61125	2.06778	2.4926	2.96065	3.37329	3.94885	4.55092
0.25805	0.61872	1.1397	1.60149	2.07991	2.49246	2.94422	3.36005	3.94863	4.51608
0.263807	0.62201	1.15439	1.59137	2.09482	2.48543	2.93868	3.35576	3.9541	4.50262
0.272891	0.622824	1.16459	1.588	2.1121	2.47449	2.93281	3.36274	3.96631	4.48743
0.282002	0.619294	1.16668	1.58702	2.12695	2.46945	2.92929	3.38305	3.98103	4.47975
0.291861	0.617863	1.15518	1.59355	2.14005	2.47451	2.92644	3.41093	3.98961	4.47984
0.301081	0.621519	1.14079	1.59831	2.14959	2.48857	2.92823	3.43139	3.99639	4.48263
0.306155	0.626449	1.13188	1.60389	2.15778	2.50462	2.92586	3.4282	3.99398	4.49618
0.307425	0.63293	1.13153	1.60847	2.15902	2.51299	2.92233	3.41337	3.98277	4.5053
0.306794	0.638687	1.13688	1.61851	2.14871	2.5202	2.91401	3.39012	3.98037	4.52292
0.301183	0.641409	1.14931	1.63174	2.12824	2.52744	2.9095	3.36859	3.97077	4.54133
0.289716	0.635398	1.15918	1.64105	2.10543	2.5268	2.90698	3.36855	3.96543	4.55001
0.27389	0.624515	1.16537	1.64277	2.08442	2.52227	2.91843	3.37419	3.95475	4.55392
0.259524	0.614555	1.166	1.63219	2.06777	2.51467	2.93485	3.3793	3.94559	4.53981
0.25166	0.608443	1.16033	1.61532	2.05528	2.50631	2.95378	3.3798	3.93948	4.5344
0.253923	0.606583	1.15457	1.59714	2.04645	2.49757	2.97198	3.37138	3.93719	4.53664
0.265476	0.604656	1.15618	1.5841	2.03734	2.49549	2.98163	3.35586	3.94056	4.54649
0.277504	0.601143	1.16087	1.58456	2.01975	2.49556	2.98411	3.35053	3.95522	4.55257
0.281929	0.602139	1.16124	1.59008	2.00096	2.49633	2.98258	3.36343	3.97193	4.57228
0.278997	0.612432	1.15283	1.59215	1.98586	2.49502	2.97133	3.38822	3.99405	4.576
0.276686	0.630489	1.13818	1.59323	1.9827	2.49716	2.95111	3.39755	3.99865	4.58425
0.277018	0.650422	1.12581	1.59169	1.99373	2.49461	2.93273	3.40147	4.0043	4.57409
0.278541	0.665911	1.11569	1.5921	2.01429	2.49021	2.92019	3.39671	3.99326	4.56638
0.280971	0.66953	1.11237	1.60268	2.03551	2.48711	2.92798	3.40082	3.98432	4.55962
0.283681	0.658525	1.11544	1.61888	2.04948	2.47947	2.93761	3.39167	3.97397	4.5616
0.285645	0.637067	1.12133	1.6309	2.06009	2.47147	2.94507	3.39075	3.97259	4.55467
0.286842	0.614334	1.12227	1.63244	2.0727	2.47191	2.95502	3.38826	3.96752	4.54026
0.288123	0.603156	1.12054	1.62283	2.09021	2.4833	2.96096	3.3831	3.97558	4.53817
0.287597	0.602292	1.11778	1.61043	2.10811	2.49664	2.95841	3.38101	3.97381	4.519
0.285549	0.606026	1.11719	1.60181	2.13001	2.50322	2.95309	3.37451	3.97195	4.50135
0.282644	0.616281	1.12747	1.59022	2.15282	2.50586	2.94423	3.36747	3.97122	4.49247
0.280207	0.630761	1.14341	1.59183	2.17311	2.5091	2.94022	3.35362	3.96718	4.48722
0.275091	0.643988	1.1583	1.6019	2.18055	2.51157	2.93994	3.34238	3.96691	4.49785
0.265616	0.650839	1.16213	1.61787	2.1757	2.50996	2.94376	3.34267	3.96336	4.51033
0.256648	0.647083	1.15968	1.63017	2.15891	2.50231	2.94372	3.36047	3.9595	4.52824
0.252746	0.63117	1.15162	1.63704	2.13291	2.49457	2.94385	3.37677	3.95702	4.547
0.257736	0.610022	1.14421	1.63743	2.10167	2.48447	2.94546	3.40151	3.95202	4.54968
0.270456	0.591015	1.13914	1.63163	2.06661	2.47247	2.95149	3.41627	3.96101	4.5613
0.284673	0.580396	1.14007	1.6155	2.03278	2.46616	2.95614	3.42167	3.97185	4.54833
0.29088	0.582515	1.14414	1.59449	2.0072	2.46666	2.9579	3.41933	3.98318	4.55423
0.285948	0.593589	1.14369	1.57834	1.99625	2.47154	2.95573	3.40934	4.00098	4.56566
0.27374	0.612531	1.1411	1.57545	2.00062	2.478	2.95903	3.41219	4.01109	4.57542
0.26208	0.630046	1.1336	1.5861	2.01737	2.48789	2.95751	3.41949	4.01521	4.58066

0.257864	0.641921	1.12104	1.61001	2.04032	2.49771	2.94445	3.42323	4.00563	4.57734
0.262076	0.649896	1.10986	1.64733	2.05833	2.51208	2.92748	3.4329	3.98304	4.57534
0.270166	0.655401	1.09731	1.6785	2.06669	2.53089	2.91951	3.43118	3.96807	4.56929
0.275478	0.655023	1.09597	1.69051	2.0743	2.53864	2.91669	3.42476	3.94735	4.57108
0.275636	0.648234	1.1058	1.68363	2.08068	2.53473	2.91513	3.4087	3.93511	4.57468
0.274237	0.643478	1.12122	1.66373	2.0835	2.52221	2.92008	3.38665	3.92159	4.56649
0.274198	0.642496	1.13398	1.63329	2.08115	2.50937	2.92811	3.36617	3.91005	4.57266
0.276607	0.644215	1.14156	1.60101	2.07834	2.4988	2.93505	3.34995	3.90637	4.57425
0.277319	0.640733	1.14798	1.57457	2.07814	2.49401	2.93685	3.3488	3.90438	4.56927
0.274041	0.631477	1.15157	1.56515	2.07583	2.48892	2.93198	3.35544	3.91044	4.54186
0.266397	0.618073	1.15387	1.56887	2.07974	2.48199	2.91636	3.36235	3.91145	4.52623
0.252874	0.602695	1.15411	1.58451	2.09593	2.48199	2.90371	3.37205	3.91135	4.49542
0.238583	0.588568	1.15301	1.60573	2.12069	2.48565	2.90009	3.37256	3.92542	4.491
0.232256	0.57921	1.14968	1.621	2.14116	2.49209	2.90826	3.37175	3.94217	4.49431
0.239226	0.576498	1.14165	1.63085	2.1506	2.49445	2.92317	3.36717	3.96628	4.50138
0.25529	0.58558	1.12708	1.63475	2.15294	2.49229	2.93382	3.36629	3.98721	4.51058
0.27085	0.606339	1.11165	1.63297	2.14912	2.48699	2.9417	3.38347	3.99309	4.52208
0.280305	0.627407	1.10216	1.6283	2.13418	2.48203	2.9457	3.40111	3.99004	4.52677
0.282453	0.641635	1.09797	1.61709	2.11547	2.47679	2.95159	3.4106	3.97971	4.52708
0.278507	0.649098	1.09718	1.60635	2.10361	2.47199	2.95875	3.41557	3.9808	4.53015
0.274018	0.651979	1.09799	1.60196	2.09516	2.47428	2.96435	3.41596	3.97845	4.54215
0.271302	0.651315	1.1002	1.60411	2.08477	2.48661	2.96925	3.40683	3.98249	4.56026
0.269602	0.651959	1.10018	1.60735	2.07044	2.50279	2.97233	3.39109	3.9861	4.56977
0.266611	0.650627	1.10619	1.61786	2.0512	2.5155	2.97945	3.3736	3.9969	4.5817
0.263245	0.646633	1.11599	1.63273	2.03001	2.51825	2.98803	3.36001	4.00157	4.57827
0.261466	0.638618	1.129	1.6489	2.01477	2.51536	2.99761	3.3609	4.00826	4.57218
0.26372	0.625705	1.14446	1.65133	2.01095	2.50819	2.99678	3.3691	4.00147	4.58219
0.267483	0.612989	1.15828	1.64478	2.02429	2.50515	2.98778	3.38272	3.99759	4.58198
0.265476	0.602887	1.168	1.64055	2.04281	2.50066	2.9669	3.396	3.98202	4.59617
0.255313	0.598418	1.16854	1.63978	2.05852	2.49477	2.94251	3.41432	3.95497	4.60634
0.2438	0.597593	1.1614	1.64275	2.07021	2.48761	2.92297	3.4255	3.92462	4.59797
0.236785	0.597441	1.15094	1.63995	2.07922	2.48356	2.90966	3.42986	3.89606	4.58555
0.234189	0.597221	1.13388	1.63488	2.08453	2.48694	2.89799	3.43824	3.88806	4.5741
0.240806	0.596189	1.11392	1.62894	2.09352	2.49901	2.88172	3.44917	3.89082	4.56525
0.258611	0.594511	1.09351	1.62033	2.10846	2.51378	2.87528	3.4456	3.90009	4.54741
0.278474	0.598487	1.07807	1.61515	2.12808	2.52265	2.88157	3.42857	3.91683	4.54189
0.289338	0.607416	1.06767	1.61135	2.15037	2.52069	2.88675	3.40284	3.93098	4.53757
0.285752	0.61633	1.07	1.60944	2.17481	2.50906	2.89504	3.37409	3.93605	4.53964
0.269979	0.625845	1.08205	1.60927	2.19522	2.4966	2.90013	3.34946	3.9377	4.54157
0.251901	0.63593	1.10082	1.60911	2.20159	2.49221	2.90654	3.33105	3.93295	4.54799
0.23908	0.641389	1.11539	1.60898	2.18682	2.49894	2.91192	3.32427	3.92809	4.55394
0.234002	0.640047	1.12122	1.60774	2.15682	2.50913	2.91622	3.33486	3.92974	4.56133
0.237761	0.633233	1.12323	1.60882	2.12262	2.51359	2.9262	3.35257	3.93131	4.56458

0.24854	0.623506	1.12644	1.6131	2.09318	2.50857	2.94806	3.36986	3.94228	4.55483
0.260507	0.616007	1.12767	1.62166	2.07087	2.49769	2.96787	3.38971	3.96634	4.53255
0.266674	0.609576	1.12569	1.62914	2.05851	2.48415	2.97756	3.3884	3.9804	4.51506
0.264843	0.601498	1.12173	1.63462	2.05596	2.46824	2.97498	3.38088	3.98984	4.50807
0.254451	0.594651	1.11769	1.63589	2.05584	2.46011	2.96218	3.36204	3.98589	4.51433
0.238683	0.588992	1.11557	1.63206	2.05033	2.4584	2.9488	3.34388	3.98359	4.52542
0.226919	0.582334	1.11721	1.62731	2.03888	2.46165	2.93694	3.34461	3.98195	4.54456
0.2242	0.579295	1.11728	1.62812	2.0293	2.47099	2.92871	3.35736	3.98309	4.57038
0.229848	0.585176	1.12039	1.63652	2.03084	2.47931	2.92518	3.38575	3.99001	4.58301
0.242514	0.595015	1.12736	1.64794	2.04096	2.48305	2.91883	3.41326	3.99691	4.58785
0.256265	0.606102	1.12806	1.66095	2.05426	2.48766	2.90752	3.43271	3.9889	4.58983
0.263916	0.613723	1.11924	1.66675	2.06924	2.49307	2.89783	3.42982	3.95794	4.58486
0.260805	0.617784	1.10923	1.65959	2.08242	2.49798	2.90107	3.4186	3.92567	4.59183
0.250499	0.620628	1.10915	1.64171	2.09229	2.50598	2.92019	3.40579	3.89309	4.60054
0.237619	0.623169	1.11301	1.62855	2.09664	2.51393	2.94645	3.40985	3.87669	4.6141
0.226137	0.62928	1.11583	1.62229	2.09952	2.51808	2.96893	3.42326	3.88348	4.61614
0.220311	0.637066	1.12338	1.61771	2.10989	2.5153	2.97579	3.42774	3.89617	4.61085
0.223434	0.63654	1.12989	1.61331	2.12474	2.50711	2.96824	3.42202	3.9314	4.59345
0.232351	0.623798	1.1388	1.60977	2.14215	2.50242	2.95125	3.40014	3.96325	4.57867
0.239002	0.606352	1.14111	1.60547	2.1568	2.50352	2.93805	3.37771	3.98096	4.56641
0.240032	0.595136	1.13747	1.60509	2.16867	2.50294	2.93206	3.3617	3.99162	4.57266
0.233646	0.59618	1.12763	1.61028	2.1768	2.49722	2.9358	3.36343	3.98903	4.56962
0.223922	0.606362	1.11472	1.61693	2.1779	2.49156	2.938	3.37318	3.99013	4.5693
0.215893	0.615013	1.10164	1.62392	2.16991	2.48731	2.93805	3.38545	3.98231	4.56405
0.215073	0.612867	1.09517	1.63211	2.15253	2.48266	2.93218	3.40238	3.98128	4.55695
0.217612	0.602309	1.10103	1.63442	2.12835	2.48752	2.92562	3.40974	3.97549	4.54677
0.219754	0.589699	1.11168	1.62934	2.10311	2.49801	2.92323	3.40336	3.96774	4.54147
0.220824	0.58515	1.12152	1.62149	2.07525	2.50961	2.9177	3.39771	3.95159	4.53883
0.223016	0.591432	1.12968	1.62217	2.05388	2.51486	2.91043	3.39292	3.94356	4.54979
0.227383	0.607116	1.13414	1.62948	2.04101	2.51069	2.90306	3.39505	3.93121	4.56763
0.234068	0.618143	1.13531	1.63835	2.03603	2.49927	2.8964	3.3929	3.92607	4.58362
0.239986	0.618618	1.1367	1.6386	2.03863	2.48372	2.90043	3.38693	3.926	4.58854
0.235501	0.613686	1.13807	1.62928	2.05002	2.46426	2.91164	3.38134	3.92969	4.59386
0.22133	0.610306	1.13572	1.61778	2.06321	2.44934	2.93225	3.37113	3.93374	4.6031
0.20554	0.611257	1.12984	1.61164	2.06824	2.45066	2.9488	3.36918	3.93769	4.60834
0.194336	0.614552	1.11998	1.61747	2.06493	2.46438	2.95373	3.3633	3.93969	4.60042
0.188903	0.618714	1.11072	1.62997	2.0554	2.48153	2.94904	3.35634	3.93803	4.59881
0.189207	0.624672	1.10589	1.63456	2.04614	2.49358	2.93651	3.356	3.94416	4.5881
0.193458	0.626287	1.10623	1.63076	2.03934	2.49777	2.91997	3.35904	3.94103	4.56867
0.199723	0.620023	1.10688	1.62739	2.04426	2.49916	2.90598	3.37124	3.9298	4.55154
0.204416	0.606917	1.11177	1.63026	2.0658	2.50007	2.90778	3.38612	3.93209	4.54686
0.202771	0.599422	1.12439	1.63258	2.09536	2.50256	2.92171	3.39658	3.93319	4.54393
0.193559	0.600338	1.13918	1.63703	2.11912	2.50855	2.93372	3.40498	3.94534	4.54838

0.183264	0.605182	1.1598	1.64123	2.14044	2.511	2.94101	3.40357	3.9522	4.55001
0.177992	0.607186	1.17197	1.64653	2.16535	2.51255	2.94489	3.40181	3.94601	4.55004
0.178419	0.60616	1.17105	1.64735	2.1871	2.50736	2.94854	3.39782	3.94745	4.55422
0.184988	0.603317	1.16245	1.64185	2.19296	2.50852	2.94738	3.3929	3.93819	4.56851
0.194408	0.595571	1.14907	1.62338	2.18479	2.51089	2.94328	3.39388	3.93257	4.58874
0.202695	0.588835	1.1335	1.60005	2.1639	2.50471	2.94296	3.39408	3.93584	4.60801
0.206809	0.587357	1.11636	1.58067	2.13477	2.49014	2.94999	3.39513	3.94661	4.61563
0.205356	0.593635	1.10538	1.56755	2.10433	2.4764	2.96229	3.40377	3.9592	4.61316
0.197289	0.60337	1.10215	1.56702	2.08558	2.46843	2.96958	3.41354	3.96957	4.6081
0.180684	0.609913	1.10521	1.58048	2.07759	2.46391	2.97412	3.41613	3.96852	4.59115
0.162962	0.611578	1.10629	1.60121	2.07555	2.4622	2.9749	3.41449	3.97709	4.56746
0.152445	0.610856	1.10576	1.62144	2.0797	2.46754	2.97117	3.39425	3.98248	4.54342
0.148315	0.607473	1.10122	1.63396	2.08465	2.47834	2.96351	3.37477	3.97422	4.53592
0.148858	0.605652	1.09857	1.64633	2.07608	2.48625	2.9414	3.36098	3.96215	4.53535
0.153969	0.608221	1.09683	1.65145	2.05762	2.48727	2.92057	3.34956	3.95323	4.54141
0.16348	0.616316	1.10345	1.6513	2.03948	2.48747	2.90251	3.35239	3.94497	4.55291
0.173977	0.627446	1.11957	1.64305	2.02605	2.49073	2.88807	3.36513	3.94489	4.5627
0.179427	0.639538	1.13845	1.63056	2.02352	2.49444	2.88157	3.38468	3.94811	4.57999
0.177095	0.645445	1.15295	1.61737	2.03387	2.49601	2.88633	3.39983	3.95911	4.59555
0.169669	0.639902	1.16124	1.60694	2.05175	2.49827	2.89575	3.40141	3.96686	4.61349
0.162459	0.625985	1.15611	1.60362	2.06665	2.49953	2.90589	3.39283	3.98117	4.62197
0.157567	0.608449	1.14762	1.60071	2.07491	2.50273	2.9122	3.38227	3.98631	4.61058
0.155967	0.592449	1.14409	1.59818	2.08188	2.51048	2.92123	3.36845	3.99341	4.5892
0.158551	0.578152	1.14658	1.59605	2.09361	2.51781	2.93005	3.36055	3.99298	4.57007
0.161448	0.568684	1.15024	1.59607	2.10618	2.52229	2.93939	3.36618	3.98964	4.55816
0.157015	0.565234	1.15242	1.60194	2.12023	2.52769	2.94762	3.37631	3.98402	4.5628
0.145075	0.564812	1.14902	1.61247	2.13507	2.52928	2.9503	3.39013	3.97872	4.56604
0.132913	0.568589	1.13937	1.62306	2.14756	2.52052	2.94725	3.39834	3.96882	4.57438
0.127331	0.576515	1.1228	1.6343	2.15074	2.50106	2.94335	3.41209	3.95319	4.57279
0.132437	0.590127	1.1073	1.64331	2.14886	2.47883	2.93738	3.4242	3.94105	4.56813
0.143843	0.608254	1.09606	1.64516	2.14188	2.46006	2.93243	3.4345	3.92918	4.57508
0.154027	0.625777	1.09065	1.64031	2.13833	2.45114	2.92906	3.43792	3.92309	4.57205
0.15694	0.635858	1.09349	1.62991	2.136	2.45135	2.93607	3.43812	3.91929	4.58468
0.151482	0.640368	1.10411	1.62173	2.1306	2.46113	2.94185	3.43099	3.91469	4.58422
0.141934	0.641301	1.1185	1.61466	2.11215	2.48001	2.93712	3.41305	3.90926	4.59158
0.131982	0.638579	1.13338	1.60711	2.08422	2.49761	2.93093	3.40026	3.89748	4.58237
0.124189	0.630409	1.14186	1.59734	2.05174	2.50539	2.92731	3.37938	3.90777	4.58079
0.123141	0.620347	1.14866	1.59029	2.02957	2.50295	2.93095	3.35873	3.92107	4.56556
0.128473	0.612029	1.15688	1.57978	2.01993	2.49846	2.93774	3.34887	3.94981	4.5548
0.134	0.605416	1.15283	1.5662	2.01545	2.49258	2.9458	3.3432	3.97197	4.54786
0.138742	0.595512	1.14274	1.56317	2.0147	2.48253	2.94364	3.34194	3.99271	4.54427
0.141866	0.58275	1.1285	1.57113	2.01971	2.47293	2.94412	3.34362	4.01142	4.55204
0.140965	0.573272	1.1147	1.58952	2.02489	2.47036	2.94184	3.35046	4.0132	4.56691

0.137452	0.575105	1.1048	1.6053	2.02846	2.48041	2.93996	3.36639	4.01226	4.58249
0.130093	0.585038	1.10097	1.61424	2.03821	2.49805	2.93937	3.38385	4.00409	4.58905
0.125109	0.599608	1.10469	1.61751	2.05206	2.51095	2.94223	3.39704	3.98989	4.58799
0.127757	0.609622	1.11719	1.61954	2.07226	2.51699	2.9476	3.40681	3.98172	4.57926
0.136026	0.612144	1.13323	1.62472	2.0967	2.5151	2.95898	3.40393	3.98213	4.57452
0.14395	0.611159	1.14514	1.63187	2.12127	2.50989	2.96477	3.40255	3.99266	4.5758
0.14867	0.609214	1.15709	1.63703	2.14271	2.50775	2.96333	3.40757	4.00032	4.57257
0.14802	0.611231	1.1628	1.63602	2.15764	2.51355	2.954	3.41615	4.00217	4.5759
0.140995	0.61486	1.16129	1.62393	2.16321	2.51744	2.94043	3.42231	3.99375	4.58004
0.128986	0.617216	1.14631	1.60786	2.15256	2.51711	2.92796	3.42392	3.98595	4.57576
0.116622	0.617689	1.12871	1.60044	2.13057	2.5069	2.92253	3.42095	3.97274	4.58425
0.109375	0.616914	1.1212	1.59726	2.10733	2.49249	2.92197	3.42413	3.97731	4.59641
0.107848	0.615887	1.12397	1.59301	2.09044	2.48028	2.92412	3.42081	3.97983	4.6063
0.111353	0.613123	1.13235	1.58699	2.084	2.47733	2.92206	3.41285	3.98277	4.61229
0.119145	0.609328	1.13791	1.58435	2.08262	2.48183	2.91484	3.40327	3.97858	4.60803
0.125845	0.608442	1.14046	1.58712	2.08175	2.48094	2.91223	3.39599	3.96397	4.60037
0.128982	0.609209	1.1397	1.59758	2.07372	2.47304	2.9139	3.39392	3.94583	4.58348
0.130508	0.608445	1.13709	1.61555	2.05603	2.45965	2.92783	3.3914	3.93345	4.56133
0.13209	0.605456	1.13412	1.63059	2.03588	2.45542	2.94859	3.38774	3.92728	4.5455
0.135156	0.602329	1.13464	1.63736	2.02269	2.46861	2.97367	3.39376	3.9357	4.54175
0.13824	0.60207	1.14272	1.63698	2.01389	2.49307	2.99091	3.39166	3.94409	4.54175
0.133533	0.603805	1.15291	1.62753	2.01166	2.51876	2.98806	3.37285	3.95573	4.53886
0.120272	0.606242	1.15788	1.61361	2.01376	2.53215	2.97171	3.3556	3.95533	4.53658
0.107831	0.61012	1.15591	1.60192	2.02311	2.53119	2.95027	3.34247	3.95492	4.55009
0.104142	0.615587	1.1461	1.60474	2.03766	2.51901	2.93003	3.34466	3.95629	4.55078
0.110695	0.620013	1.13095	1.60928	2.05142	2.50767	2.91201	3.36369	3.95412	4.56398
0.124626	0.618895	1.1138	1.60812	2.06102	2.49877	2.91056	3.38376	3.96351	4.57087
0.137677	0.612875	1.10037	1.5946	2.06756	2.49844	2.92255	3.39259	3.97836	4.57652
0.139786	0.604147	1.09387	1.58041	2.07763	2.49887	2.93831	3.40053	3.99897	4.58227
0.129674	0.596829	1.09826	1.57372	2.09539	2.4991	2.95367	3.40509	4.01128	4.56143
0.117404	0.593588	1.10924	1.57453	2.1123	2.49948	2.96843	3.40509	4.01271	4.55246
0.113862	0.595393	1.12128	1.5861	2.12305	2.49768	2.98161	3.40146	4.01371	4.53856
0.122918	0.602023	1.13144	1.60604	2.1323	2.49673	2.98429	3.39769	4.01303	4.54115
0.136387	0.612244	1.14282	1.63271	2.13933	2.49576	2.97064	3.40328	4.01132	4.55169
0.140132	0.622281	1.1507	1.6564	2.14151	2.49841	2.95067	3.4064	4.00626	4.57073
0.132731	0.628406	1.15138	1.66774	2.13348	2.50345	2.93201	3.42017	4.00445	4.59272
0.122785	0.633436	1.15544	1.66367	2.115	2.50302	2.92363	3.42914	3.99988	4.59239
0.118632	0.635273	1.16966	1.64292	2.09169	2.50103	2.92604	3.43733	3.9916	4.58494
0.12238	0.633604	1.18656	1.61714	2.07803	2.49305	2.92454	3.44644	3.9816	4.58647
0.132889	0.627176	1.18868	1.59992	2.0764	2.48394	2.92995	3.45386	3.974	4.57852
0.146233	0.616689	1.17597	1.59268	2.0819	2.47729	2.94756	3.44902	3.97095	4.58586
0.152701	0.606921	1.15677	1.59495	2.08997	2.47424	2.96531	3.43587	3.97133	4.58697
0.147887	0.606499	1.1395	1.60515	2.0978	2.47836	2.97681	3.40781	3.96662	4.5776

0.138657	0.612449	1.12998	1.61224	2.09355	2.48671	2.97942	3.37453	3.96446	4.5732
0.134463	0.619878	1.13397	1.61575	2.072	2.50022	2.97371	3.34293	3.96446	4.56252
0.135935	0.625404	1.14669	1.61276	2.04007	2.51845	2.9607	3.32164	3.96498	4.5457
0.138229	0.62383	1.15653	1.60767	2.01089	2.53415	2.9466	3.31891	3.97108	4.54299
0.136798	0.614623	1.15657	1.60996	1.9928	2.53946	2.93698	3.32286	3.99306	4.54101
0.132515	0.602879	1.14461	1.61187	1.98764	2.53079	2.92472	3.34344	4.00338	4.55192
0.12988	0.594542	1.12657	1.61621	1.99295	2.51316	2.91286	3.36216	4.0059	4.55785
0.132614	0.5942	1.11181	1.62141	2.00316	2.49478	2.91074	3.37077	3.99181	4.56857
0.138534	0.599448	1.10363	1.63197	2.01656	2.47765	2.91214	3.37457	3.97816	4.56795
0.145467	0.605736	1.10721	1.64153	2.04135	2.46722	2.92079	3.38354	3.97054	4.56037
0.151159	0.61431	1.11561	1.65033	2.07341	2.46847	2.93658	3.39678	3.96731	4.54935
0.154511	0.622271	1.12127	1.65282	2.09713	2.48806	2.95847	3.408	3.95973	4.53991
0.153243	0.630268	1.12933	1.64731	2.11834	2.5126	2.98058	3.42077	3.96203	4.54505
0.15086	0.638107	1.13481	1.63711	2.14022	2.53038	2.99565	3.42615	3.97571	4.55104
0.150641	0.643067	1.13955	1.61982	2.15756	2.53911	2.99321	3.42977	3.97605	4.56875
0.152056	0.643966	1.14733	1.60175	2.16618	2.5383	2.97572	3.43757	3.98107	4.57886
0.150857	0.642508	1.15633	1.58845	2.17095	2.53095	2.96276	3.43681	3.98402	4.5812
0.147545	0.639688	1.16651	1.58232	2.16994	2.51691	2.9557	3.44021	3.9847	4.57276
0.144722	0.635174	1.17327	1.57622	2.15902	2.49725	2.95071	3.4372	3.99039	4.55137
0.141439	0.627738	1.16744	1.57186	2.13928	2.48283	2.9537	3.42942	3.98798	4.53335
0.138935	0.620431	1.15682	1.57493	2.11634	2.47385	2.95801	3.42004	3.9832	4.52005
0.142451	0.612978	1.1464	1.59	2.09967	2.47051	2.95943	3.4115	3.98229	4.53028
0.155757	0.60538	1.1391	1.61067	2.08922	2.47023	2.95414	3.41035	3.98898	4.53992
0.172836	0.597929	1.13174	1.62785	2.07843	2.46757	2.94928	3.40696	4.00303	4.55894
0.184187	0.591629	1.12649	1.6386	2.06963	2.46287	2.95313	3.40483	3.99959	4.55971
0.185211	0.594281	1.12787	1.64717	2.05871	2.46248	2.96661	3.38307	3.99651	4.55116
0.174433	0.604715	1.13762	1.64357	2.04438	2.467	2.97358	3.3679	3.99107	4.53638
0.157841	0.612001	1.15131	1.63367	2.03529	2.47532	2.97058	3.35227	3.98706	4.53274
0.145824	0.609703	1.16347	1.62172	2.03004	2.49046	2.95351	3.33991	3.98963	4.52714
0.148238	0.601802	1.1702	1.61541	2.02368	2.50883	2.93481	3.32791	3.99438	4.51521
0.164942	0.59414	1.16747	1.61623	2.0203	2.52728	2.92559	3.32418	3.98955	4.52069
0.187432	0.594489	1.15847	1.62165	2.02224	2.54952	2.92763	3.32396	3.9905	4.53073
0.204707	0.606169	1.14492	1.62538	2.03165	2.55807	2.94097	3.32666	3.98983	4.54649
0.212024	0.626019	1.13644	1.62413	2.04463	2.55658	2.9653	3.32719	3.98686	4.56372
0.20758	0.645832	1.13323	1.61378	2.06022	2.54509	2.9831	3.3372	3.97953	4.56885
0.193208	0.654864	1.13327	1.60657	2.0781	2.52656	2.99272	3.3583	3.97165	4.56183
0.175738	0.650926	1.13048	1.60343	2.09388	2.50644	2.98609	3.38505	3.96813	4.57522
0.160247	0.642029	1.1251	1.60759	2.10678	2.49068	2.9739	3.41404	3.96802	4.56258
0.152033	0.634239	1.12336	1.61832	2.11774	2.48033	2.96865	3.44	3.97901	4.55756
0.153483	0.630089	1.12781	1.63029	2.13018	2.47398	2.9636	3.45874	3.99206	4.54609
0.16212	0.62606	1.13075	1.63938	2.14205	2.47108	2.9592	3.46423	3.99786	4.55377
0.17608	0.617382	1.1293	1.63753	2.15167	2.46776	2.94912	3.46537	3.9956	4.56676
0.194174	0.60509	1.1306	1.62772	2.15789	2.465	2.93784	3.45519	3.99049	4.58665

0.210436	0.596215	1.13641	1.61652	2.15724	2.47013	2.93529	3.4355	3.98432	4.59136
0.220536	0.594232	1.14352	1.60903	2.15015	2.48078	2.93979	3.41516	3.97824	4.59242
0.223011	0.599531	1.15551	1.60789	2.14191	2.4969	2.94382	3.40214	3.98118	4.57327
0.217801	0.609969	1.16456	1.5964	2.1301	2.52103	2.9459	3.38913	3.98013	4.54622
0.208457	0.623606	1.16481	1.57943	2.11718	2.54091	2.94586	3.38544	3.98305	4.5108
0.199873	0.631785	1.16111	1.57086	2.10445	2.5447	2.95143	3.38804	3.97965	4.48966
0.19886	0.630643	1.1516	1.57589	2.09037	2.52648	2.95664	3.38886	3.97499	4.48367
0.20823	0.623686	1.14531	1.58519	2.07704	2.50355	2.95705	3.38214	3.9792	4.48248
0.22283	0.616002	1.14295	1.59549	2.0662	2.48602	2.95441	3.37643	3.98179	4.49929
0.232008	0.608028	1.14518	1.60496	2.05434	2.47164	2.94899	3.37023	3.99482	4.51177
0.228853	0.60045	1.15002	1.61139	2.03934	2.46577	2.9407	3.37261	4.00685	4.53354
0.214532	0.594556	1.15527	1.61383	2.03044	2.47157	2.94086	3.38313	4.01106	4.54249
0.197149	0.591454	1.15561	1.61674	2.02857	2.48319	2.94171	3.39317	4.01648	4.54994
0.186614	0.592781	1.14936	1.62325	2.03413	2.49355	2.95289	3.39267	4.00796	4.56296
0.190875	0.600609	1.14117	1.62944	2.04323	2.50386	2.96565	3.38527	3.99427	4.5768
0.205969	0.609571	1.13636	1.63372	2.05138	2.51038	2.97594	3.37902	3.97756	4.58457
0.220205	0.619782	1.13937	1.6355	2.0567	2.51246	2.98456	3.37325	3.96477	4.58074
0.22849	0.627933	1.14528	1.63175	2.0625	2.51474	2.98272	3.37166	3.95604	4.58701
0.232536	0.637737	1.14508	1.6275	2.07337	2.51942	2.97974	3.36919	3.9506	4.58658
0.236218	0.642258	1.14039	1.61553	2.09348	2.52119	2.9788	3.36854	3.95503	4.58429
0.240825	0.640136	1.13715	1.60158	2.11996	2.51676	2.9725	3.36855	3.96865	4.58173
0.24656	0.632066	1.13543	1.58822	2.1414	2.51483	2.97118	3.36428	3.9891	4.56985
0.252099	0.621724	1.13442	1.57766	2.15617	2.518	2.97245	3.36696	4.01153	4.55654
0.253659	0.614764	1.13394	1.56816	2.1712	2.52116	2.96712	3.3794	4.0257	4.54033
0.250742	0.61151	1.13797	1.56575	2.18535	2.51809	2.95716	3.39242	4.02194	4.52945
0.245747	0.608222	1.14484	1.56704	2.18674	2.50566	2.94741	3.39984	4.00022	4.5104
0.241311	0.60278	1.14306	1.56954	2.17119	2.48382	2.94454	3.39706	3.97795	4.49685
0.237411	0.59572	1.13097	1.57631	2.14574	2.46357	2.95274	3.39184	3.95746	4.4969
0.231414	0.584758	1.11466	1.58453	2.11261	2.45227	2.96848	3.38321	3.95249	4.49394
0.225848	0.575599	1.10143	1.59209	2.08605	2.44841	2.98252	3.38451	3.95059	4.50907
0.224832	0.576817	1.096	1.59927	2.07386	2.4555	2.986	3.39107	3.95304	4.51711
0.230493	0.587899	1.10209	1.60287	2.07196	2.46965	2.96969	3.40542	3.95607	4.53086
0.239365	0.602405	1.11716	1.60612	2.07308	2.47844	2.94685	3.42492	3.95035	4.53939
0.246769	0.611591	1.13934	1.61026	2.07034	2.47941	2.92518	3.4366	3.96142	4.54558
0.251853	0.616598	1.15874	1.61439	2.06167	2.48373	2.90542	3.43689	3.96866	4.54095
0.253803	0.621293	1.16755	1.61584	2.04936	2.49879	2.89407	3.43168	3.98109	4.53989
0.25217	0.624368	1.16381	1.61504	2.04208	2.51895	2.8955	3.43306	3.99073	4.54416
0.250934	0.622549	1.15689	1.60867	2.04363	2.53526	2.91187	3.42667	3.99498	4.55177
0.254037	0.618935	1.15228	1.60018	2.0546	2.54347	2.93776	3.40549	3.99286	4.54704
0.262302	0.615718	1.153	1.59782	2.07314	2.5485	2.96034	3.3795	3.98347	4.54449
0.267688	0.612186	1.15328	1.59902	2.09327	2.55851	2.97478	3.35555	3.96554	4.53164
0.266016	0.607494	1.14797	1.59455	2.10869	2.56011	2.9848	3.3401	3.95104	4.52738
0.260408	0.60369	1.14088	1.58986	2.1172	2.54612	2.99507	3.34236	3.93208	4.52925

0.257934	0.60507	1.13502	1.58629	2.11956	2.52555	2.99977	3.36165	3.92605	4.53061
0.259344	0.608635	1.12962	1.57985	2.11554	2.50023	3.00187	3.38961	3.93035	4.53636
0.261684	0.610291	1.12575	1.57641	2.11099	2.47454	2.99743	3.41239	3.95565	4.54038
0.263817	0.605297	1.12469	1.56975	2.11331	2.4521	2.99101	3.41803	3.9928	4.551
0.264852	0.593735	1.13148	1.56589	2.12352	2.43973	2.97741	3.40405	4.02037	4.56455
0.263871	0.580162	1.13603	1.56978	2.1369	2.44232	2.96943	3.38737	4.03472	4.5749
0.257718	0.574635	1.13608	1.57875	2.1517	2.44845	2.95479	3.3785	4.03433	4.58274
0.247225	0.579171	1.13175	1.59014	2.16278	2.45655	2.94353	3.37867	4.018	4.57384
0.240059	0.593777	1.12448	1.60155	2.16724	2.46427	2.92678	3.38694	3.99408	4.56837
0.238635	0.611743	1.1141	1.6114	2.16304	2.47916	2.91442	3.40174	3.97402	4.55784
0.239709	0.627009	1.10821	1.61108	2.15125	2.50564	2.90998	3.41167	3.96774	4.54523
0.240829	0.633761	1.1101	1.60318	2.12959	2.53439	2.91975	3.41632	3.97777	4.53755
0.244359	0.62949	1.12019	1.58814	2.10468	2.55486	2.9331	3.42261	3.9929	4.53985
0.254503	0.617764	1.13389	1.58136	2.08591	2.55695	2.95253	3.41634	4.00306	4.53547
0.273507	0.604646	1.14882	1.58212	2.07732	2.53895	2.96937	3.41813	4.01294	4.52093
0.293017	0.592997	1.15693	1.59188	2.08051	2.50449	2.9821	3.4145	4.00451	4.51905
0.300176	0.583563	1.16193	1.60123	2.08961	2.46945	2.98542	3.40402	3.98556	4.51126
0.290534	0.575994	1.16475	1.60193	2.0922	2.44598	2.98304	3.39124	3.96318	4.52938
0.271721	0.568727	1.16219	1.59988	2.08819	2.43973	2.97798	3.37788	3.94542	4.54051
0.256293	0.564621	1.15825	1.59777	2.07999	2.44187	2.97849	3.36063	3.92831	4.54432
0.250319	0.566874	1.15919	1.59404	2.06804	2.45697	2.97728	3.34933	3.91456	4.54136
0.251534	0.579679	1.16172	1.59245	2.05993	2.47835	2.97323	3.35214	3.9085	4.53447
0.25706	0.595342	1.16287	1.5901	2.06142	2.50394	2.97517	3.36801	3.90638	4.53551
0.262156	0.609868	1.15287	1.5903	2.07146	2.52811	2.976	3.39043	3.90319	4.52989
0.262314	0.616321	1.13088	1.58719	2.0865	2.5485	2.96352	3.41402	3.91404	4.53573
0.256594	0.617711	1.10888	1.5896	2.10276	2.55662	2.93985	3.42687	3.92632	4.53604
0.251412	0.617659	1.10016	1.59111	2.12208	2.55077	2.9224	3.42305	3.93974	4.5376
0.249661	0.617474	1.11249	1.59478	2.14425	2.53362	2.91411	3.40279	3.94261	4.538
0.251297	0.621801	1.13561	1.59571	2.16617	2.51367	2.91143	3.37385	3.95699	4.53293
0.255552	0.632608	1.15787	1.59103	2.1823	2.49347	2.91286	3.35542	3.97712	4.52651
0.262568	0.639799	1.16278	1.58969	2.18981	2.47562	2.91952	3.3539	4.00158	4.51768
0.272201	0.629501	1.15401	1.58478	2.18918	2.46362	2.92543	3.36828	4.02401	4.50658
0.280269	0.603944	1.14254	1.57885	2.18013	2.45718	2.92442	3.38583	4.04079	4.51029
0.285762	0.575207	1.13189	1.57254	2.15915	2.45499	2.9223	3.41009	4.04614	4.51668
0.290626	0.558113	1.12165	1.56767	2.12854	2.45605	2.92882	3.42927	4.03879	4.52114
0.293683	0.554594	1.11565	1.56645	2.10175	2.46114	2.94358	3.44206	4.01893	4.53561
0.292769	0.560725	1.11908	1.56669	2.08741	2.47311	2.96206	3.44325	4.00689	4.54839
0.286814	0.572378	1.12976	1.57383	2.08531	2.49049	2.97412	3.43696	3.98031	4.56085
0.2804	0.584345	1.14431	1.58453	2.09188	2.5049	2.98114	3.42721	3.9653	4.56487
0.273628	0.597251	1.15106	1.5974	2.09999	2.50894	2.98012	3.41714	3.95338	4.573
0.269399	0.60826	1.1445	1.61261	2.10222	2.50526	2.96977	3.39735	3.95196	4.57628
0.270986	0.617634	1.12894	1.62564	2.09769	2.49937	2.96492	3.38296	3.96142	4.58225
0.275425	0.62255	1.11124	1.62926	2.08543	2.49607	2.9637	3.38123	3.96709	4.59035

0.275637	0.625776	1.10218	1.62457	2.06747	2.5001	2.96709	3.3806	3.97752	4.60261
0.266899	0.629565	1.10862	1.60825	2.05376	2.50986	2.96679	3.38246	3.98127	4.605
0.253119	0.632532	1.11546	1.58747	2.05163	2.52224	2.96203	3.37853	3.98113	4.59166
0.243154	0.633861	1.12128	1.57293	2.05903	2.53104	2.95762	3.37193	3.96966	4.57757
0.243267	0.635	1.12225	1.56847	2.0725	2.5347	2.95459	3.36439	3.96019	4.57514
0.253352	0.636452	1.12957	1.56852	2.08592	2.52643	2.96036	3.37234	3.95189	4.55884
0.266455	0.634191	1.14376	1.5751	2.09549	2.50942	2.96774	3.39352	3.94649	4.56576
0.278075	0.626651	1.15828	1.57923	2.09567	2.49084	2.97792	3.40625	3.94632	4.56649
0.285857	0.618071	1.16615	1.57028	2.09778	2.47174	2.979	3.4119	3.95735	4.57561
0.287498	0.606409	1.16846	1.55728	2.10648	2.45751	2.96896	3.40624	3.96605	4.58569
0.285056	0.594321	1.1642	1.55253	2.12223	2.45114	2.95749	3.39237	3.98369	4.57835
0.282796	0.584271	1.15877	1.55927	2.14243	2.45065	2.94743	3.38109	3.98156	4.57463
0.284208	0.582769	1.1484	1.58031	2.16046	2.45681	2.94193	3.37832	3.97513	4.57555
0.28729	0.591765	1.13607	1.60413	2.17334	2.46746	2.93553	3.3866	3.96121	4.56194
0.289772	0.60631	1.12669	1.6207	2.17902	2.47625	2.92093	3.40312	3.93972	4.55332
0.293433	0.62243	1.12209	1.62646	2.17662	2.48058	2.90447	3.40845	3.92644	4.52444
0.298854	0.632416	1.11971	1.62655	2.16481	2.48029	2.88653	3.41134	3.93013	4.49743
0.301801	0.634231	1.11317	1.62473	2.14508	2.4814	2.87939	3.40214	3.93842	4.49034
0.298385	0.62789	1.10509	1.62275	2.12838	2.49381	2.88378	3.39486	3.94543	4.49123
0.291076	0.615792	1.09886	1.61777	2.11563	2.50954	2.90344	3.38408	3.95141	4.50265
0.28115	0.606993	1.09699	1.61041	2.10555	2.52167	2.9302	3.38009	3.96696	4.53097
0.268042	0.605959	1.09957	1.60122	2.09663	2.53015	2.95753	3.38796	3.97208	4.55295
0.249901	0.614678	1.10873	1.59065	2.08973	2.537	2.97862	3.39697	3.96453	4.57764
0.234966	0.629368	1.11567	1.57811	2.07771	2.53879	2.99088	3.40629	3.96294	4.58876
0.232115	0.642583	1.11665	1.57047	2.06363	2.52762	2.99477	3.40772	3.95903	4.58739
0.24359	0.644612	1.11688	1.56633	2.05307	2.51408	2.99092	3.395	3.95649	4.58565
0.263518	0.637911	1.11895	1.56542	2.04453	2.50053	2.97824	3.3757	3.95766	4.58397
0.281153	0.629316	1.12289	1.57341	2.03498	2.49127	2.96368	3.35796	3.95954	4.58297
0.289609	0.626822	1.12581	1.58776	2.02901	2.48656	2.95211	3.35256	3.96377	4.58837
0.290902	0.631037	1.13028	1.60079	2.02928	2.48388	2.94106	3.37117	3.9669	4.59592
0.289696	0.637131	1.13739	1.60715	2.03666	2.48591	2.93553	3.39658	3.96607	4.60772
0.287782	0.643112	1.13926	1.60346	2.04807	2.48733	2.93627	3.4115	3.96154	4.61473
0.287258	0.645061	1.13836	1.6005	2.06309	2.49122	2.9365	3.41708	3.95935	4.61534
0.286654	0.639312	1.14558	1.59742	2.0842	2.49252	2.93922	3.40589	3.96226	4.62135
0.286539	0.632248	1.15629	1.59827	2.11496	2.49137	2.93452	3.38823	3.97081	4.61632
0.286148	0.628371	1.16251	1.59753	2.14807	2.48675	2.93419	3.37262	3.97193	4.61374
0.284097	0.624537	1.15562	1.59737	2.17173	2.48338	2.93814	3.36762	3.96424	4.60862
0.28222	0.616037	1.13425	1.59524	2.1872	2.48076	2.94683	3.37285	3.95195	4.61157
0.28273	0.601945	1.10786	1.5905	2.19205	2.47077	2.95138	3.38535	3.9456	4.61469
0.286246	0.592924	1.09215	1.59262	2.18325	2.45822	2.95783	3.39916	3.94064	4.60673
0.289496	0.591182	1.0938	1.59999	2.16136	2.45858	2.96099	3.40656	3.93986	4.58408
0.284111	0.59687	1.10791	1.60645	2.1372	2.47781	2.9649	3.40637	3.94369	4.55119
0.270938	0.608771	1.12638	1.60999	2.10893	2.50219	2.9676	3.40575	3.95055	4.52438

0.257871	0.623628	1.1443	1.6122	2.08238	2.51634	2.966	3.40461	3.96465	4.51167
0.252278	0.63502	1.15495	1.61517	2.07028	2.52305	2.96763	3.39985	3.98019	4.51921
0.257341	0.635555	1.1509	1.61361	2.07158	2.52454	2.96121	3.38861	3.98848	4.53385
0.271994	0.630538	1.13197	1.61105	2.07829	2.52734	2.94975	3.37652	3.99989	4.54322
0.291638	0.62745	1.10717	1.6084	2.0867	2.53167	2.93252	3.36505	3.99788	4.55707
0.308696	0.63054	1.09121	1.60845	2.09003	2.53417	2.91391	3.35862	3.98344	4.55933
0.312178	0.636071	1.09098	1.61008	2.08612	2.5365	2.89903	3.3604	3.97079	4.56435
0.296821	0.640424	1.10343	1.61352	2.07582	2.53588	2.89319	3.36335	3.95348	4.57667
0.271455	0.640007	1.1235	1.60946	2.05735	2.53453	2.89071	3.36801	3.93093	4.58966
0.25311	0.638563	1.13822	1.60694	2.03145	2.53122	2.89891	3.38051	3.91497	4.59865
0.253452	0.632643	1.14584	1.60485	2.00776	2.52664	2.913	3.38183	3.90082	4.60179
0.27269	0.624917	1.14296	1.60601	1.99369	2.52179	2.9346	3.38256	3.89542	4.61257
0.300536	0.619247	1.13662	1.60845	1.99505	2.51639	2.9469	3.38595	3.8967	4.62578
0.322165	0.616781	1.13239	1.60631	2.00733	2.50619	2.96106	3.40049	3.90177	4.63354
0.325893	0.612864	1.13117	1.60451	2.02574	2.49516	2.97822	3.41417	3.90535	4.62799
0.311684	0.61226	1.13051	1.60489	2.05056	2.4869	2.99243	3.41405	3.91789	4.62842
0.288503	0.613372	1.13152	1.60627	2.08006	2.48026	2.99673	3.40904	3.92673	4.62038
0.267447	0.612728	1.13356	1.60989	2.11036	2.47174	2.98611	3.39271	3.94446	4.61288
0.255199	0.610782	1.13705	1.61386	2.14125	2.46452	2.96533	3.37471	3.95932	4.59365
0.257051	0.611113	1.1407	1.61848	2.17295	2.46056	2.93684	3.3661	3.98103	4.59129
0.269448	0.611985	1.14022	1.62344	2.20168	2.45946	2.91338	3.36393	3.98507	4.58074
0.283687	0.614671	1.13249	1.62847	2.21624	2.46302	2.90543	3.376	3.98809	4.57052
0.295125	0.620345	1.12203	1.63246	2.20884	2.47525	2.91078	3.39818	3.97487	4.55615
0.302968	0.630229	1.11466	1.63398	2.17897	2.49205	2.9278	3.41227	3.95664	4.53939
0.304833	0.636808	1.1138	1.62889	2.136	2.50978	2.94603	3.41831	3.9411	4.52658
0.298894	0.63487	1.12615	1.62434	2.09821	2.51867	2.95475	3.42004	3.94717	4.52675
0.288381	0.627098	1.14223	1.62772	2.07713	2.52069	2.95866	3.4119	3.95432	4.54204
0.276948	0.620359	1.1542	1.636	2.07056	2.52308	2.95558	3.40337	3.97467	4.55446
0.269135	0.619336	1.15297	1.64697	2.07283	2.52962	2.95564	3.39879	3.98151	4.57009
0.269862	0.624481	1.14321	1.65271	2.0682	2.54143	2.9542	3.39575	3.98526	4.59587
0.277056	0.636305	1.13325	1.65015	2.05274	2.55385	2.95416	3.39242	3.98921	4.62294
0.285071	0.647755	1.12512	1.63479	2.02985	2.56332	2.94858	3.38437	3.98832	4.63836
0.292498	0.648967	1.12474	1.61717	2.00508	2.56221	2.94367	3.37107	3.98371	4.64675
0.297947	0.636789	1.12851	1.60262	1.99196	2.55072	2.93681	3.36144	3.97816	4.65066
0.299554	0.617739	1.131	1.60483	1.99116	2.5261	2.93948	3.35991	3.97368	4.6426
0.296939	0.599144	1.1247	1.61031	2.00179	2.49599	2.94614	3.3575	3.97057	4.63632
0.29223	0.588389	1.1198	1.62135	2.01809	2.47194	2.94707	3.36036	3.96575	4.6388
0.287289	0.589366	1.11514	1.62594	2.03434	2.45777	2.94599	3.37108	3.96344	4.63317
0.282119	0.596641	1.11523	1.62932	2.05359	2.4559	2.94045	3.37569	3.95846	4.62907
0.277189	0.607131	1.11988	1.6336	2.07504	2.46765	2.92956	3.37618	3.95395	4.62201
0.273868	0.619201	1.12893	1.64115	2.09877	2.48615	2.92222	3.37607	3.94189	4.60504
0.271818	0.628838	1.13018	1.65049	2.12648	2.50379	2.91772	3.37607	3.93221	4.60344
0.269154	0.630608	1.12969	1.65376	2.15305	2.51063	2.92154	3.3819	3.92249	4.591

0.268417	0.624339	1.12897	1.6546	2.17121	2.5071	2.93411	3.39254	3.91525	4.57485
0.273436	0.616606	1.12436	1.65307	2.17699	2.50372	2.95575	3.39682	3.91429	4.5705
0.282105	0.606791	1.12355	1.64963	2.17755	2.50081	2.97048	3.39531	3.91686	4.56433
0.290109	0.59936	1.12367	1.64366	2.17162	2.50197	2.96824	3.39785	3.91613	4.57065
0.294656	0.593843	1.12747	1.64603	2.15708	2.51031	2.95345	3.4005	3.91834	4.58187
0.293299	0.590884	1.13126	1.6494	2.13463	2.52321	2.93698	3.40207	3.92124	4.58595
0.285876	0.590669	1.14023	1.64899	2.11058	2.53394	2.92417	3.40059	3.93204	4.58551
0.276742	0.59238	1.15537	1.65201	2.09199	2.53826	2.91752	3.39923	3.94342	4.58558
0.271021	0.599405	1.16998	1.64489	2.0775	2.53689	2.92518	3.39518	3.96463	4.58433
0.270035	0.612271	1.17443	1.63407	2.06936	2.53632	2.93829	3.38673	3.98368	4.5915
0.272423	0.625175	1.16542	1.6245	2.068	2.53124	2.94314	3.38252	4.00662	4.59948
0.276876	0.631854	1.14664	1.62123	2.06682	2.52035	2.94111	3.38025	4.0195	4.61135
0.280798	0.62973	1.12392	1.62463	2.06195	2.51033	2.94304	3.38878	4.01873	4.61652
0.279339	0.618913	1.10472	1.63091	2.05294	2.50816	2.94902	3.40729	4.01662	4.6123
0.272979	0.603827	1.09749	1.64149	2.04214	2.50954	2.95634	3.41144	4.00747	4.60511
0.267806	0.591263	1.10362	1.65288	2.02803	2.51074	2.96795	3.40708	3.99033	4.59526
0.267658	0.585569	1.11255	1.66123	2.01732	2.51274	2.98121	3.38861	3.96268	4.58254
0.271295	0.587918	1.11658	1.66306	2.01411	2.51244	2.98947	3.3722	3.94238	4.5853
0.275028	0.596915	1.11573	1.65857	2.02155	2.51078	2.98864	3.37328	3.92916	4.60314
0.277614	0.607463	1.11622	1.65019	2.03456	2.50486	2.9819	3.38394	3.92154	4.6156
0.278064	0.615291	1.11464	1.64053	2.04334	2.49776	2.97339	3.39366	3.93098	4.62406
0.275439	0.618481	1.11514	1.6341	2.05293	2.49482	2.96852	3.40338	3.9401	4.63059
0.270626	0.613326	1.11629	1.63362	2.06934	2.4976	2.96844	3.39837	3.94944	4.64175
0.263975	0.599916	1.11976	1.63854	2.08788	2.50159	2.96516	3.38251	3.96133	4.65106
0.259672	0.583298	1.12126	1.64386	2.10529	2.50702	2.95378	3.35637	3.96288	4.65162
0.261296	0.571115	1.11886	1.65096	2.12785	2.51713	2.94253	3.3379	3.96944	4.65404
0.267317	0.567831	1.11508	1.65528	2.15512	2.52803	2.93107	3.33137	3.97934	4.64348
0.273175	0.573859	1.11422	1.65599	2.17824	2.53399	2.92503	3.33369	3.97599	4.62208
0.275657	0.584799	1.1201	1.64635	2.19242	2.53757	2.92438	3.34183	3.97704	4.59915
0.275925	0.597554	1.12527	1.63512	2.1951	2.53834	2.92577	3.3552	3.97397	4.58785
0.272384	0.609602	1.13069	1.63128	2.18734	2.53588	2.9299	3.37503	3.9756	4.58929
0.263501	0.619405	1.1376	1.63849	2.16615	2.53714	2.93002	3.3883	3.97081	4.59813
0.2503	0.621843	1.14417	1.64466	2.13368	2.5364	2.92577	3.40416	3.96329	4.60793
0.237226	0.616122	1.14768	1.65144	2.09882	2.53134	2.93065	3.42396	3.95308	4.60711
0.228973	0.603925	1.14704	1.65826	2.06997	2.5193	2.94664	3.43383	3.94496	4.60249
0.22812	0.596075	1.14432	1.6606	2.05115	2.50622	2.9652	3.44011	3.93741	4.59231
0.234814	0.599315	1.13576	1.65383	2.03824	2.49615	2.97854	3.42879	3.93238	4.5842
0.244689	0.603734	1.12304	1.64436	2.02975	2.49143	2.99226	3.41422	3.94286	4.5854
0.252299	0.607623	1.11703	1.63254	2.02019	2.49286	3.00072	3.3979	3.94574	4.58913
0.255558	0.604933	1.11958	1.62274	2.00741	2.49436	2.99687	3.40073	3.94794	4.59699
0.252617	0.594544	1.1229	1.61262	1.99771	2.49888	2.98465	3.40959	3.95164	4.59595
0.246427	0.575394	1.12493	1.60542	1.99517	2.50415	2.9571	3.41292	3.95574	4.59096
0.240804	0.553731	1.11765	1.60501	2.00452	2.50398	2.93668	3.41041	3.97157	4.59189

0.237656	0.54383	1.10739	1.61613	2.02585	2.50759	2.93048	3.38486	3.99629	4.58113
0.237636	0.553947	1.09693	1.63084	2.05285	2.51286	2.94679	3.35218	4.01101	4.57645
0.24325	0.581746	1.09166	1.64549	2.08369	2.51396	2.97641	3.32803	4.02882	4.57484
0.249792	0.613292	1.09356	1.65729	2.10958	2.51202	2.99827	3.3145	4.02901	4.57454
0.248588	0.634868	1.10011	1.66341	2.12204	2.51095	3.00124	3.3217	4.0131	4.58906
0.238882	0.637718	1.11182	1.66362	2.12629	2.50974	2.99774	3.34315	3.99557	4.60016
0.225685	0.627091	1.12534	1.65027	2.12984	2.50603	2.98534	3.3656	3.98588	4.60758
0.216094	0.608874	1.1311	1.63416	2.13885	2.50147	2.97748	3.38605	3.97616	4.61194
0.215199	0.592029	1.12533	1.61849	2.15002	2.49596	2.97017	3.39367	3.97728	4.62175
0.221847	0.580334	1.12049	1.61247	2.15831	2.49824	2.96673	3.38824	3.97327	4.62388
0.22753	0.575759	1.11593	1.61078	2.15967	2.50485	2.9663	3.38206	3.97485	4.63064
0.228079	0.576517	1.11099	1.62219	2.15801	2.52274	2.96156	3.37709	3.97369	4.63437
0.222446	0.580557	1.11363	1.63398	2.14868	2.54341	2.95212	3.38005	3.97038	4.6365
0.21353	0.58231	1.11961	1.64548	2.13213	2.55502	2.93735	3.37687	3.95533	4.6333
0.206454	0.581728	1.12513	1.64859	2.11418	2.55244	2.92328	3.36888	3.95039	4.62189
0.203571	0.580478	1.12441	1.64085	2.0968	2.53832	2.91865	3.35397	3.94534	4.61731
0.204092	0.58219	1.11761	1.63205	2.07446	2.52429	2.92645	3.34502	3.95462	4.61102
0.204801	0.590719	1.10665	1.63072	2.05237	2.51354	2.94081	3.34257	3.96901	4.60119
0.203783	0.604896	1.09556	1.63688	2.039	2.50595	2.95058	3.3425	3.97336	4.60524
0.201736	0.617026	1.09298	1.64353	2.03919	2.5024	2.96	3.35792	3.97966	4.61027
0.200002	0.618964	1.1012	1.6466	2.04614	2.5046	2.9775	3.3717	3.96869	4.6081
0.199342	0.613125	1.11774	1.64767	2.04644	2.50674	2.99127	3.3886	3.95673	4.59636
0.198965	0.606731	1.13424	1.64755	2.04074	2.50744	2.99542	3.39698	3.94384	4.58119
0.197868	0.598062	1.14226	1.64677	2.03386	2.51043	2.99033	3.39999	3.93716	4.5599
0.195778	0.590905	1.13799	1.64036	2.02887	2.51243	2.98092	3.40212	3.93448	4.55445
0.195126	0.589109	1.12392	1.62784	2.02481	2.5129	2.96765	3.4009	3.94006	4.55334
0.195749	0.591382	1.10714	1.61394	2.02802	2.51084	2.95108	3.40097	3.94949	4.56088
0.194727	0.591981	1.09246	1.60896	2.0418	2.50564	2.9382	3.4057	3.95721	4.58419
0.189635	0.586733	1.08134	1.60748	2.06362	2.49999	2.93817	3.40633	3.96166	4.6077
0.179164	0.581316	1.0787	1.60509	2.08743	2.49866	2.95039	3.40927	3.96821	4.61746
0.168671	0.579446	1.08608	1.6088	2.11161	2.49881	2.96989	3.40032	3.97206	4.611
0.164191	0.581661	1.09954	1.62241	2.12951	2.49738	2.99253	3.38625	3.98997	4.61083
0.16619	0.585778	1.1141	1.63742	2.14211	2.49944	3.01545	3.36746	4.01176	4.59536
0.169791	0.591415	1.12753	1.64739	2.15157	2.50095	3.01889	3.36002	4.02497	4.59663
0.172621	0.597871	1.1354	1.64244	2.15702	2.49846	3.00716	3.35503	4.02466	4.59025
0.175198	0.611379	1.1346	1.62717	2.15174	2.48816	2.99126	3.35599	4.01571	4.58422
0.174826	0.631765	1.12585	1.61671	2.1393	2.48565	2.97794	3.36911	3.99788	4.57808
0.168906	0.6499	1.11623	1.61522	2.12484	2.49055	2.96244	3.38637	3.99139	4.57
0.160905	0.653137	1.10688	1.61888	2.11258	2.49971	2.95535	3.39668	3.98539	4.56559
0.155351	0.643903	1.10003	1.62526	2.10879	2.51271	2.95327	3.4007	3.98099	4.57343
0.153828	0.626631	1.10137	1.63688	2.10073	2.52269	2.95284	3.39574	3.97085	4.57672
0.155713	0.607556	1.1086	1.64631	2.08478	2.52318	2.95529	3.38654	3.96598	4.58782
0.159778	0.589632	1.11316	1.64628	2.06839	2.51608	2.95594	3.36953	3.96581	4.58134

0.166656	0.575516	1.1133	1.64118	2.05411	2.50411	2.96541	3.35396	3.96485	4.57631
0.171041	0.568016	1.11321	1.63415	2.04274	2.49399	2.97528	3.34812	3.96157	4.57089
0.169476	0.568774	1.10839	1.62877	2.03499	2.48999	2.97965	3.34555	3.95863	4.57151
0.162932	0.574176	1.09993	1.62144	2.02697	2.49548	2.98003	3.35213	3.94752	4.57147
0.15459	0.578702	1.09336	1.61291	2.02303	2.50328	2.97498	3.36793	3.93317	4.5908
0.145035	0.581586	1.0965	1.60494	2.0272	2.50733	2.96506	3.37716	3.92563	4.60172
0.136002	0.583094	1.10624	1.604	2.03311	2.50384	2.95049	3.38006	3.93157	4.60609
0.132513	0.588058	1.124	1.60865	2.03543	2.50035	2.93932	3.36604	3.94868	4.60107
0.135033	0.598262	1.13818	1.61941	2.03988	2.49993	2.9329	3.35169	3.96764	4.58935
0.138781	0.615318	1.15129	1.6359	2.04806	2.50212	2.93741	3.33947	3.97782	4.5685
0.141215	0.629057	1.15823	1.63917	2.0589	2.50621	2.95175	3.33617	3.9794	4.56193
0.141734	0.63163	1.1587	1.63571	2.07576	2.51228	2.96232	3.34646	3.97708	4.55294
0.139783	0.624195	1.15401	1.62811	2.09719	2.5163	2.96148	3.37075	3.96896	4.55123
0.137903	0.612441	1.13843	1.62037	2.11603	2.51578	2.95809	3.3976	3.96822	4.55015
0.136869	0.602861	1.11544	1.61577	2.13018	2.51054	2.94585	3.41318	3.97192	4.5614
0.135177	0.601826	1.09419	1.61645	2.14249	2.50579	2.93171	3.42317	3.97512	4.57313
0.133864	0.606876	1.08404	1.62412	2.15374	2.49725	2.92923	3.42752	3.99288	4.58352
0.13543	0.613272	1.08391	1.63876	2.15867	2.48717	2.94168	3.41722	4.0005	4.59163
0.13977	0.613117	1.09063	1.64909	2.1543	2.47953	2.9659	3.41069	4.0054	4.58803
0.141888	0.609736	1.09667	1.64945	2.14403	2.47531	2.98987	3.40687	3.98796	4.58388
0.136277	0.60977	1.09959	1.63726	2.13262	2.47605	2.99843	3.40121	3.97495	4.58152
0.126026	0.60668	1.10123	1.6168	2.11904	2.48303	2.9987	3.39681	3.95534	4.57133
0.117379	0.597412	1.10012	1.59524	2.10457	2.49096	2.99407	3.38857	3.94906	4.5649
0.113898	0.587722	1.10251	1.57534	2.09252	2.49599	2.98872	3.38408	3.95379	4.56019
0.116202	0.584121	1.11525	1.56385	2.08196	2.49767	2.97266	3.3817	3.9642	4.56995
0.124763	0.592421	1.13569	1.56447	2.06597	2.49732	2.95591	3.3905	3.99344	4.57312
0.138863	0.605226	1.15254	1.58572	2.04237	2.49943	2.94276	3.39256	4.00955	4.58565
0.150657	0.620159	1.16518	1.6148	2.01906	2.49998	2.94021	3.38654	4.01282	4.60012
0.151361	0.634438	1.17117	1.6405	2.00092	2.49351	2.94751	3.38338	4.00476	4.60087
0.138234	0.643257	1.16761	1.65307	1.99266	2.48353	2.95899	3.3767	3.98547	4.60173
0.118225	0.640013	1.15506	1.64864	1.99735	2.47706	2.96942	3.36095	3.96982	4.59028
0.102244	0.624236	1.13978	1.63257	2.00844	2.47499	2.96283	3.34461	3.966	4.57667
0.09726	0.606842	1.12213	1.61007	2.0168	2.47888	2.94735	3.33996	3.96895	4.55788
0.103373	0.598691	1.10687	1.58978	2.02566	2.48749	2.9319	3.34001	3.97078	4.54186
0.116991	0.602786	1.09579	1.58265	2.04108	2.49661	2.91637	3.34573	3.98356	4.52873
0.128997	0.611673	1.09453	1.58276	2.06533	2.50324	2.91111	3.34879	3.98328	4.52648
0.13269	0.619813	1.10779	1.59623	2.09703	2.51117	2.91926	3.35137	3.98579	4.53071
0.127115	0.62302	1.12577	1.61806	2.13002	2.51277	2.93853	3.34754	3.97749	4.54194
0.119702	0.620121	1.13624	1.64115	2.15657	2.51011	2.96796	3.35253	3.96827	4.55483
0.118816	0.612675	1.14399	1.65541	2.16975	2.50387	2.98958	3.35616	3.95974	4.57444
0.123757	0.606074	1.14324	1.65424	2.17354	2.49763	2.99782	3.35958	3.96616	4.58178
0.127554	0.602274	1.13816	1.6442	2.17248	2.49415	2.99594	3.37155	3.98257	4.58293
0.12506	0.602324	1.12751	1.6269	2.16539	2.49057	2.99031	3.3832	4.00032	4.58551

0.116211	0.609199	1.11602	1.60767	2.14989	2.48589	2.98605	3.3972	4.00523	4.57919
0.104686	0.622013	1.10509	1.59084	2.1277	2.48724	2.98016	3.4085	4.00903	4.56795
0.096432	0.632886	1.10308	1.57561	2.10877	2.4934	2.97203	3.41888	4.00496	4.56236
0.094199	0.63569	1.11117	1.56318	2.09823	2.49993	2.95816	3.43322	3.99123	4.56403
0.097643	0.631917	1.12602	1.55798	2.09022	2.50207	2.94549	3.43614	3.97901	4.55842
0.106626	0.626073	1.14652	1.55479	2.08421	2.50215	2.93497	3.42318	3.97521	4.5436
0.117394	0.621555	1.16499	1.55919	2.0798	2.4978	2.9268	3.40492	3.97672	4.5496
0.127358	0.625056	1.17385	1.57487	2.07159	2.4981	2.92365	3.38732	3.9695	4.55794
0.131988	0.632083	1.1722	1.59972	2.05752	2.49838	2.92587	3.36842	3.95618	4.55844
0.128529	0.633152	1.15889	1.62772	2.04572	2.49622	2.93828	3.35719	3.93643	4.56507
0.121867	0.620457	1.1366	1.64554	2.03813	2.49092	2.94524	3.3547	3.93483	4.56321
0.118614	0.603127	1.11632	1.6492	2.03412	2.47683	2.94971	3.35601	3.92981	4.56529
0.117873	0.591997	1.10412	1.64311	2.03387	2.45833	2.95766	3.35825	3.9317	4.57109
0.116028	0.594271	1.10658	1.63377	2.03245	2.44423	2.97089	3.36178	3.94017	4.56642
0.112204	0.607844	1.12006	1.62425	2.02659	2.44275	2.98477	3.36682	3.94637	4.5547
0.110029	0.62325	1.13704	1.62025	2.0212	2.45079	2.99153	3.36917	3.96153	4.54919
0.110724	0.634486	1.15013	1.61177	2.02211	2.46341	2.99087	3.36801	3.97229	4.54762
0.11385	0.638225	1.15223	1.60335	2.03856	2.48311	2.98649	3.37374	3.98574	4.54108
0.117319	0.628484	1.15026	1.59406	2.06785	2.5016	2.97864	3.38364	3.99681	4.54069
0.118005	0.612188	1.15072	1.58539	2.10136	2.51971	2.96435	3.39388	3.99966	4.54716
0.117364	0.598613	1.15092	1.58221	2.13291	2.52894	2.94651	3.40555	3.98986	4.55632
0.115017	0.593648	1.14742	1.58485	2.15401	2.53367	2.92924	3.41634	3.99136	4.57252
0.114231	0.599995	1.14062	1.58621	2.16452	2.53448	2.91315	3.41964	3.9984	4.56146
0.121119	0.614375	1.13896	1.58427	2.16419	2.53897	2.90547	3.40633	4.01031	4.56411
0.135875	0.629517	1.14198	1.57711	2.15913	2.5369	2.91289	3.38801	4.01623	4.56776
0.150466	0.638288	1.14309	1.57126	2.15316	2.52856	2.92336	3.36283	4.02559	4.57048
0.157661	0.64	1.14594	1.57676	2.1458	2.5122	2.93088	3.33822	4.02647	4.57081
0.154299	0.635238	1.15125	1.59085	2.13744	2.48884	2.94476	3.32142	4.02273	4.56958
0.141873	0.623916	1.16047	1.60547	2.12799	2.46798	2.95616	3.31927	4.01201	4.5674
0.124342	0.613197	1.16619	1.61982	2.1187	2.46269	2.96619	3.32553	3.99755	4.57464
0.110862	0.604694	1.16665	1.62483	2.10408	2.46726	2.96965	3.3435	3.99785	4.5806
0.109779	0.603139	1.16387	1.62641	2.08587	2.47642	2.95941	3.36066	4.00267	4.59448
0.120556	0.607511	1.15773	1.62906	2.06674	2.48463	2.93711	3.37454	3.99885	4.591
0.13568	0.610358	1.1437	1.62912	2.04324	2.49502	2.91778	3.38133	3.97691	4.58386
0.149229	0.611946	1.13259	1.62407	2.02166	2.50349	2.91059	3.39165	3.94139	4.57523
0.156349	0.611643	1.12915	1.61245	2.00795	2.50513	2.91451	3.39644	3.90859	4.56069
0.155416	0.612738	1.13404	1.59685	2.00293	2.50231	2.9269	3.40609	3.89507	4.55846
0.150189	0.615683	1.14037	1.58058	2.005	2.4985	2.94751	3.41004	3.90781	4.56223
0.148801	0.621433	1.14553	1.57472	2.01379	2.48979	2.96868	3.41548	3.93183	4.56587
0.154832	0.625398	1.14704	1.57687	2.02933	2.48071	2.9822	3.41668	3.95408	4.55604
0.165217	0.626623	1.14224	1.58118	2.04872	2.47548	2.98444	3.40913	3.98279	4.55003
0.169811	0.624033	1.13044	1.58728	2.065	2.4797	2.98013	3.39109	3.99107	4.5298
0.161886	0.619627	1.12368	1.59312	2.07851	2.49565	2.97298	3.37775	3.99387	4.5189

0.143882	0.617428	1.13174	1.59965	2.09636	2.51599	2.9628	3.37269	3.98488	4.51884
0.125012	0.620271	1.14643	1.60307	2.11784	2.52971	2.957	3.37584	3.97087	4.52305
0.115245	0.624746	1.15789	1.60565	2.13814	2.52581	2.94811	3.3695	3.95348	4.55118
0.116925	0.625309	1.16518	1.61426	2.15089	2.50493	2.94104	3.36922	3.92943	4.56438
0.126632	0.625749	1.16729	1.62679	2.15917	2.48235	2.94383	3.36955	3.92123	4.58034
0.1391	0.631086	1.15883	1.63511	2.15939	2.46578	2.94032	3.38193	3.91408	4.59764
0.14902	0.638734	1.14861	1.63416	2.15247	2.46581	2.93637	3.39476	3.92398	4.59842
0.154732	0.644016	1.1411	1.61763	2.1413	2.47555	2.92843	3.40575	3.95123	4.59327
0.161405	0.644791	1.13749	1.59524	2.12948	2.48887	2.92081	3.41238	3.98766	4.58026
0.174095	0.641497	1.1358	1.58011	2.11924	2.50167	2.9176	3.41732	4.01869	4.57161
0.189182	0.629563	1.142	1.57187	2.10747	2.50965	2.91698	3.41204	4.03493	4.56272
0.198172	0.614148	1.14966	1.57627	2.09306	2.50565	2.92358	3.40721	4.04246	4.54554
0.199762	0.59882	1.15749	1.57935	2.07871	2.49355	2.92815	3.38895	4.03519	4.55029
0.198176	0.587111	1.15821	1.58593	2.06568	2.48135	2.92194	3.37521	4.0235	4.55109
0.194574	0.58174	1.15454	1.58715	2.05684	2.47532	2.91501	3.36336	4.02019	4.56278
0.186776	0.58411	1.15084	1.58435	2.05129	2.47596	2.91766	3.35145	4.01897	4.56518
0.178644	0.589824	1.14452	1.5841	2.04782	2.48251	2.93224	3.34362	4.02258	4.56216
0.17243	0.600791	1.1363	1.59489	2.04396	2.50124	2.94348	3.3395	4.02159	4.56197
0.168126	0.613165	1.12993	1.61041	2.04127	2.51996	2.95487	3.34314	4.01704	4.57289

Appendix H: Optimum Particle Spacing Details Using Improved Method

Table 52 through Table 54 show the output file details from the convergence optimizer using the improved height method for the three models with the optimum particle spacing highlighted in yellow and Table 55 through Table 57 show the printed information from the convergence optimizer for the three models using the improved height method.

Table 52: Small Box Convergence Optimizer Output File Details Using Improved Fluid Height Method

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
0.1	0.029677	23.51534
0.05	0.078571	23.69736
0.025	0.08872	39.38125
0.0125	0.095271	194.4271
0.00625	0.099361	2436.615
0.003125	0.101217	31553
0.002344	0.09986	101663.2
0.009375	0.092869	510.1832
0.007813	0.099908	1007.404
0.008594	0.098143	700.4971
0.008984	0.092602	580.4932
0.008789	0.095499	640.9677
0.008887	0.096676	586.8446
0.008936	0.091035	573.6738
0.008911	0.097455	633.8583
0.008923	0.097539	654.9385
0.008929	0.090904	577.685
0.008926	0.097572	612.696
0.008928	0.097856	619.3644

Table 53: Medium Box Convergence Optimizer Output File Details Using Improved Fluid Height Method

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
2	3.163105	37.01212

1	5.118497	39.61427
0.5	4.953608	50.41288
0.25	4.907479	182.0364
0.125	5.089558	1837.473
1.5	4.316904	39.91928
1.25	4.610925	37.57015
1.125	3.921778	36.92711
1.0625	3.556892	36.84511
1.03125	3.280053	36.92111
1.015625	3.528362	37.28913
1.007813	3.4789	38.16218
1.003906	3.696999	37.78916
1.001953	3.258203	38.22319
1.000977	3.489519	37.96817
1.000488	3.413765	37.73716
1.000244	3.312004	39.13224
1.000122	3.487134	37.64115
1.000061	3.538908	38.68821
1.000031	3.22129	39.19424
1.000015	3.420261	37.72316
1.000008	3.271665	39.78828
1.000004	3.352646	38.02818
1.000002	3.36877	41.23936

Table 54: Large Box Convergence Optimizer Output File Details Using Improved Fluid Height Method

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
5	37.28597	291.2197
2.5	38.86227	367.944
1.25	39.11476	1043.326
0.625	39.8481	6290.052
0.3125	40.58852	96623.85
3.75	32.2625	305.8525
3.125	33.64204	322.3534
2.8125	36.55569	330.6809
2.65625	33.32269	334.0301
2.578125	31.29173	330.7439
2.539063	30.8349	335.6012
2.519531	30.08928	329.3168
2.509766	29.7925	338.7784
2.504883	29.78349	342.9026

2.502441	29.4136	333.8341
2.501221	29.4562	372.1823
2.50061	29.50027	427.4084
2.500305	29.54609	338.7644
2.500153	29.35059	350.6471
2.500076	29.30751	337.0933
2.500038	29.53202	344.1267
2.500019	29.50361	374.1474
2.50001	29.49683	336.4522
2.500005	29.43981	332.107
2.500002	29.39759	338.9474
2.500001	29.54654	336.0802

Table 55: Small Box Printed Convergence Optimizer Information Using Improved Fluid Height Method

```
The starting average is 0.02967650000000001
Current Spacing = 0.05
Average = 0.07857075000000001
Difference Value = 1.647574680302596
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 0.07857075000000001
Smallest non-converged particle spacing: 0.1
Smallest non-converged particle spacing average: 0.02967650000000001
Current Spacing = 0.025
Average = 0.08872045255102042
Difference Value = 0.12917914810562972
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 0.08872045255102042
Smallest non-converged particle spacing: 0.05
Smallest non-converged particle spacing average: 0.07857075000000001
Current Spacing = 0.0125
Average = 0.09527092499999999
Difference Value = 0.07383272132446125
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 0.09527092499999999
Smallest non-converged particle spacing: 0.025
Smallest non-converged particle spacing average: 0.08872045255102042
Current Spacing = 0.00625
Average = 0.09936115051020408
Difference Value = 0.042932568464136255
Converged iteration = 1
```

```
Difference Value = 0.04116523902150304
Largest converged particle spacing: 0.0125
Largest converged particle spacing average: 0.09527092499999999
Smallest converged particle spacing: 0.00625
Smallest converged particle spacing average: 0.09936115051020408
Smallest non-converged particle spacing: 0.025
Smallest non-converged particle spacing average: 0.08872045255102042
Current Spacing = 0.003125
Average = 0.10121688949367347
Difference Value = 0.018676705874886245
Converged iteration = 2
Difference Value = 0.0587447858101303
Updated the largest converged particle spacing since it was not longer
within convergence criteria
Largest converged particle spacing: 0.00625
Largest converged particle spacing average: 0.09936115051020408
Smallest converged particle spacing: 0.003125
Smallest converged particle spacing average: 0.10121688949367347
Smallest non-converged particle spacing: 0.0125
Smallest non-converged particle spacing average: 0.09527092499999999
Current Spacing = 0.00234375
Average = 0.09985992113294082
Difference Value = 0.013406540820615409
Converged iteration = 3
Difference Value = 0.0049947027503930526
Largest converged particle spacing: 0.00625
Largest converged particle spacing average: 0.09936115051020408
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.0125
Smallest non-converged particle spacing average: 0.09527092499999999
Optimizing particle spacing.
Current Spacing = 0.009375000000000001
Average = 0.09286946768707345
Difference Value = 0.07000259329827792
Did not converge
Largest converged particle spacing: 0.00625
Largest converged particle spacing average: 0.09936115051020408
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.009375000000000001
Smallest non-converged particle spacing average: 0.09286946768707345
Optimizing particle spacing.
Current Spacing = 0.0078125
Average = 0.09990794451530613
Difference Value = 0.0004809074733934748
Difference Value = 0.0004809074733934748
Largest converged particle spacing: 0.0078125
Largest converged particle spacing average: 0.09990794451530613
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.009375000000000001
Smallest non-converged particle spacing average: 0.09286946768707345
Optimizing particle spacing.
Current Spacing = 0.00859375
Average = 0.09814327239708878
Difference Value = 0.01719056771101096
```

```
Difference Value = 0.01719056771101096
Largest converged particle spacing: 0.00859375
Largest converged particle spacing average: 0.09814327239708878
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.009375000000000001
Smallest non-converged particle spacing average: 0.09286946768707345
Optimizing particle spacing.
Current Spacing = 0.008984375000000001
Average = 0.09260185368076121
Difference Value = 0.07268248732659359
Did not converge
Largest converged particle spacing: 0.00859375
Largest converged particle spacing average: 0.09814327239708878
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008984375000000001
Smallest non-converged particle spacing average: 0.09260185368076121
Optimizing particle spacing.
Current Spacing = 0.0087890625
Average = 0.09549944522018979
Difference Value = 0.04366592586174834
Difference Value = 0.04366592586174834
Largest converged particle spacing: 0.0087890625
Largest converged particle spacing average: 0.09549944522018979
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008984375000000001
Smallest non-converged particle spacing average: 0.09260185368076121
Optimizing particle spacing.
Current Spacing = 0.008886718750000001
Average = 0.09667562567131634
Difference Value = 0.03188762243648592
Difference Value = 0.03188762243648592
Largest converged particle spacing: 0.008886718750000001
Largest converged particle spacing average: 0.09667562567131634
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008984375000000001
Smallest non-converged particle spacing average: 0.09260185368076121
Optimizing particle spacing.
Current Spacing = 0.008935546875000002
Average = 0.09103490579445611
Difference Value = 0.08837394660803106
Did not converge
Largest converged particle spacing: 0.008886718750000001
Largest converged particle spacing average: 0.09667562567131634
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008935546875000002
Smallest non-converged particle spacing average: 0.09103490579445611
Optimizing particle spacing.
Current Spacing = 0.008911132812500002
Average = 0.09745545739346634
Difference Value = 0.024078366097180066
Difference Value = 0.024078366097180066
Largest converged particle spacing: 0.008911132812500002
```

```
Largest converged particle spacing average: 0.09745545739346634
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008935546875000002
Smallest non-converged particle spacing average: 0.09103490579445611
Optimizing particle spacing.
Current Spacing = 0.008923339843750003
Average = 0.09753897113318467
Difference Value = 0.023242057207979602
Difference Value = 0.023242057207979602
Largest converged particle spacing: 0.008923339843750003
Largest converged particle spacing average: 0.09753897113318467
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008935546875000002
Smallest non-converged particle spacing average: 0.09103490579445611
Optimizing particle spacing.
Current Spacing = 0.008929443359375002
Average = 0.09090440943877552
Difference Value = 0.08968074070720589
Did not converge
Largest converged particle spacing: 0.008923339843750003
Largest converged particle spacing average: 0.09753897113318467
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008929443359375002
Smallest non-converged particle spacing average: 0.09090440943877552
Optimizing particle spacing.
Current Spacing = 0.008926391601562503
Average = 0.09757191425950408
Difference Value = 0.0229121638338846
Difference Value = 0.0229121638338846
Largest converged particle spacing: 0.008926391601562503
Largest converged particle spacing average: 0.09757191425950408
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008929443359375002
Smallest non-converged particle spacing average: 0.09090440943877552
Optimizing particle spacing.
Current Spacing = 0.008927917480468752
Average = 0.09785551450691325
Difference Value = 0.020072183147022027
Difference Value = 0.020072183147022027
Largest converged particle spacing: 0.008927917480468752
Largest converged particle spacing average: 0.09785551450691325
Smallest converged particle spacing: 0.00234375
Smallest converged particle spacing average: 0.09985992113294082
Smallest non-converged particle spacing: 0.008929443359375002
Smallest non-converged particle spacing average: 0.09090440943877552
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 0.008927917480468752
Total runtime: 143632.4804842472
```

Table 56: Medium Box Printed Convergence Optimizer Information Using Improved Fluid Height Method

```
The starting average is 3.1631053846151924
Current Spacing = 1.0
Average = 5.118497386363636
Difference Value = 0.6181874341775453
Did not converge
Largest converged particle spacing: 0
Largest converged particle spacing average: 0
Smallest converged particle spacing: 0
Smallest converged particle spacing average: 5.118497386363636
Smallest non-converged particle spacing: 2
Smallest non-converged particle spacing average: 3.1631053846151924
Current Spacing = 0.5
Average = 4.953607714646465
Difference Value = 0.03221446828447322
Converged iteration = 1
Difference Value = 0.033286784342982316
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.5
Smallest converged particle spacing average: 4.953607714646465
Smallest non-converged particle spacing: 2
Smallest non-converged particle spacing average: 3.1631053846151924
Current Spacing = 0.25
Average = 4.907478592171717
Difference Value = 0.009312227598959237
Converged iteration = 2
Difference Value = 0.042999432443481325
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.25
Smallest converged particle spacing average: 4.907478592171717
Smallest non-converged particle spacing: 2
Smallest non-converged particle spacing average: 3.1631053846151924
Current Spacing = 0.125
Average = 5.0895576010101005
Difference Value = 0.037102354176099916
Converged iteration = 3
Difference Value = 0.0056861101931122305
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 2
Smallest non-converged particle spacing average: 3.1631053846151924
Optimizing particle spacing.
Current Spacing = 1.5
Average = 4.31690375641
Difference Value = 0.1518115925138081
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
```

```
Smallest non-converged particle spacing: 1.5
Smallest non-converged particle spacing average: 4.31690375641
Optimizing particle spacing.
Current Spacing = 1.25
Average = 4.610925437710808
Difference Value = 0.09404199752141534
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.25
Smallest non-converged particle spacing average: 4.610925437710808
Optimizing particle spacing.
Current Spacing = 1.125
Average = 3.921777898351346
Difference Value = 0.22944621010419272
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.125
Smallest non-converged particle spacing average: 3.921777898351346
Optimizing particle spacing.
Current Spacing = 1.0625
Average = 3.5568915453296155
Difference Value = 0.3011393476274449
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0625
Smallest non-converged particle spacing average: 3.5568915453296155
Optimizing particle spacing.
Current Spacing = 1.03125
Average = 3.2800532280219232
Difference Value = 0.35553274269438534
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.03125
Smallest non-converged particle spacing average: 3.2800532280219232
Optimizing particle spacing.
Current Spacing = 1.015625
Average = 3.5283615656565654
Difference Value = 0.30674493890071935
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.015625
Smallest non-converged particle spacing average: 3.5283615656565654
```

```
Optimizing particle spacing.
Current Spacing = 1.0078125
Average = 3.478899962121212
Difference Value = 0.31646319094005904
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0078125
Smallest non-converged particle spacing average: 3.478899962121212
Optimizing particle spacing.
Current Spacing = 1.00390625
Average = 3.6969992067307698
Difference Value = 0.2736108918391957
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.00390625
Smallest non-converged particle spacing average: 3.6969992067307698
Optimizing particle spacing.
Current Spacing = 1.001953125
Average = 3.258202840909091
Difference Value = 0.35982592273590713
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.001953125
Smallest non-converged particle spacing average: 3.258202840909091
Optimizing particle spacing.
Current Spacing = 1.0009765625
Average = 3.4895189393939394
Difference Value = 0.314376766518687
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0009765625
Smallest non-converged particle spacing average: 3.4895189393939394
Optimizing particle spacing.
Current Spacing = 1.00048828125
Average = 3.4137647186147473
Difference Value = 0.3292610112247804
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.00048828125
Smallest non-converged particle spacing average: 3.4137647186147473
Optimizing particle spacing.
Current Spacing = 1.000244140625
```

```
Average = 3.3120035984848486
Difference Value = 0.349255110537008
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.000244140625
Smallest non-converged particle spacing average: 3.3120035984848486
Optimizing particle spacing.
Current Spacing = 1.0001220703125
Average = 3.487133611111111
Difference Value = 0.3148454375643502
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0001220703125
Smallest non-converged particle spacing average: 3.487133611111111
Optimizing particle spacing.
Current Spacing = 1.00006103515625
Average = 3.5389079292929293
Difference Value = 0.30467278165973033
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.00006103515625
Smallest non-converged particle spacing average: 3.5389079292929293
Optimizing particle spacing.
Current Spacing = 1.000030517578125
Average = 3.2212897095959594
Difference Value = 0.36707864177494615
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.000030517578125
Smallest non-converged particle spacing average: 3.2212897095959594
Optimizing particle spacing.
Current Spacing = 1.0000152587890625
Average = 3.420260858585858
Difference Value = 0.3279846448919146
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0000152587890625
Smallest non-converged particle spacing average: 3.420260858585858
Optimizing particle spacing.
Current Spacing = 1.0000076293945312
Average = 3.271665315656566
Difference Value = 0.3571808058509341
```

```
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0000076293945312
Smallest non-converged particle spacing average: 3.271665315656566
Optimizing particle spacing.
Current Spacing = 1.0000038146972656
Average = 3.3526464772727267
Difference Value = 0.34126956798615604
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0000038146972656
Smallest non-converged particle spacing average: 3.3526464772727267
Optimizing particle spacing.
Current Spacing = 1.0000019073486328
Average = 3.3687702020202024
Difference Value = 0.33810156675471786
Did not converge
Largest converged particle spacing: 1.0
Largest converged particle spacing average: 5.118497386363636
Smallest converged particle spacing: 0.125
Smallest converged particle spacing average: 5.0895576010101005
Smallest non-converged particle spacing: 1.0000019073486328
Smallest non-converged particle spacing average: 3.3687702020202024
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 1.0
Total runtime: 2873.355346441269
```

Table 57: Large Box Printed Convergence Optimizer Information Using Improved Fluid Height Method

```
The starting average is 37.28597267857143
Current Spacing = 2.5
Average = 38.86227133838384
Difference Value = 0.042275916291660026
Converged iteration = 1
Difference Value = 0.04056115624553101
Largest converged particle spacing: 5
Largest converged particle spacing average: 37.28597267857143
Smallest converged particle spacing: 2.5
Smallest converged particle spacing average: 38.86227133838384
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 0
Current Spacing = 1.25
Average = 39.114759785353534
Difference Value = 0.006497006949779407
Converged iteration = 2
Difference Value = 0.04675439953658844
Largest converged particle spacing: 5
```

```
Largest converged particle spacing average: 37.28597267857143
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 39.114759785353534
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 0
Current Spacing = 0.625
Average = 39.84809626262626
Difference Value = 0.018748331353611408
Converged iteration = 3
Difference Value = 0.06429726447077105
Updated the largest converged particle spacing since it was not longer
within convergence criteria
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 39.84809626262626
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 37.28597267857143
Current Spacing = 0.3125
Average = 40.58851954861616
Difference Value = 0.018581145786990893
Converged iteration = 3
Difference Value = 0.04253045514913775
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 37.28597267857143
Optimizing particle spacing.
Current Spacing = 3.75
Average = 32.262504545455556
Difference Value = 0.20513226635891124
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 3.75
Smallest non-converged particle spacing average: 32.262504545455556
Optimizing particle spacing.
Current Spacing = 3.125
Average = 33.64204184704041
Difference Value = 0.17114390420807027
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 3.125
Smallest non-converged particle spacing average: 33.64204184704041
Optimizing particle spacing.
Current Spacing = 2.8125
Average = 36.55569393939393
Difference Value = 0.09935877568512413
Did not converge
Largest converged particle spacing: 2.5
```

```
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.8125
Smallest non-converged particle spacing average: 36.55569393939393
Optimizing particle spacing.
Current Spacing = 2.65625
Average = 33.322693434343435
Difference Value = 0.17901185347669205
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.65625
Smallest non-converged particle spacing average: 33.322693434343435
Optimizing particle spacing.
Current Spacing = 2.578125
Average = 31.29172512626262
Difference Value = 0.2290498526613669
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.578125
Smallest non-converged particle spacing average: 31.29172512626262
Optimizing particle spacing.
Current Spacing = 2.5390625
Average = 30.83489987373737
Difference Value = 0.24030488875545436
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.5390625
Smallest non-converged particle spacing average: 30.83489987373737
Optimizing particle spacing.
Current Spacing = 2.51953125
Average = 30.089282323232325
Difference Value = 0.25867504757861515
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.51953125
Smallest non-converged particle spacing average: 30.089282323232325
Optimizing particle spacing.
Current Spacing = 2.509765625
Average = 29.792498989898988
Difference Value = 0.2659870495100444
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
```

```
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.509765625
Smallest non-converged particle spacing average: 29.792498989898988
Optimizing particle spacing.
Current Spacing = 2.5048828125
Average = 29.783491919191917
Difference Value = 0.2662089612921749
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.5048828125
Smallest non-converged particle spacing average: 29.783491919191917
Optimizing particle spacing.
Current Spacing = 2.50244140625
Average = 29.413603409090914
Difference Value = 0.2753220926459302
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.50244140625
Smallest non-converged particle spacing average: 29.413603409090914
Optimizing particle spacing.
Current Spacing = 2.501220703125
Average = 29.456196464646457
Difference Value = 0.2742727058727929
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.501220703125
Smallest non-converged particle spacing average: 29.456196464646457
Optimizing particle spacing.
Current Spacing = 2.5006103515625
Average = 29.500265530303025
Difference Value = 0.2731869538880775
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.5006103515625
Smallest non-converged particle spacing average: 29.500265530303025
Optimizing particle spacing.
Current Spacing = 2.50030517578125
Average = 29.546087752525256
Difference Value = 0.2720580084933743
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.50030517578125
```

```
Smallest non-converged particle spacing average: 29.546087752525256
Optimizing particle spacing.
Current Spacing = 2.500152587890625
Average = 29.350588888888886
Difference Value = 0.2768746134302015
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.500152587890625
Smallest non-converged particle spacing average: 29.350588888888886
Optimizing particle spacing.
Current Spacing = 2.5000762939453125
Average = 29.307505050505053
Difference Value = 0.2779360918694983
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.5000762939453125
Smallest non-converged particle spacing average: 29.307505050505053
Optimizing particle spacing.
Current Spacing = 2.5000381469726562
Average = 29.5320202020202
Difference Value = 0.2724045978901176
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.5000381469726562
Smallest non-converged particle spacing average: 29.5320202020202
Optimizing particle spacing.
Current Spacing = 2.500019073486328
Average = 29.50361275252525
Difference Value = 0.2731044866717452
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.500019073486328
Smallest non-converged particle spacing average: 29.50361275252525
Optimizing particle spacing.
Current Spacing = 2.500009536743164
Average = 29.496825
Difference Value = 0.2732717199830543
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.500009536743164
Smallest non-converged particle spacing average: 29.496825
Optimizing particle spacing.
```

```
Current Spacing = 2.500004768371582
Average = 29.439812121212118
Difference Value = 0.2746763752752877
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.500004768371582
Smallest non-converged particle spacing average: 29.439812121212118
Optimizing particle spacing.
Current Spacing = 2.500002384185791
Average = 29.397589520202022
Difference Value = 0.27571663497137056
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.500002384185791
Smallest non-converged particle spacing average: 29.397589520202022
Optimizing particle spacing.
Current Spacing = 2.5000011920928955
Average = 29.546536868686875
Difference Value = 0.27204694339007385
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 38.86227133838384
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 40.58851954861616
Smallest non-converged particle spacing: 2.5000011920928955
Smallest non-converged particle spacing average: 29.546536868686875
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 2.5
Total runtime: 111808.46707868576
```

Appendix I: Particle Spacing Protocol Verification Test Details

Table 58 through Table 60 show the output file details from the convergence optimizer using the improved height method for the three test models with the optimum particle spacing highlighted in yellow and Table 61 through Table 63 show the printed information from the convergence optimizer for the three test models using the improved height method.

Table 58: Small Test Box Convergence Optimizer Output File Details Using Improved Fluid Height Method

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
0.01	0.048449	67.49586
0.005	0.056621	164.7144
0.0025	0.058374	1252.2
0.00125	0.058976	16487.52
0.000625	0.059409	220082.7
0.0075	0.025058	85.57189
0.00625	0.045799	107.6002
0.005625	0.036965	116.7457
0.005312	0.033407	126.2502
0.005156	0.031356	133.3816
0.005078	0.030293	137.4779
0.005039	0.030464	131.3435
0.00502	0.029766	135.9478
0.00501	0.030142	132.3066
0.005005	0.029977	133.8947
0.005002	0.029934	129.5494
0.005001	0.030005	134.3147

Table 59: Medium Test Box Convergence Optimizer Output File Details Using Improved Fluid Height Method

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
5	13.95113	406.5883
2.5	16.8342	420.6351

1.25	16.94303	617.8373
0.625	17.10733	2937.431
0.3125	17.62168	18591.37
3.75	8.206629	403.6731
3.125	14.02486	427.0764
2.8125	11.60985	431.2177
2.65625	9.665746	421.5241
2.578125	9.193847	412.3826
2.539063	8.627743	420.7551
2.519531	9.317273	414.0467
2.509766	9.045679	413.9967
2.504883	8.817479	424.6333
2.502441	9.059897	448.4446
2.501221	8.842158	430.5826
2.50061	8.922939	509.0001
2.500305	9.012265	457.7732
2.500153	8.730282	409.6814
2.500076	8.709626	420.266
2.500038	8.840738	391.7654
2.500019	8.733356	405.9372
2.50001	8.825089	414.7287
2.500005	8.892828	412.3566
2.500002	8.887418	402.797
2.500001	8.821927	398.1368

Table 60: Large Test Box Convergence Optimizer Output File Details Using Improved Fluid Height Method

Particle Spacing (m)	Fluid Height (m)	Computational Runtime (sec)
20	28.92614	92.80131
10	49.90814	101.0618
5	51.36721	119.6658
2.5	51.94182	406.8133
1.25	52.4663	2708.553
15	40.19369	99.19567
13	31.5256	93.12533
11.5	45.96578	93.09232
10.75	33.99071	93.69536
10.375	38.39679	101.2008
10.1875	35.73324	97.9956
10.09375	35.65834	94.4094
10.04688	34.56162	95.91949

10.02344	33.75955	99.6987
10.01172	35.51838	95.62947
10.00586	32.69982	140.082
10.00293	34.22729	121.2069
10.00146	34.5053	100.5648
10.00073	35.48911	102.8109
10.00037	33.61556	101.7118
10.00018	35.96689	103.5309
10.00009	37.72462	101.8318
10.00005	33.12644	101.3898
10.00002	34.24263	102.0678
10.00001	36.22752	102.5239
10.00001	33.52434	99.11167

Table 61: Small Test Box Printed Convergence Optimizer Information Using Improved Fluid Height Method

```
The starting average is 0.0484492196969697
Current Spacing = 0.005
Average = 0.05662083863636364
Difference Value = 0.16866358200408837
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.005
Smallest converged particle spacing average: 0.05662083863636364
Smallest non-converged particle spacing: 0.01
Smallest non-converged particle spacing average: 0.0484492196969697
Current Spacing = 0.0025
Average = 0.058373749999999995
Difference Value = 0.030958767228689184
Converged iteration = 1
Difference Value = 0.030029103212254705
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.0025
Smallest converged particle spacing average: 0.058373749999999995
Smallest non-converged particle spacing: 0.01
Smallest non-converged particle spacing average: 0.0484492196969697
Current Spacing = 0.00125
Average = 0.05897603888888888
Difference Value = 0.010317803617017677
Converged iteration = 2
Difference Value = 0.039934866717014454
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.00125
Smallest converged particle spacing average: 0.05897603888888888
Smallest non-converged particle spacing: 0.01
Smallest non-converged particle spacing average: 0.0484492196969697
```

```
Current Spacing = 0.000625
Average = 0.059409166982323236
Difference Value = 0.007344136730687706
Converged iteration = 3
Difference Value = 0.04693431144707416
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.01
Smallest non-converged particle spacing average: 0.0484492196969697
Optimizing particle spacing.
Current Spacing = 0.0075
Average = 0.02505766936027172
Difference Value = 0.5782188064052902
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.0075
Smallest non-converged particle spacing average: 0.02505766936027172
Optimizing particle spacing.
Current Spacing = 0.00625
Average = 0.04579935664982929
Difference Value = 0.2290860320688127
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.00625
Smallest non-converged particle spacing average: 0.04579935664982929
Optimizing particle spacing.
Current Spacing = 0.005625
Average = 0.03696483771043838
Difference Value = 0.3777923578454315
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.005625
Smallest non-converged particle spacing average: 0.03696483771043838
Optimizing particle spacing.
Current Spacing = 0.0053124999999999995
Average = 0.033406925252525255
Difference Value = 0.43768063163610355
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.0053124999999999995
Smallest non-converged particle spacing average: 0.033406925252525255
Optimizing particle spacing.
Current Spacing = 0.005156249999999999
```

```
Average = 0.03135626978114445
Difference Value = 0.47219812406266737
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.005156249999999999
Smallest non-converged particle spacing average: 0.03135626978114445
Optimizing particle spacing.
Current Spacing = 0.005078124999999999
Average = 0.030292650252525248
Difference Value = 0.49010141378453254
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.005078124999999999
Smallest non-converged particle spacing average: 0.030292650252525248
Optimizing particle spacing.
Current Spacing = 0.0050390625
Average = 0.030463861363636366
Difference Value = 0.4872195165991695
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.0050390625
Smallest non-converged particle spacing average: 0.030463861363636366
Optimizing particle spacing.
Current Spacing = 0.0050195312500000006
Average = 0.029765551515151513
Difference Value = 0.4989737606654536
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.0050195312500000006
Smallest non-converged particle spacing average: 0.029765551515151513
Optimizing particle spacing.
Current Spacing = 0.005009765625000001
Average = 0.030141620959595963
Difference Value = 0.49264360214704944
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.005009765625000001
Smallest non-converged particle spacing average: 0.030141620959595963
Optimizing particle spacing.
Current Spacing = 0.0050048828125
Average = 0.029977371212121205
Difference Value = 0.4954083227418295
```

```
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.0050048828125
Smallest non-converged particle spacing average: 0.029977371212121205
Optimizing particle spacing.
Current Spacing = 0.0050024414062500005
Average = 0.029934306818181818
Difference Value = 0.49613320067105887
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.0050024414062500005
Smallest non-converged particle spacing average: 0.029934306818181818
Optimizing particle spacing.
Current Spacing = 0.005001220703125
Average = 0.03000475176767677
Difference Value = 0.49494744175415784
Did not converge
Largest converged particle spacing: 0.005
Largest converged particle spacing average: 0.05662083863636364
Smallest converged particle spacing: 0.000625
Smallest converged particle spacing average: 0.059409166982323236
Smallest non-converged particle spacing: 0.005001220703125
Smallest non-converged particle spacing average: 0.03000475176767677
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 0.005
Total runtime: 239559.08157253265
```

Table 62: Medium Test Box Printed Convergence Optimizer Information Using Improved Fluid Height Method

```
The starting average is 13.95113316875
Current Spacing = 2.5
Average = 16.834200345758585
Difference Value = 0.20665469551007837
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 2.5
Smallest converged particle spacing average: 16.834200345758585
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 13.95113316875
Current Spacing = 1.25
Average = 16.943027196969698
Difference Value = 0.00646462849294365
Converged iteration = 1
```

```
Difference Value = 0.006423105502101581
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 16.943027196969698
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 13.95113316875
Current Spacing = 0.625
Average = 17.107330602904042
Difference Value = 0.009697405547677472
Converged iteration = 2
Difference Value = 0.015965685324342303
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.625
Smallest converged particle spacing average: 17.107330602904042
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 13.95113316875
Current Spacing = 0.3125
Average = 17.621684492983672
Difference Value = 0.03006628573556155
Converged iteration = 3
Difference Value = 0.04468835811574283
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 5
Smallest non-converged particle spacing average: 13.95113316875
Optimizing particle spacing.
Current Spacing = 3.75
Average = 8.206628986110973
Difference Value = 0.5342880534844123
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 3.75
Smallest non-converged particle spacing average: 8.206628986110973
Optimizing particle spacing.
Current Spacing = 3.125
Average = 14.024860518145834
Difference Value = 0.20411351572376438
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 3.125
Smallest non-converged particle spacing average: 14.024860518145834
Optimizing particle spacing.
Current Spacing = 2.8125
Average = 11.609849668110105
Difference Value = 0.34116118849292
Did not converge
Largest converged particle spacing: 2.5
```

```
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.8125
Smallest non-converged particle spacing average: 11.609849668110105
Optimizing particle spacing.
Current Spacing = 2.65625
Average = 9.665746349206946
Difference Value = 0.4514856764655216
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.65625
Smallest non-converged particle spacing average: 9.665746349206946
Optimizing particle spacing.
Current Spacing = 2.578125
Average = 9.193847118055556
Difference Value = 0.4782651385163422
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.578125
Smallest non-converged particle spacing average: 9.193847118055556
Optimizing particle spacing.
Current Spacing = 2.5390625
Average = 8.627743246527778
Difference Value = 0.5103905503493131
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.5390625
Smallest non-converged particle spacing average: 8.627743246527778
Optimizing particle spacing.
Current Spacing = 2.51953125
Average = 9.317273003472222
Difference Value = 0.47126093381242706
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.51953125
Smallest non-converged particle spacing average: 9.317273003472222
Optimizing particle spacing.
Current Spacing = 2.509765625
Average = 9.045678819444444
Difference Value = 0.48667343221097215
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
```

```
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.509765625
Smallest non-converged particle spacing average: 9.045678819444444
Optimizing particle spacing.
Current Spacing = 2.5048828125
Average = 8.817478939393938
Difference Value = 0.4996233792005103
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.5048828125
Smallest non-converged particle spacing average: 8.817478939393938
Optimizing particle spacing.
Current Spacing = 2.50244140625
Average = 9.059897474747475
Difference Value = 0.4858665482091225
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.50244140625
Smallest non-converged particle spacing average: 9.059897474747475
Optimizing particle spacing.
Current Spacing = 2.501220703125
Average = 8.84215751736111
Difference Value = 0.4982229127481119
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.501220703125
Smallest non-converged particle spacing average: 8.84215751736111
Optimizing particle spacing.
Current Spacing = 2.5006103515625
Average = 8.922939097222221
Difference Value = 0.49363869834492735
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.5006103515625
Smallest non-converged particle spacing average: 8.922939097222221
Optimizing particle spacing.
Current Spacing = 2.50030517578125
Average = 9.012265225694444
Difference Value = 0.4885695956432084
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.50030517578125
```

```
Smallest non-converged particle spacing average: 9.012265225694444
Optimizing particle spacing.
Current Spacing = 2.500152587890625
Average = 8.730282326388888
Difference Value = 0.5045716355967571
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.500152587890625
Smallest non-converged particle spacing average: 8.730282326388888
Optimizing particle spacing.
Current Spacing = 2.5000762939453125
Average = 8.709625677083332
Difference Value = 0.5057438645805289
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.5000762939453125
Smallest non-converged particle spacing average: 8.709625677083332
Optimizing particle spacing.
Current Spacing = 2.5000381469726562
Average = 8.8407384375
Difference Value = 0.4983034430664067
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.5000381469726562
Smallest non-converged particle spacing average: 8.8407384375
Optimizing particle spacing.
Current Spacing = 2.500019073486328
Average = 8.733355902777777
Difference Value = 0.5043972154730674
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.500019073486328
Smallest non-converged particle spacing average: 8.733355902777777
Optimizing particle spacing.
Current Spacing = 2.500009536743164
Average = 8.825088958333332
Difference Value = 0.4991915238383045
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.500009536743164
Smallest non-converged particle spacing average: 8.825088958333332
Optimizing particle spacing.
```

```
Current Spacing = 2.500004768371582
Average = 8.89282810763889
Difference Value = 0.4953474447247255
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.500004768371582
Smallest non-converged particle spacing average: 8.89282810763889
Optimizing particle spacing.
Current Spacing = 2.500002384185791
Average = 8.887417743055556
Difference Value = 0.4956544735212908
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.500002384185791
Smallest non-converged particle spacing average: 8.887417743055556
Optimizing particle spacing.
Current Spacing = 2.5000011920928955
Average = 8.821926631944443
Difference Value = 0.49937098037040556
Did not converge
Largest converged particle spacing: 2.5
Largest converged particle spacing average: 16.834200345758585
Smallest converged particle spacing: 0.3125
Smallest converged particle spacing average: 17.621684492983672
Smallest non-converged particle spacing: 2.5000011920928955
Smallest non-converged particle spacing average: 8.821926631944443
Optimizing particle spacing.
Minimum change in particle spacing was reached. Exit Optimizer.
The maximum converged particle spacing found was: 2.5
Total runtime: 31844.665410757065
```

Table 63: Large Test Box Printed Convergence Optimizer Information Using Improved Fluid Height Method

```
The starting average is 28.92614444444848
Current Spacing = 10.0
Average = 49.90813700154039
Difference Value = 0.7253643013982386
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 10.0
Smallest converged particle spacing average: 49.90813700154039
Smallest non-converged particle spacing: 20
Smallest non-converged particle spacing average: 28.92614444444848
Current Spacing = 5.0
Average = 51.36721056471414
Difference Value = 0.029235183896540146
Converged iteration = 1
```

```
Difference Value = 0.028404765357767734
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 5.0
Smallest converged particle spacing average: 51.36721056471414
Smallest non-converged particle spacing: 20
Smallest non-converged particle spacing average: 28.92614444444848
Current Spacing = 2.5
Average = 51.94182488825657
Difference Value = 0.011186403100836254
Converged iteration = 2
Difference Value = 0.03915318514686943
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 2.5
Smallest converged particle spacing average: 51.94182488825657
Smallest non-converged particle spacing: 20
Smallest non-converged particle spacing average: 28.92614444444848
Current Spacing = 1.25
Average = 52.46630478267171
Difference Value = 0.010097448357724581
Converged iteration = 3
Difference Value = 0.048758299097447055
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 20
Smallest non-converged particle spacing average: 28.92614444444848
Optimizing particle spacing.
Current Spacing = 15.0
Average = 40.19369150292728
Difference Value = 0.23391419179567924
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 15.0
Smallest non-converged particle spacing average: 40.19369150292728
Optimizing particle spacing.
Current Spacing = 12.5
Neutrino crashed with particle spacing. Need to try larger spacing.
Current Spacing = 13.0
Average = 31.525597249372368
Difference Value = 0.39912678470574375
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 13.0
Smallest non-converged particle spacing average: 31.525597249372368
Optimizing particle spacing.
Current Spacing = 11.5
Average = 45.965776551114146
Difference Value = 0.12389910550179487
```

```
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 11.5
Smallest non-converged particle spacing average: 45.965776551114146
Optimizing particle spacing.
Current Spacing = 10.75
Average = 33.99070661654211
Difference Value = 0.3521421651984079
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.75
Smallest non-converged particle spacing average: 33.99070661654211
Optimizing particle spacing.
Current Spacing = 10.375
Average = 38.39678925986842
Difference Value = 0.2681628824649015
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.375
Smallest non-converged particle spacing average: 38.39678925986842
Optimizing particle spacing.
Current Spacing = 10.1875
Average = 35.73323651315789
Difference Value = 0.3189298034009883
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.1875
Smallest non-converged particle spacing average: 35.73323651315789
Optimizing particle spacing.
Current Spacing = 10.09375
Average = 35.658342499999996
Difference Value = 0.3203572722015475
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.09375
Smallest non-converged particle spacing average: 35.658342499999996
Optimizing particle spacing.
Current Spacing = 10.046875
Average = 34.56162210526315
Difference Value = 0.3412606005239772
Did not converge
Largest converged particle spacing: 10.0
```

```
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.046875
Smallest non-converged particle spacing average: 34.56162210526315
Optimizing particle spacing.
Current Spacing = 10.0234375
Average = 33.75955476973684
Difference Value = 0.3565478851697831
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.0234375
Smallest non-converged particle spacing average: 33.75955476973684
Optimizing particle spacing.
Current Spacing = 10.01171875
Average = 35.51837601973684
Difference Value = 0.3230250125892674
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.01171875
Smallest non-converged particle spacing average: 35.51837601973684
Optimizing particle spacing.
Current Spacing = 10.005859375
Average = 32.69981694313948
Difference Value = 0.3767463312198156
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.005859375
Smallest non-converged particle spacing average: 32.69981694313948
Optimizing particle spacing.
Current Spacing = 10.0029296875
Average = 34.22728569078947
Difference Value = 0.34763300307564493
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.0029296875
Smallest non-converged particle spacing average: 34.22728569078947
Optimizing particle spacing.
Current Spacing = 10.00146484375
Average = 34.50530310855263
Difference Value = 0.3423340322616192
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
```

```
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.00146484375
Smallest non-converged particle spacing average: 34.50530310855263
Optimizing particle spacing.
Current Spacing = 10.000732421875
Average = 35.48910544407895
Difference Value = 0.32358290542694934
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.000732421875
Smallest non-converged particle spacing average: 35.48910544407895
Optimizing particle spacing.
Current Spacing = 10.0003662109375
Average = 33.61556036184211
Difference Value = 0.35929239726170936
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.0003662109375
Smallest non-converged particle spacing average: 33.61556036184211
Optimizing particle spacing.
Current Spacing = 10.00018310546875
Average = 35.966890838815786
Difference Value = 0.3144763865532467
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.00018310546875
Smallest non-converged particle spacing average: 35.966890838815786
Optimizing particle spacing.
Current Spacing = 10.000091552734375
Average = 37.72462282894736
Difference Value = 0.2809742751045268
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.000091552734375
Smallest non-converged particle spacing average: 37.72462282894736
Optimizing particle spacing.
Current Spacing = 10.000045776367188
Average = 33.126441611842104
Difference Value = 0.3686149282084961
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.000045776367188
```

```
Smallest non-converged particle spacing average: 33.126441611842104
Optimizing particle spacing.
Current Spacing = 10.000022888183594
Average = 34.242631249999995
Difference Value = 0.3473405189894474
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.000022888183594
Smallest non-converged particle spacing average: 34.242631249999995
Optimizing particle spacing.
Current Spacing = 10.000011444091797
Average = 36.22752074013158
Difference Value = 0.30950881922798174
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.000011444091797
Smallest non-converged particle spacing average: 36.22752074013158
Optimizing particle spacing.
Current Spacing = 10.000005722045898
Average = 33.52434375
Difference Value = 0.3610309723761556
Did not converge
Largest converged particle spacing: 10.0
Largest converged particle spacing average: 49.90813700154039
Smallest converged particle spacing: 1.25
Smallest converged particle spacing average: 52.46630478267171
Smallest non-converged particle spacing: 10.000005722045898
Smallest non-converged particle spacing average: 33.52434375
Maximum iterations was reached. Exit Optimizer.
The maximum converged particle spacing found was: 10.0
Total runtime: 5654.724431753159
```