

Theory, models, and numerical methods for classical and non-classical transport

Zur Erlangung des akademischen Grades eines

DOKTORS DER NATURWISSENSCHAFTEN

von der KIT-Fakultät für Mathematik des
Karlsruher Instituts für Technologie (KIT)
genehmigte

DISSERTATION

von

THOMAS CAMMINADY, M.Sc.

Tag der mündlichen Prüfung: 13.01.2021

Erstgutachter: Prof. Dr. Martin Frank

Zweitgutachterin: PD Dr. Gudrun Thäter

Drittgutachter: Prof. em. Edward Larsen, PhD

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

I declare that I have developed and written the enclosed thesis completely by myself, and have not used sources or means without declaration in the text.

.....
Thomas Camminady
Karlsruhe, 14.12.2020

Mathematics is not a deductive science—that’s a cliché. When you try to prove a theorem, you don’t just list the hypotheses, and then start to reason. What you do is trial and error, experimentation, guesswork. You want to find out what the facts are, and what you do is in that respect similar to what a laboratory technician does.

PAUL R. HALMOS, I Want to be a Mathematician

Running to him was real, the way he did it the realest thing he knew. It was all joy and woe, hard as diamond; it made him weary beyond comprehension. But it also made him free.

JOHN L. PARKER JR., Once a Runner

Acknowledgments

Throughout the last decade, Martin Frank has been my teacher and mentor. With his rare ability to combine enthusiasm, rigor, and ease, he sparked my interest in mathematics the moment I first entered his *Introductory Mathematics* lecture at RWTH Aachen University in 2009. Paraphrasing one of his favorite analogies, he has been the shepherd that guided me to the fruitful, green grass that is the field of kinetic theory. A uniquely approachable advisor, his constant guidance and the many insightful discussions have been integral to my academic and personal development. From the beginning, he introduced me to the scientific community and taught me to communicate concisely and clearly towards peers and non-peers alike. For all this and much more, I want to thank Martin Frank and express my gratefulness.

Edward Larsen is one of the people that Martin Frank introduced to me. Discussing non-classical particle transport in front of a blackboard back in Aachen is one of the highlights of my years as a Ph.D. student. His experience in and deep understanding of transport theory is all the more impressive, given that it was always paired with a welcoming openness towards input from everyone involved in a discussion—even from inexperienced students like me. Our many (written) conversations made me a more knowledgeable person and a better writer, too. I am immensely grateful for these experiences. Additionally, his many remarks helped to improve the quality of my work.

I want to thank Gudrun Thäter for her fruitful input, for reading and improving my thesis, and for being part of my Ph.D. committee. Her help made this a better, more understandable thesis.

My colleagues, both in Aachen and Karlsruhe, have been a constant source of help and support. The many fruitful discussions broadened my understanding of mathematics and science in general. At the same time, I was fortunate to call them my friends. In particular, I want to thank Kerstin Küpper and Birte Schmidtmann for making the time at MathCCES unforgettable; Maren Hattebuhr, Sarah Schönbrodt, Maike Sube, and Kirsten Wohak for teaching me the importance of making mathematics accessible to a diverse audience and future generations; and Jonas Kusch and Jannick Wolters for helping me overcome many challenging problems and always being there to answer questions about kinetic theory. I will miss our coffee breaks.

I also want to thank everyone I have ever run a single mile with. Be it with my friends from the Aachener TG, the LG Schanz, or the KIT SC, a solid run with good company has never failed to brighten my day. Additionally, Wolfgang Glöde taught me many lessons about mental and physical strength. I wouldn't be where I am today without the balance that running—and the many, many people I met along the way—provided.

My family's never-ending support and love helped me overcome many stressful moments throughout my life. They have always been there for me, regardless of the paths I took or the decisions I made. I couldn't wish for better parents.

Lastly, my deepest gratitude goes to Nayana Karkanematt Jagadish. Both her joyful nature and motivating words have been invaluable.

Preface

The linear transport equation is an integro-differential equation that describes the evolution of the angular flux ψ . Depending on time t , spatial position \mathbf{x} , speed v , and direction of flight Ω , the angular flux is defined as $\psi(t, \mathbf{x}, \Omega) := v N(t, \mathbf{x}, \Omega)$ where $N(t, \mathbf{x}, \Omega) d\mathbf{x} d\Omega$ represents the number of particles inside the infinitesimal phase space element $[\mathbf{x}, \mathbf{x} + d\mathbf{x}] \times [\Omega, \Omega + d\Omega]$ at time t . Including information on the angular direction but not on each particle individually, the transport equation models particle systems from a *mesoscopic* viewpoint; poised between a *microscopic* description where the positions and velocities of all particles are known, and a *macroscopic* description that only uses moments of the angular flux to describe the dynamics.

Even in its simplest form, the transport equation describes the evolution of a six-dimensional quantity and, unsurprisingly, analytical solutions to non-trivial applications are scarce. One goal of this thesis is therefore to increase the accuracy and performance of numerical algorithms that approximate the transport equation's solution—specifically of the discrete ordinates (S_N) method.

The S_N method reduces the high dimensionality of the problem by discretizing the angular variable and restricting transport to a finite number of ordinates. However, the resulting system of coupled advection equations suffers from a numerical artifact called ray effects. These undesirable oscillations of the particle density throughout the spatial domain can impact the solution quality dramatically. In Chapter 2, this thesis presents and analyzes two variations of the S_N method, called rS_N method [21] and $as-S_N$ method [37]. We will see that both methods can mitigate ray effects significantly, once by adding a rotation-and-interpolation step to the S_N method (rS_N), and once by adding artificial scattering in form of a carefully chosen scattering operator to the transport equation's right-hand side ($as-S_N$).

Similar to S_N , rS_N and $as-S_N$ require proper angular quadrature points to produce satisfactory results. Here, we present spherical quadrature sets that are based on triangulating Platonic solids, resulting in a highly uniform distribution of points on the unit sphere, a low variance in the respective quadrature weights, and an underlying connectivity that allows the interpolation of function values at arbitrary points that are not included in the quadrature set [22].

After further discretizing the system of S_N equations in both space and time in a way that respects initial and boundary conditions, solution algorithms need to solve the fully discretized system. Transport sweeps, for example, can be used to update the state in every spatial cell at every time step. We provide a proof, stating that this procedure is always possible for two-dimensional domains that are discretized by triangles.

Another significant portion of this thesis is concerned with the analysis of non-classical transport. Under certain assumptions, classical transport theory fails to model the particle system's dynamics correctly. A new, so-called non-classical transport equation augments the phase space of the classical transport equation by an additional variable. Quantities that were equivalent in classical transport—e.g., the distribution of distances *to* the next collision and the distribution of distances *from* the last collision—now need to be distinguished carefully.

Moreover, non-classical transport in the presence of heterogeneities has only recently been investigated [20]. Heterogeneous versions of non-classical cross sections and path length distributions are provided that model the effect of domain interfaces along particles' trajectories.

Lastly, the notion of *correlated* and *uncorrelated* particles [33] is extended to include heterogeneities.

This thesis relies, partially, on work that has previously been published with coauthors in journals or presented at conferences and included in the respective proceedings. Especially the work on ray effect mitigation techniques is noteworthy, since it has subsequently been included in the Ph.D. thesis of Kusch [55]. The following two paragraphs serve to clarify my contributions to the rS_N and $as-S_N$ method.

The rS_N method was published in Camminady et al. [21]. I developed and implemented the quadrature sets and implemented the rotation-and-interpolation subroutine. Additionally, I reimplemented large parts of an explicit version of the S_N algorithm in the Julia programming language. Using this code, I analyzed the influence of the rotation magnitude (with varying numbers of quadrature points) on the line-source and the lattice test case. The idea and subsequent realization of the rS_N method (mitigating ray effects by adding a rotation-and-interpolation step) has been developed by all authors collaboratively.

The $as-S_N$ method was published in Frank et al. [37]. I significantly contributed to the analysis and implementation of the artificial scattering operator and performed the related asymptotic analysis. The code that was used to generate the numerical results is largely based on the rS_N code, except for the implicit time integration, which was added by a coauthor. I performed numerical experiments that resulted in optimal parameter configurations for the explicit time discretization. The core idea of the $as-S_N$ method (mitigating ray effects by adding artificial scattering) has been initiated by a coauthor.

Contents

Preface	ix
List of figures	xv
List of tables	xvii
List of algorithms	xix
I. Classical transport	1
1. Transport equations	3
1.1. Microscopic viewpoint	4
1.2. Liouville equation	7
1.3. Boltzmann-Grad limit	8
1.4. Linear transport equation	9
1.4.1. Properties of the collision kernel and operator	10
1.4.2. Entropy for the linear transport equation	12
1.4.3. Initial and boundary conditions	13
1.4.4. Stationary transport	13
1.4.5. Slab geometry	14
1.5. Neutron transport	14
1.5.1. Criticality calculations	16
1.6. Asymptotic limits	16
1.7. Hyperbolic conservation laws	19
2. Numerical solution methods	23
2.1. Monte Carlo method	23
2.1.1. Random numbers	24
2.1.1.1. Inverse transform sampling	25
2.1.1.2. Rejection sampling	25
2.1.1.3. Monte Carlo as a numerical integrator	27
2.1.2. Monte Carlo for transport equations	28
2.1.3. Advantages and disadvantages of the Monte Carlo method	29
2.2. Discrete ordinates method	29
2.2.1. Angular discretization	30
2.2.2. Quadrature rules in one dimension	32
2.2.3. Spherical quadrature rules	33
2.2.3.1. Comparison and source code	38
2.2.4. Finite volume schemes	40
2.2.5. Source iteration and transport sweeps	41
2.2.5.1. Source iteration for time dependent problems	47

2.2.6. Advantages and disadvantages of the discrete ordinates method	47
2.3. Moment methods	49
2.3.1. P_N method	53
2.3.2. Advantages and disadvantages of moment methods	54
3. Ray effects and their mitigation	55
3.1. Understanding ray effects	55
3.2. rS_N method	59
3.2.1. rS_N and the line-source test case	63
3.2.2. rS_N and the lattice test case	64
3.2.3. rS_N without change of ordinates	70
3.3. $as-S_N$ method	70
3.3.1. Angular discretization	72
3.3.2. Modified equation and asymptotic analysis	72
3.3.3. $as-S_N$ and the line-source test case	76
3.3.4. $as-S_N$ and the lattice test case	80
II. Non-classical transport	83
4. Non-classical transport equations	85
4.1. Particle billiard redux	88
4.2. Generalized linear Boltzmann equations	92
4.3. Correlated and uncorrelated particles	95
5. Non-classical transport in heterogeneous materials	99
5.1. The crux of particles crossing material interfaces	99
5.2. Defining heterogeneity	101
5.3. Memory-resetting ansatz	103
5.4. Memory-preserving ansatz	107
5.4.1. Heterogeneity via density fluctuations	107
5.4.2. True heterogeneity	108
5.5. Sampling path lengths	112
5.5.1. The general case	113
5.5.2. Sampling in piecewise homogeneous materials	113
5.6. Correlated and uncorrelated particles with heterogeneities	115
III. Concluding remarks	119
6. Summary	121
7. Outlook	125
Bibliography	127

List of figures

1.1. The white ball undergoes elastic collisions on the billiard table while the black balls' positions are fixed.	3
1.2. The collision tube for a particle that underwent a collision at (x_n, y_n) moving to the right (post-collision). After a distance δt_n , the next collision takes place with the obstacle at (x_i, y_i)	5
1.3. Trajectory of a particle with $r = 0$ in an obstacle field. The domain is periodic and obstacles were allowed to overlap. The picture shows 180 obstacles with $\sigma_t = 8$. The white line overlaps with the obstacles only as a result of plotting with a non-zero linewidth.	6
1.4. A distribution for the distances between consecutive collisions with $\sigma_t = 4$, 10000 obstacles, and 100000 traced trajectories. Additionally to the histogram, we have a best fit with ansatz $\lambda \exp(-\lambda s)$. The best λ is close to the expected value $\sigma_t = 4$. Statistical noise becomes visible for long distances.	7
1.5. The Henyey-Greenstein scattering kernel $\log_{10} s_g(\cos(\theta))$ for different values of g with $\cos \theta = \Omega \cdot \Omega'$. The radial component is the \log_{10} -probability of scattering into the corresponding angle (relative to the pre-scattering direction). For $g = 0$ (orange), particles scatter isotropically. For $g = 1 - \varepsilon$ with $0 < \varepsilon \ll 1$, particles mostly scatter into the pre-scattering direction (green).	11
1.6. Spatial domain $V \subsetneq \mathbb{R}^3$ with boundary ∂V and outward facing normal \mathbf{n}	13
1.7. Slab geometry for two-dimensional transport. Transport is projected onto the x -axis.	14
1.8. Linear advection equation with $\rho_0(x) = \exp(-x^2)$	20
1.9. Burgers's equation with initial data $u_0(x)$ shown in green.	20
1.10. Evolution of an initial distribution $\psi_0(\mathbf{x}, \Omega)$ through space along a fixed direction Ω with constant σ_a . Each <i>bump</i> represents the solution at a different time step.	21
2.1. To generate samples from f_X via rejection sampling, we use $3 \cdot f_Y$ as an upper bound. Here, f_Y is a normal distribution with mean $\mu = 0$ and variance $\sigma^2 = 9$	25
2.2. Output of the rejection sampling algorithm. Purple points are being rejected. For the orange points, the corresponding x values are distributed according to f_X , illustrated by the orange dashes at the bottom. We observe that many points are being rejected, indicating that f_Y is no optimal choice.	26
2.3. For a fixed quadrature order N , the level-symmetric quadrature computes the integral exactly (up to machine precision) if $i + k \leq N$	34
2.4. The level-symmetric quadrature becomes more accurate the higher the number of quadrature points gets. Eventually, the integral is computed exactly (up to machine precision). The number of quadrature points grows quadratically with the order.	35
2.5. For a fixed quadrature order N , the level-symmetric quadrature computes the integral exactly (up to machine precision) if $i + k \leq 2N - 1$	35

2.6. The spread of the quadrature weights, i.e., the ratio between the maximal weight and the minimal weight, diverges for the Gauss-Legendre quadrature. This also holds true in the three-dimensional case, when using a tensorized quadrature that is based on the Gauss-Legendre points and weights. Nevertheless, the Gauss-Legendre quadrature's capability to satisfy the fifth point on the list of desirable properties makes it one of the most commonly used quadratures.	36
2.7. Generation of the quadrature sets. One face of the octahedron and icosahedron have been further refined. Vertices will be projected onto the unit sphere and taken as quadrature points.	37
2.8. Triangulation for order = 4 with the corresponding quadrature points in the plane and on the surface of the unit sphere, together with the connectivity and the quadrature weight for the center point in green. Quadrature points and weights for the seven other octants result from symmetry. Figure 2.8a demonstrates the lerp procedure. For the slerp quadratures, the points in Figure 2.8b would have to be spaced equally.	38
2.9. Octahedron and icosahedron quadrature. Due to the implementation of the spherical linear interpolation, the slerp versions of the quadratures have less symmetry than the lerp versions. They are, however, not as spread in their weight distribution. The smallest quadrature weights are associated with the vertices of the respective platonic solids. Since their quadrature weights are significantly lower, they are excluded from the colorbar and highlighted in white to avoid distortion.	39
2.10. The spread of the quadrature weights, i.e., the ratio between the maximal weight and the minimal weight, diverges for the octahedron and icosahedron quadratures. Note the gap in the y -axis, inserted to make it possible to include all four graphs.	40
2.11. Two possible situations for sweeping through a domain of triangles: There are either two inflow edges (green) and one outflow edge (blue), or one inflow edge and two outflow edges.	42
2.12. We augment the situation by one additional variable. For triangles with two inflow edges, we solve for flux values at the two outflow vertices. These two outflow vertices are inflow vertices for the adjacent triangle. Known variables are green, unknown variables blue.	43
2.13. For a given spatial discretization, a direction Ω induces a directed dependency graph $G = (V, E)$. A directed edge $e_{i,j} \in E$ from V_i to V_j implies that for computations to start in cell C_j , we must have finished computations in cell C_i	45
2.14. For sweeping to work, there can not be a circular dependency in the mesh.	46
2.15. Sketches for the proof of Theorem 2.2.3.	46
2.16. Quadrilaterals can be arranged in a circle such that no topological ordering can be obtained for $\Omega = (0, 0, -1)^T$	47
2.17. Ray effects for the line-source problem occur due to the finite set of angular directions.	49
2.18. Real parts of the spherical harmonics for different orders.	51
2.19. Imaginary parts of the spherical harmonics for different orders.	52
3.1. Infinite medium test case with a circular source of radius $r = 0.4$ at $(0, 0)$. The domain is purely absorbing with $\sigma_a = 1$. We evaluate the angular and scalar flux along the black line at $x = 1$ and take a closer look at the three points $(1, 0.3)$, $(1, 0.15)$, and $(1, 0)$ in orange, purple, and blue, respectively. Scene drawn to scale.	56

3.2. Angular and scalar flux with $n_q = 11$ for the test case presented in Figure 3.1.	56
3.3. Angular and scalar flux with $n_q = 21$ for the test case presented in Figure 3.1.	56
3.4. For the line-source test case, particles are released isotropically at $t = 0$ along every point on the z -axis (depicted here for a finite number of points).	57
3.5. The two contributions that make up the scalar flux $\phi^G(1, \mathbf{x})$ for the line-source test case using the exact (up to numerical integration) angular flux and 36 ordinates.	59
3.6. Comparison between the scalar flux $\phi^G(1, \mathbf{x})$, based upon the exact angular flux, and the scalar flux that is the result of an S_6 computation.	59
3.7. Planar barycentric interpolation works by using the areas A_0, A_1 , and A_2 as interpolation weights to obtain a new function value at P'_0 based upon the function values at P_0, P_1 , and P_2 . In this example, the point P'_0 is the result of shifting P_0 by $\varepsilon \cdot \Omega$. For the spherical case, the corresponding transformation would be the application of the rotation matrix to point P_0	61
3.8. Each quadrature point is assigned a value of $\psi = 1.0$ if it is positioned in the first octant and $\psi = 0.0$ otherwise. The associated area is colored accordingly. The rotation and interpolation procedure around a random axis is then applied 5, 10, and 20 times, respectively. In each time step, we rotate forth and back around the same axis by the same magnitude to keep the orientation of the quadrature points. The diffusive behavior is clearly visible.	62
3.12. Layout of the lattice test case.	64
3.9. Scalar flux for the line-source test case at $t = 1$ second using the octahedron (lerp) quadrature. The colorbar is omitted but identical to that in Figure 2.17a. The spatial domain is $[-1.5 \text{ cm}, 1.5 \text{ cm}]^2$ with $n_x = n_y = 100$. Rows vary in n_q and columns vary in δ , respectively.	66
3.10. Scalar flux for the line-source test case at $t = 1$ second using the octahedron (lerp) quadrature. The solution is visualized along three cuts: Horizontally (blue), vertically (purple), and diagonally (orange). The reference solution is given in black. The spatial domain is $[-1.5 \text{ cm}, 1.5 \text{ cm}]^2$ with $n_x = n_y = 100$. Rows vary in n_q and columns vary in δ , respectively.	67
3.11. Scalar flux for the line-source test case at $t = 1$ second using the octahedron (lerp) quadrature. The solution is visualized along four concentric circles: $r = 1.0$ (blue), $r = 0.9$ (orange), $r = 0.6$ (purple), and $r = 0.2$ (green). The reference solutions are given in black for all radii. The spatial domain is $[-1.5 \text{ cm}, 1.5 \text{ cm}]^2$ with $n_x = n_y = 100$. Rows vary in n_q and columns vary in δ , respectively.	68
3.13. Scalar flux (\log_{10}) for the lattice test case at $t = 3.2$ seconds using the octahedron (lerp) quadrature. The spatial domain is $[0 \text{ cm}, 7 \text{ cm}]^2$ with $n_x = n_y = 280$. Rows vary in n_q and columns vary in δ , respectively. Isolines are drawn to highlight the values -5 (white), -4 (gray), and -3 (black).	69
3.14. Comparison of two ways to implement the rS_N method.	70
3.15. For the as- S_N method, two parameters (σ_{as} and β) need to be chosen. All three images show line-source cuts with $n_q \times n_x \times n_y = 92 \times 200 \times 200$. Left, artificial scattering is not used. Center, $\sigma_{\text{as}} = 4.0$ and $\beta = 8.0$, which corresponds to the optimal parameter found in the parameter study in Figure 3.17 on the same grid. Right, $\sigma_{\text{as}} = 18.0$ and $\beta = 3.5$, which are the optimal parameters from the coarse parameter study ($n_q \times n_x \times n_y = 12 \times 50 \times 50$).	77

3.16. Parameter study for σ_{as} and β on the grid $n_q \times n_x \times n_y = 12 \times 50 \times 50$ in an explicit calculation. For every set of parameters, we compute the L^2 error of the scalar flux ϕ with respect to the semi-analytical reference solution on the same spatial grid. The number in each field of the heatmap is then the baseline normalized error, i.e., the L^2 error obtained for that specific parameter configuration divided by the error obtained without artificial scattering. The optimal parameter configuration, highlighted in yellow, reduces the error down to 43%. Close-to-optimal parameter pairs are highlighted in green.	78
3.17. Parameter study for σ_{as} and β on the grid $n_q \times n_x \times n_y = 92 \times 200 \times 200$ in an explicit calculation. For every set of parameters, we compute the L^2 error of the scalar flux ϕ with respect to the semi-analytical reference solution on the same spatial grid. The number in each field of the heatmap is then the baseline normalized error, i.e., the L^2 error obtained for that specific parameter configuration divided by the error obtained without artificial scattering. The optimal parameter configuration, highlighted in yellow, reduces the error down to 23%. Close-to-optimal parameter pairs are highlighted in green.	79
3.18. Results for the lattice test case with different amounts of artificial scattering. From left to right: A high-order reference solution; the best-case scenario; the baseline; and the worst-case scenario.	80
3.19. Parameter study for σ_{as} and β on the grid $n_q \times n_x \times n_y = 42 \times 280 \times 280$ in an explicit lattice calculation. For every set of parameters, we compute the L^2 error of the scalar flux ϕ with respect to a reference solution ($n_q = 1962$) on the same spatial grid. The number in each field of the heatmap is then the baseline normalized error, i.e., the L^2 error obtained for that specific parameter configuration divided by the error obtained without artificial scattering. The optimal parameter configuration, highlighted in yellow, reduces the error down to 69%. Close-to-optimal parameter pairs are highlighted in green. The colorbar ranges to a maximum value of 3.00 to avoid color distortion.	81
4.1. For the month of July 2020, the image shows the averaged cloud fraction with a resolution of 0.25 degrees. The data was recorded with the Aqua/MODIS satellite by NASA Earth Observations. This visualization reconstructs the image by Reto Stockil, hosted on the NASA website [34] (together with the corresponding data).	86
4.2. A cloud rendered with a non-exponentially decaying path length distribution. Light penetrates deeper than in the case of exponentially decaying path lengths. This image was provided by Bitterli and a side-by-side comparison of classical and non-classical renders can be found in Bitterli et al. [14]. Heterogeneity is introduced into the simulation by assuming a statistically heterogeneous material with identical correlation, similar to Camminady et al. [20].	87
4.3. A non-classical billiard game where scatterers' centers are arranged according to a lattice structure. This setup is also known under the name <i>periodic Lorentz gas</i> . The (solid) white ball's trajectory is shown.	89
4.4. Path length distributions in the periodic Lorentz gas with $\sigma = 1.0$ and 10^6 obstacles, which results in a radius of $2 \cdot 10^{-6}$ for each obstacle. The lattice gap—the minimal distance between obstacles' boundaries—is 0.000996. The first two plots use 100 log-spaced bins, the last plot uses 30.	90

4.5. Trajectory of 2000 particles from their birth (at a random position, not touching an obstacle) to their first collision. For all three images, $n_{\text{obs}} \cdot r$ is constant. The orientation of the lattice fades as we increase the number of obstacles, visible in the trajectories. The domains are periodic.	91
4.6. Scatterers of radius $r = 0.01$ generated in the domain $[0, 1] \times [0, 1]$. Both pictures show 696 scatterers.	95
4.7. Life cycle of correlated and uncorrelated particles.	96
5.1. Two materials with cross sections $\sigma_1(s)$ and $\sigma_2(s)$, and a particle traversing from one side of the interface to the other. The material extends to the right indefinitely.	100
5.2. The path length distribution and cross section are the outcome of the Boltzmann-Grad limit. This involves an ensemble average over all possible configurations of obstacles C_μ , even for the periodic Lorentz gas. For example, obstacle positions could be translated or rotated. This holds true for other arrangements as well, e.g., Poisson disk sampling, where obstacles are at least a prescribed minimum distance apart from another. The subscript μ denotes a singular instance out of all possible arrangements and $\langle \cdot \rangle_\mu$ is the respective ensemble average.	102
5.3. Material heterogeneities can be the result of density fluctuations (Figure 5.3a), or structural differences (Figure 5.3b).	103
5.4. A particle traversing multiple materials with different cross sections $\sigma_i(s)$ and related density functions $p_i(s)$ with interfaces located distances s_i away from its origin. The cross sections and density functions are to be interpreted in the homogeneous sense, i.e., if the particle would be in a homogeneous, infinite domain of Material i , it would experience cross section $\sigma_i(s)$ and density function $p_i(s)$	103
5.5. Different materials defined by their density functions and cross sections. Here, p_1 and p_2 are exponential distributions, p_4 is a gamma distribution, and p_4 is a Pareto distribution. Cross sections are computed with (4.14).	105
5.6. Combined path length distribution for a domain with two materials that are identical; both have gamma distributed path lengths.	106
5.7. Distribution and cross section for a domain composed of three materials: Material 1 is defined by a gamma distribution, Material 2 by an exponential distribution, and Material 3 by a Pareto distribution.	106
5.8. Path length distributions for a heterogeneous domain with a single interface and varying base distribution. The different lines show the distributions for different density fluctuations.	109
5.9. Life cycle of correlated and uncorrelated particles in the heterogeneous case.	115
5.10. An infinite, heterogeneous domain with an interface at $x = 0$	116

List of tables

2.1. Table of weight functions and corresponding orthogonal polynomials.	32
3.1. Cross sections and source for the lattice test case.	64
4.1. Juxtaposition of the forward and backward formulation.	94
5.1. Cross sections and distribution functions for the materials in Figure 5.5.	105
5.2. Requirement (ii) is satisfied for all probability density functions, showcased for those functions that we have already encountered. For each one, we observe that the general definition equals the conversion. The \mathbf{x} -dependency of ρ was only omitted to shorten notation.	107

List of algorithms

2.1. Rejection sampling	26
3.1. The S_N method	60
3.2. The rS_N method	60
3.3. The rotation-and-interpolation step	62
5.1. Generating a single sample in the general case	113
5.2. Generating multiple samples in the general case	113
5.3. Generating a single sample in the piecewise homogeneous case (without void) . .	114

Part I.

Classical transport

1. Transport equations

Imagine an infinitely large billiard table. A particle—represented by the white billiard ball—moves frictionlessly along straight lines until undergoing elastic, instantaneous collisions with background obstacles—the black billiard balls—that do not move due to their infinite weight and neglectable velocity. The situation is depicted in Figure 1.1.

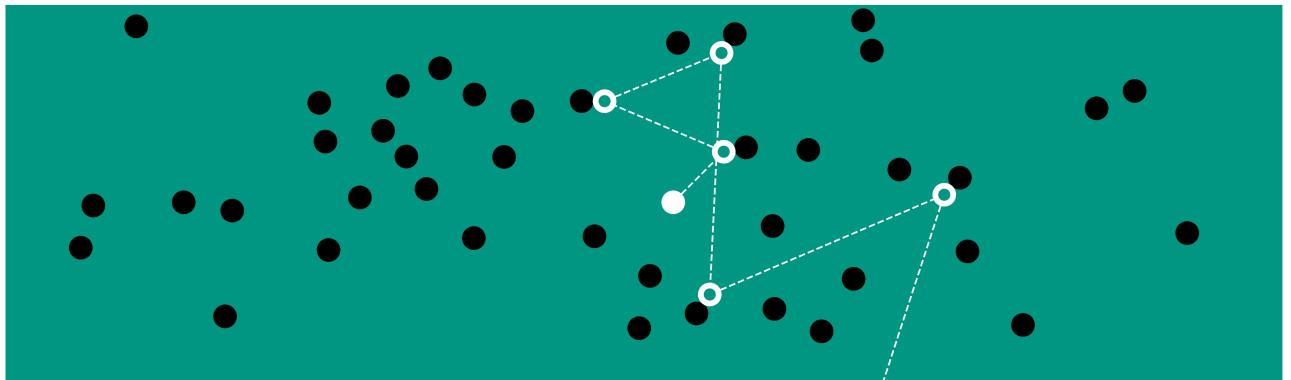


Figure 1.1.: The white ball undergoes elastic collisions on the billiard table while the black balls' positions are fixed.

This in many ways simplified microscopic description of particle interactions will serve as the starting point for this thesis. Despite its simplicity, it allows the derivation of governing mesoscopic and macroscopic equations, as well as the construction of one of the most prominent numerical methods in transport theory—the Monte Carlo method. Furthermore, changing the microscopic picture will motivate non-classical transport, discussed in the second part of this thesis.

A magnitude of real-world phenomena and topics can be formulated in the language of kinetic theory. These include, but are not limited to, the theory of gases; radiation therapy for cancer treatment; modeling of nuclear reactors; or illumination in movies and computer games.

Transport equations try to achieve the following: Instead of describing a system by the behavior of every single molecule, every single particle, or every single light-ray, they seek a description in terms of statistical quantities that evolve in time. This is both necessary and often desirable. Necessary, since it already becomes computationally impossible to evolve the trajectory of each of the roughly $6.022 \cdot 10^{23}$ particles in one mole. Desirable, since it is generally of no interest to know each of these trajectories individually. Thus, transport equations are convenient tools that distill complex physical systems to manageable equations.

Throughout the scope of this thesis we are going to consider uncharged particle transport; that is, particles do not interact with another via long range interactions—like electrons do—but with the background medium or through direct collisions with another. Given these assumptions,

particles move through phase space according to Newton's laws of motion

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{v}(t), \\ \dot{\mathbf{v}}(t) = \frac{1}{m} \mathbf{F}(t, \mathbf{x}(t), \mathbf{v}(t)), \end{cases} \quad (1.1)$$

with $\mathbf{x}(t), \mathbf{v}(t) \in \mathbb{R}^3$, $t \in \mathbb{R}^{\geq 0}$, $m \in \mathbb{R}^{>0}$, $\mathbf{F} : \mathbb{R}^{\geq 0} \times \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^3$ sufficiently smooth, and subject to initial conditions $\mathbf{x}(0) = \mathbf{x}_0$ and $\mathbf{v}(0) = \mathbf{v}_0$. If we again assume that particles move with a much smaller mass and faster than background obstacles, and if we also neglect particle-particle interactions, it is easy to track particle trajectories through the obstacle field as already seen in Figure 1.1.

If the number of particles becomes large, tracking every single trajectory is infeasible and, more importantly, not necessary since the distribution of particles in phase space is usually sufficient information. Transitioning from the microscopic viewpoint to the mesoscopic viewpoint is rigorously formalized in the Boltzmann-Grad limit [43] and will be explained in a following section. From it follows the definition of the particle number $N : \mathbb{R}^{\geq 0} \times \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^{\geq 0}$. Given an incremental volume $d\mathbf{x}$ and a normalized incremental angle $d\mathbf{v}$, we define

$$N(t, \mathbf{x}, \mathbf{v}) d\mathbf{x} d\mathbf{v} := \text{"the expected number of particles in a phase space element } d\mathbf{x} d\mathbf{v} \text{ about } \mathbf{x} \text{ and } \mathbf{v} \text{ at time } t." \quad (1.2)$$

With this fundamental variable, we can define the angular flux $\psi : \mathbb{R}^{\geq 0} \times \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^{\geq 0}$ by $\psi(t, \mathbf{x}, \mathbf{v}) = v N(t, \mathbf{x}, \mathbf{v})$, where v is the particle's speed and \mathbf{v} the particle's direction of flight. According to Prinja and Larsen [82], we can interpret this quantity as

$$\psi(t, \mathbf{x}, \mathbf{v}) d\mathbf{v} dS := \text{"the absolute rate at which particles in } d\mathbf{v} \text{ about } \mathbf{v} \text{ travel through the surface increment } dS \text{ orthogonal to } \mathbf{v} \text{ at time } t." \quad (1.3)$$

Knowing the angular flux, we can derive moments by integrating ψ against \mathbf{v}^i over the space of possible velocities. For example, the zeroth-order moment is the scalar flux

$$\phi(t, \mathbf{x}) := \int_{\mathbb{R}^3} \psi(t, \mathbf{x}, \mathbf{v}) d\mathbf{v}, \quad (1.4)$$

commonly referred to as the (particle) *density*. We are now going to discuss the governing equations that describe how the angular flux evolves in time, given an initial distribution of particles and varying assumptions about the underlying physical context.

Parts of the following sections are inspired by the classical books of Chandrasekhar [29] and Lewis and Miller [64]; the lecture notes of Frank [36]; and a collection of lecture notes, edited by Bellomo [12].

1.1. Microscopic viewpoint

Before discussing transport problems from the mesoscopic perspective, the microscopic viewpoint deserves further consideration. Let $\mathcal{C}_r := \{\mathbf{c}_i = (x_i, y_i)\}_{i=1, \dots, N}$ be the set of N circles, centered at positions (x_i, y_i) , each with radius r . For any prescribed initial position $\mathbf{x}_0 = (x_0, y_0)$ and unit velocity $\mathbf{v}_0 = (v_x, v_y)$, the trajectory of a particle with radius r is unambiguous. Let t_n

denote the time—and therefore the distance, because the particle travels with unit speed—until undergoing the n -th collision. Furthermore, let $i(n)$ denote the index of the obstacle that causes collision n . With $\mathbf{x}(t_n^-)$ and $\mathbf{v}(t_n^-)$ as the pre-collision position and velocity, the post-collision position and velocity are given by

$$\mathbf{x}(t_n^+) = \mathbf{x}(t_n^-), \quad (1.5a)$$

$$\mathbf{v}(t_n^+) = \mathbf{T}(\mathbf{c}_{i(n)}, \mathbf{x}(t_n^-), \mathbf{v}(t_n^-)). \quad (1.5b)$$

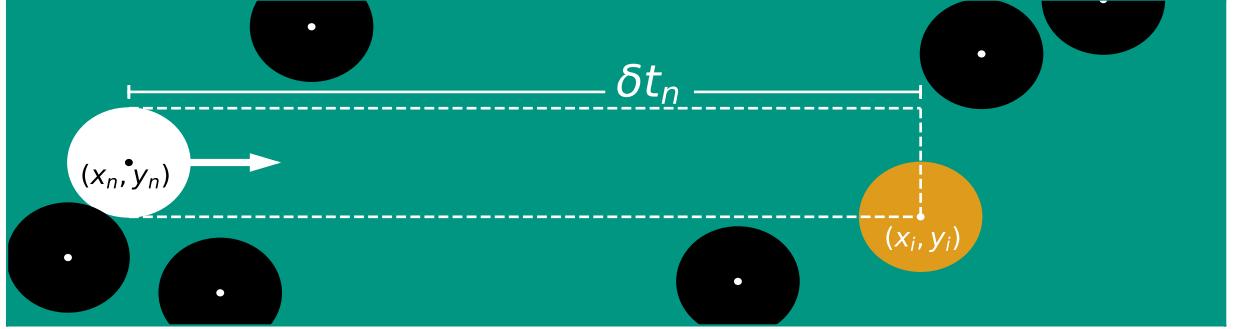


Figure 1.2.: The collision tube for a particle that underwent a collision at (x_n, y_n) moving to the right (post-collision). After a distance δt_n , the next collision takes place with the obstacle at (x_i, y_i) .

The operator \mathbf{T} results from geometric considerations and simply translates incoming into outgoing velocities in compliance with Newtonian mechanics. Consequently, $\delta t_n := t_n - t_{n-1}$ is the time (or equivalently distance) between collisions. The distribution of the δt_n follows from a geometric argument, sketched in Figure 1.2: For a particle to travel distance δt_n between consecutive collisions, there can not be an obstacle with position \mathbf{c}_i that overlaps with the tube of width $2r$ and length δt_n (shown in Figure 1.2 with dashed lines). For a reference domain of area A with N obstacles, the likelihood of *not* finding the center of an obstacle in a tube of length δt and width $4r$ (i.e., we shrink the obstacles to points and double the radius of the particle) is given by

$$Q_{r,A,N}(\delta t) = \left(1 - \frac{4r\delta t}{A}\right)^N. \quad (1.6)$$

Define $4r \cdot N/A = \sigma_t$ as the total cross section—the dimensionless number that represents the likelihood to scatter—to get

$$Q_{r,A,N}(\delta t) = \left(1 - \frac{\sigma_t \delta t}{N}\right)^N. \quad (1.7)$$

For a fixed domain size A we can let $N \rightarrow \infty$ and $r \rightarrow 0$ such that σ_t is constant to get

$$Q_{\sigma_t}(\delta t) = e^{-\sigma_t \delta t}. \quad (1.8)$$

With $Q_{\sigma_t}(\delta t)$ as the probability of particles *not* colliding before traveling a distance δt , $1 - Q_{\sigma_t}(\delta t)$ is the probability of particles traveling *at most* a distance δt . Differentiating $1 - Q_{\sigma_t}(\delta t)$, we

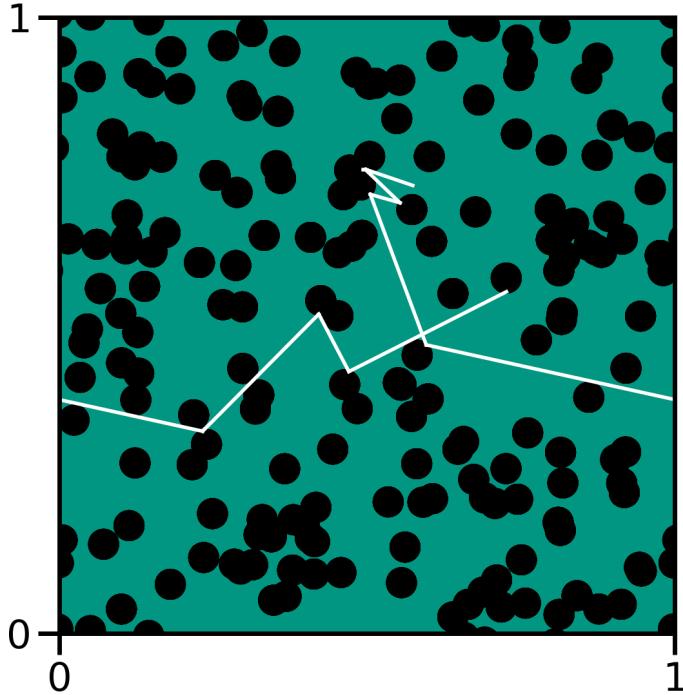


Figure 1.3.: Trajectory of a particle with $r = 0$ in an obstacle field. The domain is periodic and obstacles were allowed to overlap. The picture shows 180 obstacles with $\sigma_t = 8$. The white line overlaps with the obstacles only as a result of plotting with a non-zero linewidth.

define the path length distribution $p_{\sigma_t}(\delta t)$ as

$$\begin{aligned}
p_{\sigma_t}(\delta t) &:= \text{"the probability that a particle will travel a distance } \delta t \\
&\text{between consecutive collisions in a random obstacle field with} \\
&\text{total cross section } \sigma_t" \\
&= \sigma_t e^{-\sigma_t \delta t}.
\end{aligned} \tag{1.9}$$

Another physical interpretation of the total cross section σ_t is given by its inverse relation to the mean free path (MFP), i.e., the mean distance that particles can travel in the obstacle field of Figure 1.1. This can be confirmed numerically. The software PLE.jl¹ allows, among other things, to efficiently trace particles through an ensemble of obstacles, interacting with the obstacles on the basis of Newtonian mechanics. Consider therefore a random ensemble of obstacles with centers in $[0, 1] \times [0, 1]$. A sample simulation of the first ten collisions of a single particle in a periodic domain is visualized in Figure 1.3. When we increase the number of obstacles and simultaneously shrink their radii, the theoretical results of the Boltzmann-Grad limit can be reproduced numerically. Figure 1.4 summarizes the trajectory of a particle over 100000 collisions in a field of 10000 obstacles with $\sigma_t = 4$. Together with the data (green) we show the best fit with ansatz $\lambda \exp(-\lambda s)$. As predicted by (1.9), the best fit λ is reasonably close to σ_t .

¹PLE.jl, a path length estimator that I implemented in the Julia programming language, is available under the MIT license at <https://github.com/camminady/PLE.jl>.

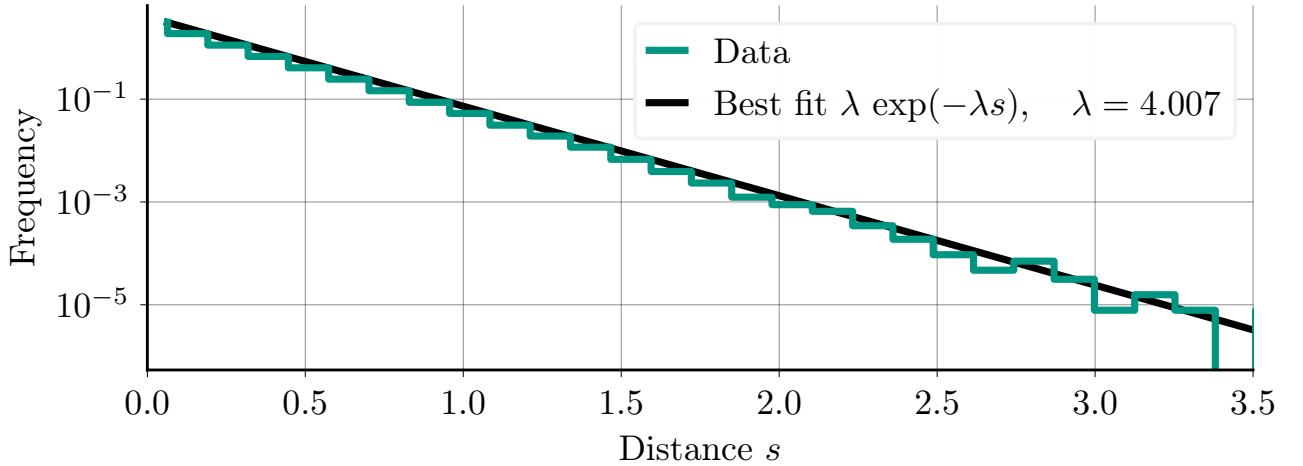


Figure 1.4.: A distribution for the distances between consecutive collisions with $\sigma_t = 4$, 10000 obstacles, and 100000 traced trajectories. Additionally to the histogram, we have a best fit with ansatz $\lambda \exp(-\lambda s)$. The best λ is close to the expected value $\sigma_t = 4$. Statistical noise becomes visible for long distances.

1.2. Liouville equation

We may ask ourselves why we would wish to describe a fully deterministic system—i.e., a system following Newton’s laws as described in (1.1)—in a probabilistic way—i.e., using the definition from (1.4). Clearly, the evolution of the distribution $\psi(t, \mathbf{x}, \mathbf{v})$ should follow the same deterministic dynamics as with the microscopic viewpoint; otherwise the evolution of $\psi(t, \mathbf{x}, \mathbf{v})$ in time would not relate to the evolution of the particle billiard at all. However, a compelling argument for the statistical description of the otherwise deterministic processes is its inherent uncertainty. Not only does quantum physics imply uncertainty in the initial data, but the sheer impracticality of knowing every particle’s position and velocity prompts an argument for a probabilistic approach.

The partial differential equation (PDE) that describes the evolution of $\psi(t, \mathbf{x}, \mathbf{v})$ in time is called Liouville’s equation. Consider a system of particles described by (1.1). Let us define ψ by the empirical measure as

$$\psi(t, \mathbf{x}, \mathbf{v}) := \delta(\mathbf{x} - \mathbf{x}(t))\delta(\mathbf{v} - \mathbf{v}(t)), \quad (1.10)$$

with δ as the Dirac delta. If we now multiply with a test function $u : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}$, integrate over $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{v} \in \mathbb{R}^3$, and differentiate with respect to t , we obtain

$$\begin{aligned}
& \frac{d}{dt} \int \int \psi(t, \mathbf{x}, \mathbf{v}) u(\mathbf{x}, \mathbf{v}) d\mathbf{x} d\mathbf{v} \\
&= \frac{d}{dt} u(\mathbf{x}(t), \mathbf{v}(t)) \\
&= \left\langle \frac{d\mathbf{x}(t)}{dt}, \nabla_{\mathbf{x}(t)} u(\mathbf{x}(t), \mathbf{v}(t)) \right\rangle + \left\langle \frac{d\mathbf{v}(t)}{dt}, \nabla_{\mathbf{v}(t)} u(\mathbf{x}(t), \mathbf{v}(t)) \right\rangle \\
&= \langle \mathbf{v}(t), \nabla_{\mathbf{x}(t)} u(\mathbf{x}(t), \mathbf{v}(t)) \rangle + \left\langle \frac{1}{m} \mathbf{F}(t, \mathbf{x}(t), \mathbf{v}(t)), \nabla_{\mathbf{v}(t)} u(\mathbf{x}(t), \mathbf{v}(t)) \right\rangle \quad (1.11) \\
&= \int \int \left(\langle \mathbf{v}, \nabla_{\mathbf{x}} u(\mathbf{x}, \mathbf{v}) \rangle + \left\langle \frac{1}{m} \mathbf{F}(t, \mathbf{x}, \mathbf{v}), \nabla_{\mathbf{v}} u(\mathbf{x}, \mathbf{v}) \right\rangle \right) \\
&\quad \cdot \delta(\mathbf{x} - \mathbf{x}(t)) \delta(\mathbf{v} - \mathbf{v}(t)) d\mathbf{x} d\mathbf{v} \\
&= \int \int \psi(t, \mathbf{x}, \mathbf{v}) \left(\langle \mathbf{v}, \nabla_{\mathbf{x}} u(\mathbf{x}, \mathbf{v}) \rangle + \left\langle \frac{1}{m} \mathbf{F}(t, \mathbf{x}, \mathbf{v}), \nabla_{\mathbf{v}} u(\mathbf{x}, \mathbf{v}) \right\rangle \right) d\mathbf{x} d\mathbf{v}.
\end{aligned}$$

Here, $\langle f, g \rangle$ denotes the scalar product $\int \int f(\mathbf{x}, \mathbf{v}) g(\mathbf{x}, \mathbf{v}) d\mathbf{x} d\mathbf{v}$. Using the first and last term of (1.11), integrating by parts, and making use of the fact that the test function u was chosen arbitrarily, yields Liouville's equation

$$\partial_t \psi(t, \mathbf{x}, \mathbf{v}) + \mathbf{v} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \mathbf{v}) + \frac{1}{m} \mathbf{F}(t, \mathbf{x}, \mathbf{v}) \cdot \nabla_{\mathbf{v}} \psi(t, \mathbf{x}, \mathbf{v}) = 0. \quad (1.12)$$

Liouville's equation describes the evolution of an initial distribution $\psi_0(\mathbf{x}, \mathbf{v}) := \psi(0, \mathbf{x}, \mathbf{v})$ in phase space under the presence of a non-contact force $\mathbf{F}(t, \mathbf{x}, \mathbf{v})$. Under certain assumptions, we can derive conservation of probability along characteristics [10].

1.3. Boltzmann-Grad limit

The idea of the Boltzmann-Grad limit that we have mentioned before was rigorously derived by Grad [43] and formalizes the transition from the microscopic particle billiard to the mesoscopic transport equation.

Clearly the microscopic viewpoint is reversible: At any given time t we can stop the billiard flow, flip the velocity vector, let the process run for another time t , and return to the original arrangement of particles and obstacles. Thus, when the positions of all obstacles $\mathcal{C}_r := \{\mathbf{c}_i = (x_i, y_i)\}_{i=1, \dots, N}$, as well as the initial distribution of particles $\psi_0(\mathbf{x}_0, \mathbf{v}_0)$ are known, the time evolution of particles $\psi(t, \mathbf{x}, \mathbf{v} | \mathcal{C}_r)$ is known, too. The expression $\psi(t, \mathbf{x}, \mathbf{v} | \mathcal{C}_r)$ simply involves tracing back the particles' trajectories through the obstacle field to their origin.

The Boltzmann-Grad limit is then the ensemble average $\langle \psi(t, \mathbf{x}, \mathbf{v} | \mathcal{C}_r) \rangle_{\mathcal{C}_r}$ with $N \rightarrow \infty$ and $r \rightarrow 0$ such that $N \cdot r = \text{constant}$. It involves geometric arguments about the trajectories of particles and statistical arguments about the arrangement of obstacles.

Several assumptions are necessary to arrive at the ensemble-averaged distribution: (i) Obstacles are placed independently. (ii) Situations where particles bounce forth and back between two (or more) obstacles infinitely often are omitted. (iii) Collisions involve exactly two participants, one particle and one obstacle.

Assumption (i) allows for obstacles to overlap. However, the likelihood for this to happen decreases when the obstacles' radii shrink and the contribution of overlapping obstacles will subsequently be dropped in the Boltzmann-Grad limit since it is of higher order (in r).

Interestingly, the resulting expression for the ensemble average is provably non-reversible, causing several controversies in the scientific community [26]. The non-reversibility follows from the non-decrease² in entropy, discussed in Section 1.4.2.

1.4. Linear transport equation

Modeling the evolution of the distribution function for the particle billiard shown in Figure 1.1 can be done—in a first simplified setting—via the linear transport equation. We ignore the presence of an external force \mathbf{F} and instead assume that particles only interact with the background obstacles. These interactions can have two possible outcomes: (i) Particles change their direction due to scattering, and (ii) particles are being absorbed. The probability of these two possible scenarios is determined by the cross sections σ_s and σ_a , respectively. Together with $\sigma_t = \sigma_a + \sigma_s$, the probability to scatter in the case of an event is given by σ_s/σ_t and the probability of being absorbed is given by σ_a/σ_s . In a heterogeneous material, the cross sections will have an additional spatial dependency. Given certain materials, there might also be a dependency on the angle, i.e., the likelihood to undergo an event depends on a particle's direction of travel.

Using all the information above and assuming particles with unit speed and velocity Ω , we obtain the linear transport equation as

$$\begin{aligned} \partial_t \psi(t, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \sigma_a \psi(t, \mathbf{x}, \Omega) \\ = \sigma_s \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') (\psi(t, \mathbf{x}, \Omega') - \psi(t, \mathbf{x}, \Omega)) d\Omega' + q(t, \mathbf{x}, \Omega). \end{aligned} \quad (1.13)$$

Let us dissect the equation above into its main parts. The left-hand side of the equation is an advection-plus-absorption part: Particles move along straight lines while being absorbed at a certain rate. The right-hand side of the equation describes the gain and loss due to in- and out-scattering, as well as the contribution due to a source $q(t, \mathbf{x}, \Omega)$. The scattering kernel $s(\Omega \cdot \Omega')$ describes the probability that a particle with direction Ω changes direction to Ω' (or vice versa) in case of scattering. A common convention is

$$\Omega = \begin{pmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{pmatrix} = \begin{pmatrix} \sqrt{1 - \mu^2} \cos(\phi) \\ \sqrt{1 - \mu^2} \sin(\phi) \\ \mu \end{pmatrix}, \quad (1.14)$$

with the *azimuthal angle* ϕ , the *polar angle* θ , and $\mu = \cos(\theta)$. With a non-negative scattering kernel that integrates to unity, we can split the scattering operator

$$S(\psi)(t, \mathbf{x}, \Omega) := \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') (\psi(t, \mathbf{x}, \Omega') - \psi(t, \mathbf{x}, \Omega)) d\Omega' \quad (1.15)$$

into in-scattering

$$S^+(\psi)(t, \mathbf{x}, \Omega) := \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(t, \mathbf{x}, \Omega') d\Omega' \quad (1.16)$$

²We will actually see a non-increase in *mathematical* entropy instead.

and out-scattering

$$S^-(\psi)(t, \mathbf{x}, \Omega) := \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') d\Omega' \psi(t, \mathbf{x}, \Omega') = \psi(t, \mathbf{x}, \Omega). \quad (1.17)$$

The simplest choice for the scattering kernel models isotropic scattering via $s(\Omega \cdot \Omega') = 1/4\pi$, i.e., particles scatter into all directions equally likely. Another choice is the Henyey-Greenstein kernel [46], satisfying the relation

$$s_g(\Omega' \cdot \Omega) = \frac{1 - g^2}{4\pi (1 - 2g\Omega' \cdot \Omega + g^2)^{3/2}} = \sum_{n=0}^{\infty} g^n \sum_{m=-n}^n Y_n^m(\Omega) \overline{Y_n^m(\Omega')}, \quad (1.18)$$

where Y_n^m are the spherical harmonic basis functions [74] and the over-bar is used for complex conjugation. The Henyey-Greenstein kernel is not the correct physical model for scattering but is commonly used for two reasons: (i) The parameter $g \in [-1, 1]$ allows to tune the scattering from purely backward ($g = -1$) to isotropic scattering ($g = 0$) and to purely forward ($g = 1$), and (ii) it has the convenient aforementioned expansion in terms of spherical harmonics. The scattering kernel is visualized in Figure 1.5 and the relation to the spherical harmonics is discussed in Section 2.3.

1.4.1. Properties of the collision kernel and operator

Let us discuss the scattering operator from (1.15), given by

$$S(\psi)(t, \mathbf{x}, \Omega) := \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') (\psi(t, \mathbf{x}, \Omega') - \psi(t, \mathbf{x}, \Omega)) d\Omega'$$

and its scattering kernel $s(\Omega \cdot \Omega')$ more thoroughly. For every $(t, \mathbf{x}) \in \mathbb{R}^{\geq 0} \times \mathbb{R}^3$, the operator S defined by (1.15) is self-adjoint from $L^2(\mathbb{S}^2)$ to $L^2(\mathbb{S}^2)$. Moreover, we assume the scattering kernel to be bounded in the sense (cf. [30])

$$\sigma_s \int_{\mathbb{S}^2} \int_{\mathbb{S}^2} s(\Omega \cdot \Omega')^2 d\Omega d\Omega' \leq K < \infty. \quad (1.19)$$

An additional assumption is $s(\Omega \cdot \Omega') \geq 0$. In general, the scattering kernel can depend on Ω and Ω' individually—as opposed to $\Omega \cdot \Omega'$. By writing $s(\Omega \cdot \Omega')$, scattering from Ω into Ω' is as likely as scattering from Ω' into Ω . Replacing the dependency on Ω and Ω' by the dependency on $\Omega \cdot \Omega'$ —the cosine of the *scattering angle*—is a significant mathematical simplification. Lastly, since particles have to scatter into *some* direction,

$$\int_{\mathbb{S}^2} s(\Omega \cdot \Omega') d\Omega' = 1 \quad (1.20)$$

is required for any $\Omega \in \mathbb{S}^2$.

Lemma 1.4.1 (Range and kernel of S). *Under the above assumptions, the range and kernel of the scattering operator S are given by*

- $\mathcal{R}(S) = \{\psi(t, \mathbf{x}, \Omega), \text{ such that } \int_{\mathbb{S}^2} \psi(t, \mathbf{x}, \Omega) d\Omega = 0\}$, and
- $\mathcal{K}(S) = \{\psi(t, \mathbf{x}, \Omega), \text{ where } \psi(t, \mathbf{x}, \Omega) \text{ is constant in } \Omega\}$,

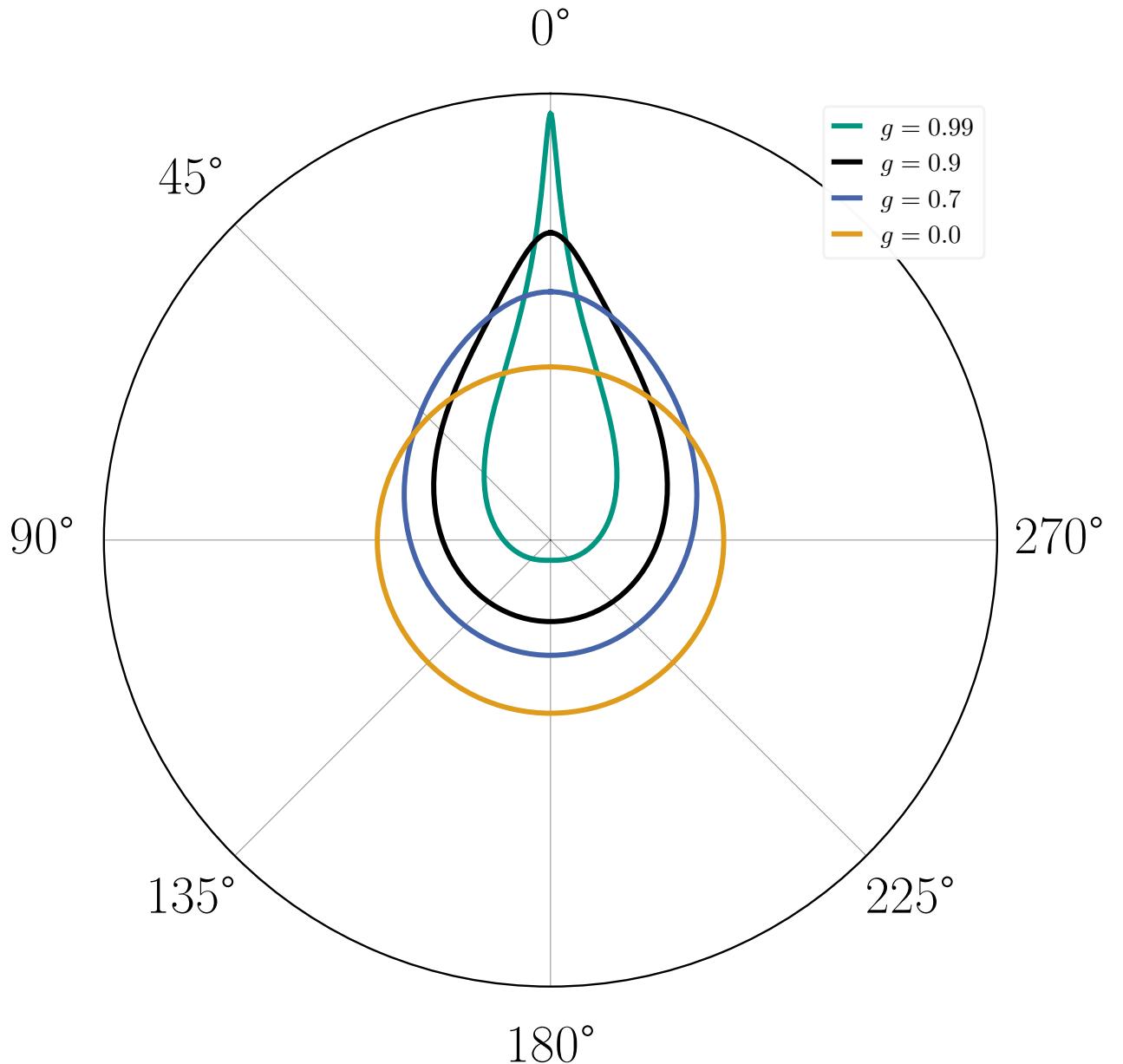


Figure 1.5.: The Henyey-Greenstein scattering kernel $\log_{10} s_g(\cos(\theta))$ for different values of g with $\cos \theta = \Omega \cdot \Omega'$. The radial component is the \log_{10} -probability of scattering into the corresponding angle (relative to the pre-scattering direction). For $g = 0$ (orange), particles scatter isotropically. For $g = 1 - \varepsilon$ with $0 < \varepsilon \ll 1$, particles mostly scatter into the pre-scattering direction (green).

respectively. See, e.g., the lecture notes by Frank [36] for a proof of both properties.

Intuitively, this implies that:

- The scattering operator is mass-preserving. Locally (at t and \mathbf{x}), particles are only redistributed (in Ω). Overall, no particles are gained or lost.
- Unless ψ is isotropic (independent of Ω), scattering will cause the pre-collision state to be different from the post-collision state and, regardless of the scattering kernel s , the equilibrium state of the scattering operator S is isotropic. This does not imply the absence of scattering events; it merely implies a balance of in- and out-scattering.

1.4.2. Entropy for the linear transport equation

Let $\eta : \mathbb{R} \rightarrow \mathbb{R}$ be a differentiable, convex function, i.e., $\forall x_1, x_2 \in \mathbb{R}$ and $\forall t \in [0, 1]$, we have $\eta(t x_1 + (1 - t) x_2) \leq t \eta(x_1) + (1 - t) \eta(x_2)$. Since η is convex, its derivative is monotonically non-decreasing. Furthermore, recall that the scattering component of the transport equation (1.13) was given by

$$S(\psi)(t, \mathbf{x}, \Omega) := \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') (\psi(t, \mathbf{x}, \Omega') - \psi(t, \mathbf{x}, \Omega)) d\Omega'.$$

It follows that

$$\int_{\mathbb{S}^2} \eta'(\psi(t, \mathbf{x}, \Omega)) S(\psi)(t, \mathbf{x}, \Omega) d\Omega \leq 0, \quad (1.21)$$

because

$$\int_{\mathbb{S}^2} \eta'(\psi(t, \mathbf{x}, \Omega)) S(\psi)(t, \mathbf{x}, \Omega) d\Omega \quad (1.22a)$$

$$= \int_{\mathbb{S}^2} \int_{\mathbb{S}^2} \eta'(\psi(t, \mathbf{x}, \Omega)) s(\Omega \cdot \Omega') (\psi(t, \mathbf{x}, \Omega') - \psi(t, \mathbf{x}, \Omega)) d\Omega' d\Omega \quad (1.22b)$$

$$= \int_{\mathbb{S}^2} \int_{\mathbb{S}^2} \eta'(\psi(t, \mathbf{x}, \Omega')) s(\Omega' \cdot \Omega) (\psi(t, \mathbf{x}, \Omega) - \psi(t, \mathbf{x}, \Omega')) d\Omega d\Omega' \quad (1.22c)$$

$$= \frac{1}{2} \int_{\mathbb{S}^2} \int_{\mathbb{S}^2} \underbrace{(\eta'(\psi(t, \mathbf{x}, \Omega)) - \eta'(\psi(t, \mathbf{x}, \Omega')))}_{=:c_1} \underbrace{s(\Omega \cdot \Omega')}_{\geq 0} \underbrace{(\psi(t, \mathbf{x}, \Omega') - \psi(t, \mathbf{x}, \Omega))}_{=:c_2} d\Omega d\Omega' \quad (1.22d)$$

$$\leq 0, \quad (1.22e)$$

since the signs of c_1 and c_2 are opposite due to the convexity of η . Additionally, if $\psi(t, \mathbf{x}, \Omega) \equiv 0$ for $\|\mathbf{x}\|_2 \rightarrow \infty$, the mathematical entropy

$$H(\psi)(t) := \int_{\mathbb{R}^3} \int_{\mathbb{S}^2} \eta(\psi(t, \mathbf{x}, \Omega)) d\Omega d\mathbf{x} \quad (1.23)$$

is non-increasing. The proof follows from the convexity of η and the symmetry of the scattering kernel.

With a non-increasing entropy the process cannot be reversible. Although, on the microscopic level, particles return to their initial position when we continue the billiard game for an additional time t with reversed velocities, this cannot be the case here. Flipping the velocities

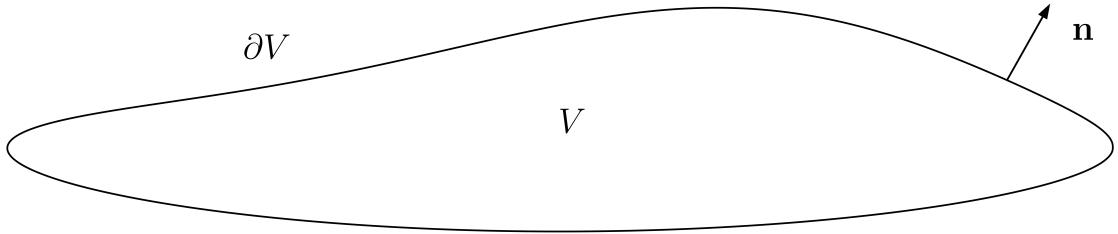


Figure 1.6.: Spatial domain $V \subsetneq \mathbb{R}^3$ with boundary ∂V and outward facing normal \mathbf{n} .

at time t does not change the fact that the entropy will continue to decrease. Therefore, the original state of the system can never be recovered.

An explanation for this discrepancy is provided by Spohn [91]. Due to the assumptions that we make for the Boltzmann-Grad limit, certain states (situations where obstacles overlap or particles bounce forth and back indefinitely) are omitted from the description with the linear transport equation. As a result, transport described by the linear transport equation is not exactly equivalent to that described via the microscopic perspective.

1.4.3. Initial and boundary conditions

In general, (partial) differential equations describe *how* a system evolves due to physical laws. Boundary and initial conditions describe *which* system we are considering exactly. Thus, the linear transport equation (1.13) needs to be equipped with suitable boundary and initial conditions.

For a bounded domain $V \subsetneq \mathbb{R}^3$ with boundary ∂V and outward facing normal \mathbf{n} as depicted in Figure 1.6, we set the incoming flux

$$\psi(t, \mathbf{x}, \boldsymbol{\Omega}) = \psi^{\text{bc}}(t, \mathbf{x}, \boldsymbol{\Omega}) \quad \text{for all } t \in \mathbb{R}^{\geq 0}, \mathbf{x} \in \partial V, \boldsymbol{\Omega} \in \mathbb{S}^2 \text{ such that } \boldsymbol{\Omega} \cdot \mathbf{n} < 0. \quad (1.24)$$

However, different ways to treat the boundary exist. We can as well have periodic boundaries or (specular) reflective boundaries, i.e., particles that hit the boundary are being (isotropically) re-emitted into the domain.

For the initial condition, we have to prescribe the angular flux at $t = 0$, i.e.,

$$\psi(0, \mathbf{x}, \boldsymbol{\Omega}) = \psi_0(\mathbf{x}, \boldsymbol{\Omega}) \quad \text{for } \mathbf{x} \in V, \boldsymbol{\Omega} \in \mathbb{S}^2. \quad (1.25)$$

Furthermore, we require the source $q(t, \mathbf{x}, \boldsymbol{\Omega})$ to be non-negative for all times $t \in \mathbb{R}^{\geq 0}$. Given these conditions, the uniqueness of the angular flux follows.

1.4.4. Stationary transport

Frequently, the stationary state is of more interest than the actual evolution in time towards that state. In the absence of time-dependent internal sources or boundary conditions, we can

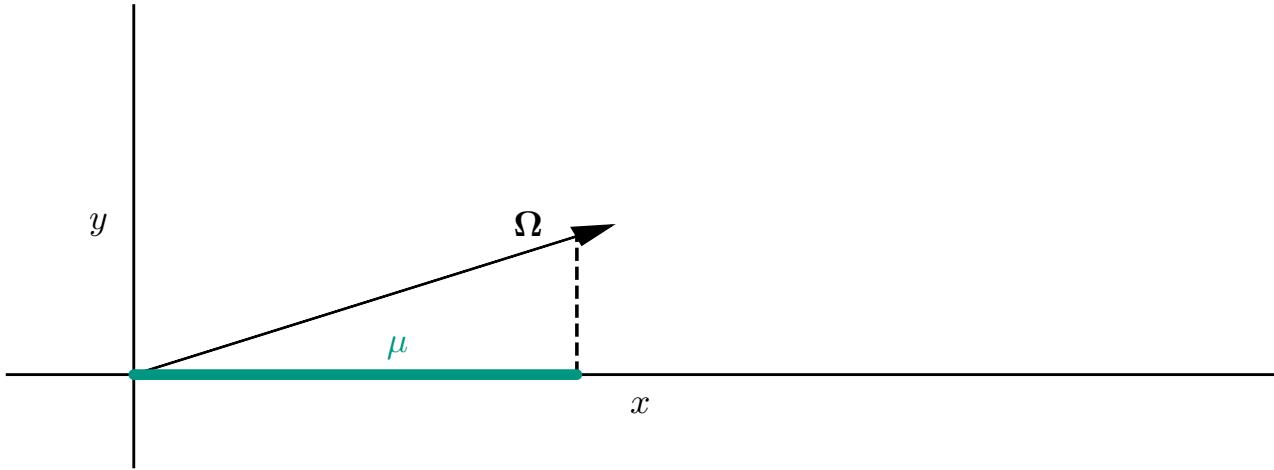


Figure 1.7.: Slab geometry for two-dimensional transport. Transport is projected onto the x -axis.

reduce (1.13) to its time-independent counterpart

$$\begin{aligned} & \Omega \cdot \nabla_x \psi(\mathbf{x}, \Omega) + \sigma_a \psi(\mathbf{x}, \Omega) \\ &= \sigma_s \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') (\psi(\mathbf{x}, \Omega') - \psi(\mathbf{x}, \Omega)) d\Omega' + q(\mathbf{x}, \Omega). \end{aligned} \quad (1.26)$$

Especially when applied to nuclear engineering related problems, the resolution of the temporal scale is often not necessary. Equipped with a positive source $q(\mathbf{x}, \Omega)$ and positive boundary conditions $\psi^{bc}(\mathbf{x}, \Omega)$, there exists a non-negative solution to (1.26). In Section 1.5 we will show that this steady state solution exists only for critical systems (and not for super- or subcritical ones) when introducing more physics into the equations.

1.4.5. Slab geometry

A common simplification of the spatially two- or three-dimensional transport equation is slab geometry. Consider Figure 1.7, depicting the situation for the two-dimensional case. When exclusively interested in the transport behavior along a single axis (the x -axis in this case), we can project onto that axis and reduce the dimension to one. Consequently, the solution is the same for every horizontal cut through the slab. The transport equation for slab geometry is given by

$$\begin{aligned} & \partial_t \psi(t, x, \mu) + \mu \partial_x \psi(t, x, \mu) + \sigma_a \psi(t, x, \mu) \\ &= \sigma_s \int_{-1}^1 s(\mu, \mu') (\psi(t, x, \mu') - \psi(t, x, \mu)) d\mu' + q(t, x, \mu), \end{aligned} \quad (1.27)$$

with $\mu = \Omega \cdot (1, 0, 0)^T$.

1.5. Neutron transport

After the Boltzmann equation had established its track record as a useful tool in the study of the theory of gases, neutron transport became a highly researched topic with the development of nuclear reactors and the Manhattan project in the 1940s. Different from the scenarios

considered so far, the phase space is augmented by an additional energy dependency. Consequently, particles—or the respective angular flux—are no longer solely defined by their position in space \mathbf{x} and their direction of flight $\boldsymbol{\Omega}$ at a given time t , but also by their energy E , defined by $E = \frac{1}{2}mv^2$. Here, m is the particle's mass and v its speed. Additionally, scattering and absorption are not the only possibilities when colliding: A fission event might be initiated by the interaction of the nuclei.

The (isotropic) energy-dependent cross sections then satisfy

$$\sigma_a(E) + \sigma_s(E) + \sigma_f(E) = \sigma_t(E), \quad (1.28)$$

where $\sigma_f(E)$ is the fission cross section. The scattering kernel is interpreted as

$$s(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}', E \rightarrow E') d\boldsymbol{\Omega}' dE' := \text{“the probability that a neutron which scatters and has pre-collision direction } \boldsymbol{\Omega} \text{ and energy } E, \text{ has a post-collision direction in } d\boldsymbol{\Omega}' \text{ about } \boldsymbol{\Omega}' \text{ and energy in } dE' \text{ about } E'.” \quad (1.29)$$

In the case of a fission event, the nucleus splits into two nuclei. On average, this produces $\nu(E)$ new neutrons that can be divided into *prompt neutrons* and *delayed neutrons*, occurring with probability $1 - \beta(E)$ and $\beta(E)$, respectively [82]. Here, $\beta(E)$ denotes the *delayed neutron fraction* and is usually small (≈ 0.001). When we assume (i) fission neutrons to be born with uniform direction and (ii) energy distributed according to their prompt fission spectrum $\chi_p(E)$, we can write the respective gain term (gain of $\psi(t, \mathbf{x}, \boldsymbol{\Omega}, E)$) as

$$\chi_p(E) \int_0^\infty \int_{\mathbb{S}^2} (1 - \beta(E')) \sigma_f(E') \nu(E') \psi(t, \mathbf{x}, \boldsymbol{\Omega}', E') d\boldsymbol{\Omega}' dE'. \quad (1.30)$$

In the absence of delayed neutrons (β being small), the energy-dependent transport equation that includes fission reads

$$\begin{aligned} & \frac{1}{v} \partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}, E) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}, E) + \sigma_a(E) \psi(t, \mathbf{x}, \boldsymbol{\Omega}, E) \\ &= \int_{\mathbb{S}^2} \int_0^\infty \sigma_s(E') s(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}', E' \rightarrow E) (\psi(t, \mathbf{x}, \boldsymbol{\Omega}', E') - \psi(t, \mathbf{x}, \boldsymbol{\Omega}, E)) dE' d\boldsymbol{\Omega}' \\ &+ \chi_p(E) \int_{\mathbb{S}^2} \int_0^\infty (1 - \beta(E')) \sigma_f(E') \nu(E') \psi(t, \mathbf{x}, \boldsymbol{\Omega}', E') dE' d\boldsymbol{\Omega}' \\ &+ q(t, \mathbf{x}, \boldsymbol{\Omega}, E), \end{aligned} \quad (1.31)$$

with the neutron speed $v = \sqrt{2E/m}$. Dropping the time derivative in (1.31) results in the time-independent, energy-dependent neutron transport equation

$$\begin{aligned} & \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(\mathbf{x}, \boldsymbol{\Omega}, E) + \sigma_a(E) \psi(\mathbf{x}, \boldsymbol{\Omega}, E) \\ &= \int_{\mathbb{S}^2} \int_0^\infty \sigma_s(E') s(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}', E' \rightarrow E) (\psi(\mathbf{x}, \boldsymbol{\Omega}', E') - \psi(t, \mathbf{x}, \boldsymbol{\Omega}, E)) dE' d\boldsymbol{\Omega}' \\ &+ \chi_p(E) \int_{\mathbb{S}^2} \int_0^\infty (1 - \beta(E')) \sigma_f(E') \nu(E') \psi(\mathbf{x}, \boldsymbol{\Omega}', E') dE' d\boldsymbol{\Omega}' \\ &+ q(\mathbf{x}, \boldsymbol{\Omega}, E). \end{aligned} \quad (1.32)$$

1.5.1. Criticality calculations

In the absence of sources and with homogeneous boundary conditions the fundamental behavior of the time-independent neutron transport equation is determined by its *criticality*. Here, criticality refers to the largest eigenvalue k of

$$\begin{aligned} & \Omega \cdot \nabla_{\mathbf{x}} \psi(\mathbf{x}, \Omega, E) + \sigma_a(E) \psi(\mathbf{x}, \Omega, E) \\ &= \int_{\mathbb{S}^2} \int_0^\infty \sigma_s(E') s(\Omega \cdot \Omega', E' \rightarrow E) (\psi(\mathbf{x}, \Omega', E') - \psi(\mathbf{x}, \Omega, E)) dE' d\Omega' \\ &+ \frac{1}{k} \chi_p(E) \int_{\mathbb{S}^2} \int_0^\infty \sigma_f(E') \nu(E') \psi(\mathbf{x}, \Omega', E') dE' d\Omega', \end{aligned} \quad (1.33)$$

with vacuum boundary conditions $\psi(\mathbf{x}, \Omega, E) = 0$ for $\mathbf{x} \in \partial V$, $\Omega \cdot \mathbf{n}(\mathbf{x}) < 0$, and all energies E . We assume that it is possible to control the expected number of fission neutrons that are generated and hence substituted $\nu(E')$ by $\nu(E')/k$ in the time-independent energy-dependent neutron transport equation. Clearly, $\psi \equiv 0$ is a trivial solution to (1.33). However, for non-trivial solutions the system can be in one of the following three states, depending on the largest eigenvalue k :

1. The system is subcritical for $k < 1$. Fission does not generate enough particles and absorption is dominant.
2. The system is critical for $k = 1$. Fission generates particles at the exact same rate that particles are being absorbed or are leaking out of the system. A non-trivial solution to (1.33) exists.
3. The system is supercritical for $k > 1$. Neutrons are being generated at a higher rate than being lost.

This so-called k eigenvalue problem is of significant importance in the field of steady-state reactor physics. There, the goal is to operate a reactor at a critical state, controlling the rate of generated fission neutrons such that the reactor neither becomes supercritical (too many neutrons being generated), nor subcritical (too few neutrons being generated).

1.6. Asymptotic limits

Let us, once again, consider the velocity-dependent, linear transport equation, expressed in terms of the total cross section σ_t and the probability that a colliding particle will scatter c , given by

$$\begin{aligned} & \frac{1}{v} \partial_t \psi(t, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \sigma_t \psi(t, \mathbf{x}, \Omega) \\ &= c \sigma_t \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(t, \mathbf{x}, \Omega') d\Omega' + q(t, \mathbf{x}, \Omega). \end{aligned}$$

To obtain a dimensionless equation, we set

$$t = \bar{t} T, \quad (1.34a)$$

$$\mathbf{x} = \bar{\mathbf{x}} L, \quad (1.34b)$$

$$v = \bar{v} V, \quad (1.34c)$$

$$\sigma_t = \bar{\sigma}_t \Sigma_t, \quad (1.34d)$$

where variables with a superscript bar denote dimensionless variables and capitalized variables denote the reference time, length, velocity, and cross section, respectively. We further assume the scattering kernel to already be dimensionless. Substituting (1.34) in the linear transport equation and omitting the source yields

$$\begin{aligned} \frac{1}{\bar{v}V} \partial_{\bar{t}} T \psi(\bar{t}T, \bar{\mathbf{x}}L, \Omega) + \Omega \cdot \nabla_{\bar{\mathbf{x}}} L \psi(\bar{t}T, \bar{\mathbf{x}}L, \Omega) + \bar{\sigma}_t \Sigma_t \psi(\bar{t}T, \bar{\mathbf{x}}L, \Omega) \\ = c \bar{\sigma}_t \Sigma_t \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(\bar{t}T, \bar{\mathbf{x}}L, \Omega) d\Omega. \end{aligned} \quad (1.35)$$

We define³ $\psi(\bar{t}T, \bar{\mathbf{x}}L, \Omega) = \bar{\psi}(\bar{t}, \bar{\mathbf{x}}, \Omega) \Psi$ and directly omit the superscript bar everywhere to get

$$\begin{aligned} \frac{\Psi}{TV} \frac{1}{\bar{v}} \partial_t \psi(t, \mathbf{x}, \Omega) + \frac{\Psi}{L} \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \Psi \Sigma_t \sigma_t \psi(t, \mathbf{x}, \Omega) \\ = c \Psi \Sigma_t \sigma_t \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(t, \mathbf{x}, \Omega') d\Omega', \end{aligned} \quad (1.36)$$

or equivalently

$$\begin{aligned} \frac{L}{VT} \frac{1}{v} \partial_t \psi(t, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \Sigma_t L \sigma_t \psi(t, \mathbf{x}, \Omega) \\ = c \Sigma_t L \sigma_t \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(t, \mathbf{x}, \Omega') d\Omega'. \end{aligned} \quad (1.37)$$

Equation (1.37) can be rewritten in terms of the Strouhal number ($St := L/(VT)$) and the Knudsen number ($Kn := 1/(\Sigma_t L)$). We additionally set $v = 1$ since the velocity can be controlled by V , and v does not occur in the equation without being multiplied by V . The transport equation in dimensionless form then reads

$$\begin{aligned} St \partial_t \psi(t, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \frac{1}{Kn} \sigma_t \psi(t, \mathbf{x}, \Omega) \\ = \frac{c}{Kn} \sigma_t \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(t, \mathbf{x}, \Omega') d\Omega'. \end{aligned} \quad (1.38)$$

The inverse Knudsen number $\Sigma_t L$ is a measure for thickness of the system in units of mean free paths. The Strouhal number is interpreted as the ratio between the time needed to cross the domain by the characteristic velocity and the characteristic time scale [36].

Under different assumptions on the order of the Knudsen and Strouhal number, we are now able to obtain different asymptotic limits to the standard transport equation. Due to its importance in theoretical considerations, as well as its implications for numerical methods, we will discuss the diffusive scaling in more detail. Diffusive scaling uses the following assumptions:

1. The mean free path is much smaller than the reference length ($Kn = \varepsilon$).
2. Interactions happen on a much smaller time-scale than the reference time ($St = \varepsilon$).
3. The likelihood to scatter is much, much larger than the likelihood to be absorbed in case of a collision ($c = 1 - \varepsilon^2$).

³It is the velocity v , not the direction Ω that has to be scaled since Ω is, by definition, a unit-vector.

First we rewrite (1.38) by moving out-scattering back on the right-hand side.

$$\begin{aligned} \text{St} \partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \frac{1-c}{\text{Kn}} \sigma_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) \\ = \frac{c}{\text{Kn}} \sigma_t \int_{\mathbb{S}^2} s(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') (\psi(t, \mathbf{x}, \boldsymbol{\Omega}') - \psi(t, \mathbf{x}, \boldsymbol{\Omega})) d\boldsymbol{\Omega}'. \end{aligned} \quad (1.39)$$

We now substitute $\text{Kn} = \varepsilon$ and $\text{St} = \varepsilon$. Next, $(1-c)\sigma_t$ and $c\sigma_t$ can be expressed in terms of ε and dimensionless quantities as $\varepsilon^2\sigma_a$ and σ_s , respectively. A final multiplication with ε results in

$$\begin{aligned} \varepsilon^2 \partial_t \psi_{(\varepsilon)}(t, \mathbf{x}, \boldsymbol{\Omega}) + \varepsilon \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_{(\varepsilon)}(t, \mathbf{x}, \boldsymbol{\Omega}) + \varepsilon^2 \sigma_a \psi_{(\varepsilon)}(t, \mathbf{x}, \boldsymbol{\Omega}) \\ = \sigma_s \int_{\mathbb{S}^2} s(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') (\psi_{(\varepsilon)}(t, \mathbf{x}, \boldsymbol{\Omega}') - \psi_{(\varepsilon)}(t, \mathbf{x}, \boldsymbol{\Omega})) d\boldsymbol{\Omega}'. \end{aligned} \quad (1.40)$$

The diffusion limit lets $\varepsilon \rightarrow 0$. To compute $\psi_{(\varepsilon)}$ for $\varepsilon \rightarrow 0$, we assume the existence of a Hilbert expansion for $\psi_{(\varepsilon)}$, that is

$$\psi_{(\varepsilon)}(t, \mathbf{x}, \boldsymbol{\Omega}) = \sum_{n=0}^{\infty} \varepsilon^n \psi_i(t, \mathbf{x}, \boldsymbol{\Omega}). \quad (1.41)$$

The ψ_i in the Hilbert expansion are independent of ε . We substitute (1.41) into (1.40) and collect matching orders of ε (up to second order) to get

$$\mathcal{O}(\varepsilon^0) : \quad 0 = \sigma_s S(\psi_0), \quad (1.42a)$$

$$\mathcal{O}(\varepsilon^1) : \quad \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_0 = \sigma_s S(\psi_1), \quad (1.42b)$$

$$\mathcal{O}(\varepsilon^2) : \quad \partial_t \psi_0 + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_1 + \sigma_a \psi_0 = \sigma_s S(\psi_2), \quad (1.42c)$$

where we omit the dependency on t , \mathbf{x} , and $\boldsymbol{\Omega}$ in the ψ_i and the collision operator $S(\psi_i)$. Section 1.4.1 tells us that the kernel of S consists of functions that are independent of $\boldsymbol{\Omega}$; this implies $\psi_0(t, \mathbf{x}, \boldsymbol{\Omega}) = \psi_0(t, \mathbf{x})$.

Since S is linear, we can rewrite (1.42b) as

$$\psi_1(t, \mathbf{x}, \boldsymbol{\Omega}) = -\frac{1}{\sigma_s} \sum_{i=1}^3 e_i(\boldsymbol{\Omega}) \partial_{x_i} \psi_0(t, \mathbf{x}), \quad (1.43)$$

for some $e_i \in \mathcal{R}(S)$ with $S(e_i) = -\boldsymbol{\Omega}_i$ for $i = 1, 2, 3$ [36]. Inserting this into (1.42c), we get

$$\partial_t \psi_0(t, \mathbf{x}) - \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \frac{1}{\sigma_s} \sum_{i=1}^3 e_i(\boldsymbol{\Omega}) \partial_{x_i} \psi_0(t, \mathbf{x}) + \sigma_a \psi_0(t, \mathbf{x}) = \sigma_s S(\psi_2). \quad (1.44)$$

For the left-hand side to be in the range of S , ψ_0 has to be chosen such that

$$\int_{\mathbb{S}^2} 1 \cdot \left(\partial_t \psi_0(t, \mathbf{x}) - \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \frac{1}{\sigma_s} \sum_{i=1}^3 e_i(\boldsymbol{\Omega}) \partial_{x_i} \psi_0(t, \mathbf{x}) + \sigma_a \psi_0(t, \mathbf{x}) \right) d\boldsymbol{\Omega} = 0. \quad (1.45)$$

Changing the order of differentiation and integration in the second term, we obtain

$$\partial_t \psi_0(t, \mathbf{x}) - \nabla_{\mathbf{x}} \frac{1}{\sigma_s} A \nabla_{\mathbf{x}} \psi_0(t, \mathbf{x}) + \sigma_a \psi_0(t, \mathbf{x}) = 0, \quad (1.46)$$

with a positive definite matrix $A \in \mathbb{R}^{3 \times 3}$ that has entries $A_{i,j} = \int_{\mathbb{S}^2} \boldsymbol{\Omega}_i e_j(\boldsymbol{\Omega}) d\boldsymbol{\Omega}$.

We assumed constant cross sections independent of \mathbf{x} , but (1.46) can analogously be written for spatially varying cross sections.

1.7. Hyperbolic conservation laws

Conservation laws are building blocks for the mathematical description of physical reality. They describe the evolution of one or several conserved quantities through space-time; common conserved quantities are mass, momentum, or energy. We adapt the notation of the classical book by LeVeque [62]. A scalar hyperbolic conservation law has the form

$$\partial_t u(t, x) + \partial_x f(u(t, x)) = 0. \quad (1.47)$$

The conserved quantity is $u : \mathbb{R}^{\geq 0} \times V \rightarrow \mathbb{R}$. Time is denoted by $t \in \mathbb{R}^{\geq 0}$ and space by $x \in V \subseteq \mathbb{R}$. For a spatially infinite domain, conservation means that $\int_{-\infty}^{\infty} u(t, x) dx$ is constant in time. The flux function of the system is given by $f : \mathbb{R} \rightarrow \mathbb{R}$ and can be interpreted as the flux of a conserved quantity across a surface. For n conserved quantities $\mathbf{u}(t, x) = (u_1(t, x), \dots, u_n(t, x))^T$ in one space dimension with fluxes $\mathbf{f}(\mathbf{u}(t, x)) = (f_1(\mathbf{u}(t, x)), \dots, f_n(\mathbf{u}(t, x)))^T$,

$$\partial_t \mathbf{u}(t, x) + \partial_x \mathbf{f}(\mathbf{u}(t, x)) = 0 \quad (1.48)$$

is the corresponding system of n conservation laws. The system is said to be hyperbolic if

$$A(\mathbf{u}(t, x)) := \mathbf{f}'(\mathbf{u}(t, x)) \quad (1.49)$$

is diagonalizable with n real eigenvalues for all $\mathbf{u}(t, x)$. If the n real eigenvalues are also distinct, the system is *strictly* hyperbolic. The quasilinear form of (1.48) is

$$\partial_t \mathbf{u}(t, x) + A(\mathbf{u}(t, x)) \partial_x \mathbf{u}(t, x) = 0. \quad (1.50)$$

For problems with two spatial dimension and $\mathbf{x} = (x, y) \in \mathbb{R}^2$, we write

$$\partial_t \mathbf{u}(t, \mathbf{x}) + \partial_x \mathbf{f}(\mathbf{u}(t, \mathbf{x})) + \partial_y \mathbf{g}(\mathbf{u}(t, \mathbf{x})) = 0, \quad (1.51)$$

with $\mathbf{f}(\mathbf{u}(t, \mathbf{x})) = (f_1(\mathbf{u}(t, \mathbf{x})), \dots, f_n(\mathbf{u}(t, \mathbf{x})))^T$ and $\mathbf{g}(\mathbf{u}(t, \mathbf{x}))$ defined analogously.

Example 1.7.1 (Linear advection equation). The simplest hyperbolic conservation law is the one-dimensional linear advection equation with constant speed a ,

$$\partial_t \rho(t, x) + a \partial_x \rho(t, x) = 0, \quad (1.52)$$

equipped with initial condition $\rho(0, x) = \rho_0(x)$ for $t \in \mathbb{R}^{\geq 0}$ and $x \in \mathbb{R}$. The solution of (1.52) is given by $\rho(t, x) = \rho_0(x - at)$ (that is, the initial mass is just shifted under consideration of a), shown in Figure 1.8.

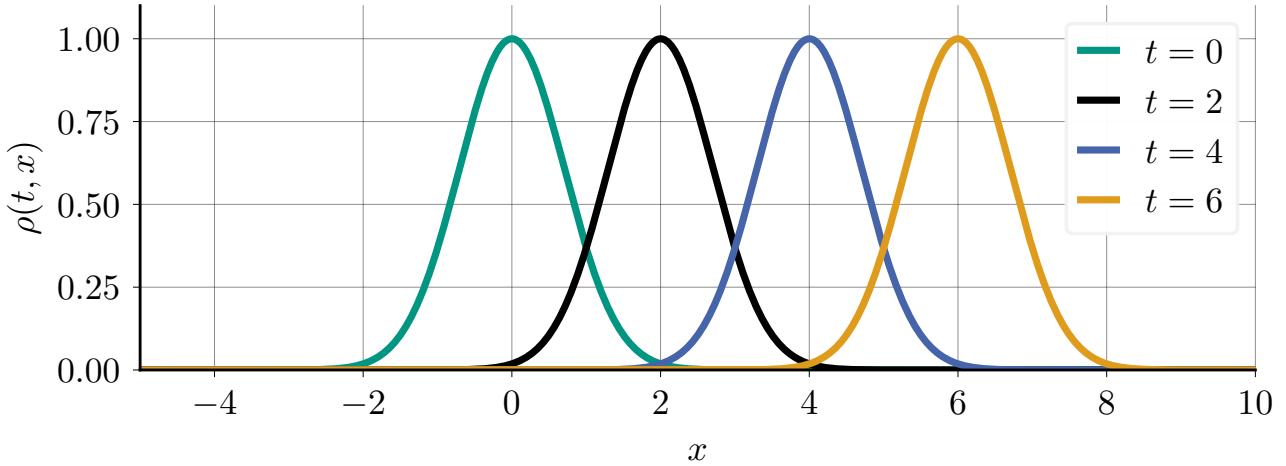


Figure 1.8.: Linear advection equation with $\rho_0(x) = \exp(-x^2)$.

Example 1.7.2 (Inviscid Burgers's equation). Let us now consider the inviscid Burgers's equation as a toy model for non-linear hyperbolic conservation laws. The equation is given by

$$\partial_t u(t, x) + u \partial_x u(t, x) = 0, \quad (1.53)$$

subject to initial data $u_0(x) = u(0, x)$. Figure 1.9 illustrates the evolution of the initial condition

$$u_0(x) = \begin{cases} 2, & x < 0, \\ 2 - x/2, & 0 \leq x < 2, \\ 1, & x \geq 2, \end{cases} \quad (1.54)$$

drawn in green. Since the propagation speeds of the solution differ, a jump in the solution—called a shock—forms in finite time. Shocks are a ubiquitous phenomena in physical applications; exemplified by the formation of traffic jams in traffic flow simulations or pressure jumps in the flow field around airfoils.

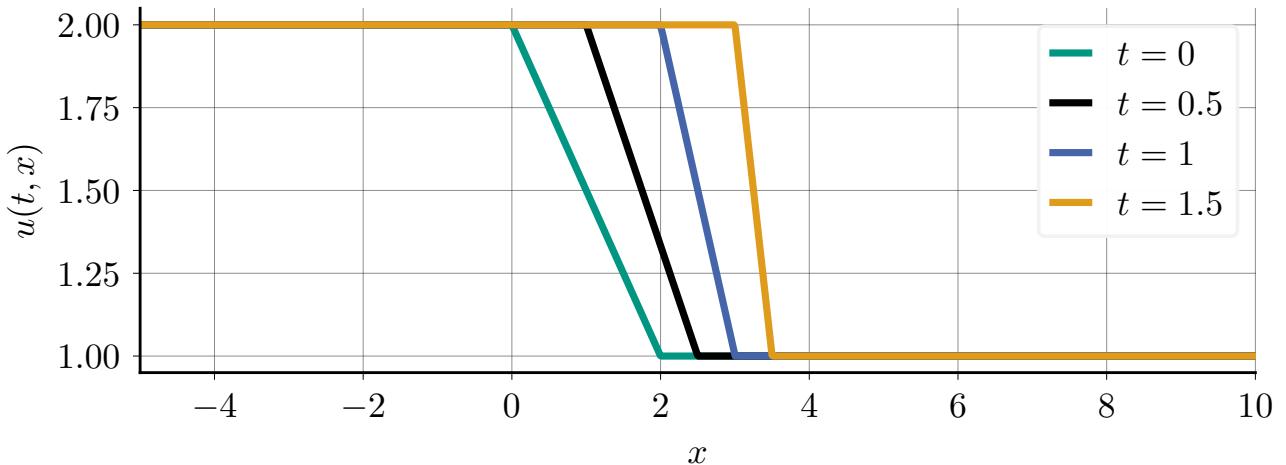


Figure 1.9.: Burgers's equation with initial data $u_0(x)$ shown in green.

Since the analytical solution to non-linear hyperbolic conservation laws is known for exceptionally few—and usually exceptionally simple—test cases, numerical methods need to be considered. Significant contributions in the field of numerical methods for these types of equations

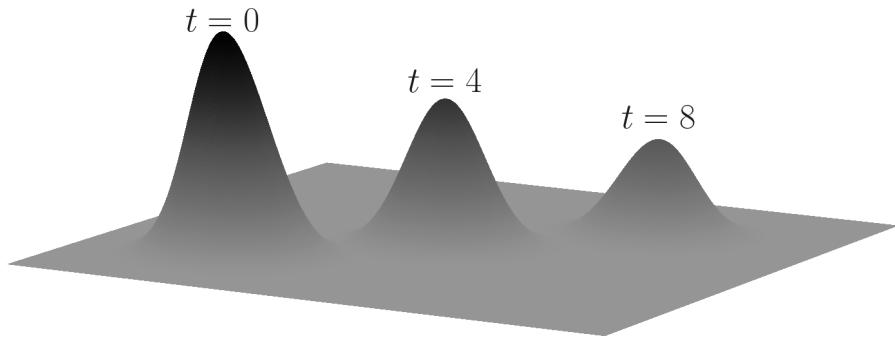


Figure 1.10.: Evolution of an initial distribution $\psi_0(\mathbf{x}, \Omega)$ through space along a fixed direction Ω with constant σ_a . Each *bump* represents the solution at a different time step.

date back to the 1950s and 1960s with work by Sergei Konstantinovich Godunov, Peter Lax, Kurt Otto Friedrichs, and Burton Wendroff.⁴

Example 1.7.3 (Radiative transfer with absent scattering in two space dimensions). Recall that the linear transport equations without scattering or sources can be written as

$$\partial_t \psi(t, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \sigma_a \psi(t, \mathbf{x}, \Omega) = 0.$$

With scattering being absent, the equation can easily be solved for any fixed direction Ω . Choosing $\psi(0, \mathbf{x}, \Omega) = \psi_0(\mathbf{x}, \Omega)$, the solution becomes

$$\psi(t, \mathbf{x}, \Omega) = \psi_0(\mathbf{x} - t\Omega, \Omega) e^{-\sigma_a t}.$$

Thus, the initial condition is propagated along direction Ω with absorption taking place at the same time. When considering scattering, the solution cannot be derived analytically, instead numerical algorithms need to be used. Note that without scattering, the linear transport equation resembles the structure of the simple linear advection equation. This is illustrated in Figure 1.10, where we can see the solution $\psi(t, \mathbf{x}, \Omega)$ to the linear transport equation for different times t with a fixed direction Ω . As initial condition, we chose $\psi_0(\mathbf{x}, \Omega) = \exp(-\|\mathbf{x}\|_2^2)$, the fixed direction is $\Omega = (1/\sqrt{2}, 1/\sqrt{2})^T$, and $\sigma_a = 0.1$.

⁴This is by no means a complete list of all contributions. The names were selected since there exist famous numerical schemes named after all of these researchers.

2. Numerical solution methods

“There ain’t no such thing as a free lunch.” Popularized by Robert Heinlein’s 1966 novel *The Moon is a Harsh Mistress*, this statement tends to be ubiquitous in mathematics and the sciences as well. We have seen in the previous section that the transport equation is a powerful tool, allowing us to describe a variety of systems in a detailed manner. Thus it is unsurprising that solving the transport equation is costly at best and infeasible at worst. Solution methods—analytical and numerical in nature—have been research topics for decades. Because analytical solutions can only be derived in oversimplified scenarios, we will exclusively investigate numerical solution methods.

Broadly speaking, numerical solution methods fall into one of the following two categories: They either involve some degree of (pseudo-)randomness or they are purely deterministic. Both categories have their merits and deficiencies; are both rich in mathematical theory and relevant for applications; and tend to cause partisan debates about which of them is better.

Though this section focuses more strongly on deterministic solution methods, we will also sketch the fundamental idea of the Monte Carlo method as a representative of non-deterministic methods. On the deterministic side, the main focus will be on the discrete ordinates method—also known as the S_N method—but moment methods will be discussed as well. We will start with the Monte Carlo method, because it nicely transforms the underlying physics of the transport equation into a numerical method, allowing us to recall some of the principles and ideas that were discussed so far. A short introduction with relevant historical context is prepended to the mathematical description of the respective methods.

2.1. Monte Carlo method

Origins of the Monte Carlo method date back as early as 1946. Back then, Stanisław Marcin Ulam suggested the method to John von Neumann as a way to compute the success rate for the card game Canfield solitaire [35]. Canfield solitaire is luck based since not all configurations allow to successfully finish the game. The fundamental idea that Ulam and von Neumann pioneered—and that has arguably been unchanged until today—was the following: Instead of coming up with an analytical way to predict the success rate, one could repeatedly play newly shuffled instances of the game and keep track of the number of successful finishes; ultimately gauging the success rate as the ratio of that number by the total number of games played. One year later, in March 1947, Ulam, von Neumann, and Robert Davis Richtmeyer wrote a report that outlined the applicability of the Monte Carlo method for neutron diffusion and multiplication problems, together with tentative computing sheets [96]. In that report, von Neumann also speculates on the computational costs:

I cannot assert this with certainty yet, but it seems to me very likely that the instructions given on this 'computing sheet' do not exceed the 'logical' capacity of

the ENIAC¹. I doubt that the processing of 100 'neutrons' will take much longer than the reading, punching, and (once) sorting time of 100 cards; i.e., about 3 minutes. Hence, taking 100 'neutrons' through 100 of these stages should take about 300 minutes; i.e., 5 hours.

The Monte Carlo method, ENIAC, and its 1951 predecessor MANIAC—Mathematical and Numerical Integrator and Calculator—played essential roles in the Manhattan project and the development of the first hydrogen bomb [70, 13].

Nowadays, the Monte Carlo method is used in civil nuclear applications as well. Both radiation therapy [99, 77, 87] and nuclear reactor design [88, 102] inherently rely on the Monte Carlo method. But also nearly every other discipline in the sciences uses the Monte Carlo method to compute quantities that are as diverse as the fields themselves. For example, biology [66, 72], chemistry [44, 76], social sciences [9, 25, 45], or finance [47, 50] make use of Monte Carlo methods. This work will focus on the Monte Carlo method as a tool to solve the transport equation approximately.

2.1.1. Random numbers

Without random numbers, there would be no Monte Carlo method. Not only is it important to sample random numbers *fast enough*, they also need to be *good enough*. True random number generators are infeasible when random numbers need to be generated quickly, limiting most applications to pseudo-random numbers instead.

A common pseudo-random number generator is the linear congruential generator, which generates a sequence of numbers

$$X_{i+1} = aX_i + b \bmod c, \quad (2.1)$$

with a , b , and c being large, carefully chosen integers. A pseudo-random number generator does therefore not generate random numbers in the sense that they are irreproducible. Any sequence can be reproduced when a , b , and c are known. However, being presented with only a set of pseudo-random numbers $\{X_0, X_1, \dots, X_i\}$, it is hard to predict X_{i+1} . The linear congruential generator is one of the simpler generators and therefore unsuited for cryptographic applications. More sophisticated generators are the Mersenne Twister [68], the WELL (well equidistributed long-period linear) generator [78], or xorshift random number generators [67] to only name a few prominent representatives. Hereinafter, we assume that we are equipped with an algorithm that generates uniform random numbers in the interval $[0, 1]$. The distinction between pseudo-random and random is omitted; pseudo-randomness is assumed implicitly from now on.

Since, for example, the distance that a particle travels between consecutive collisions is not uniformly distributed, samples from non-uniform distributions with arbitrary probability density f_X need to be generated as well. Two common methods for generating these samples are inverse transform sampling and rejection sampling.

¹The Electronic Numerical Integrator and Computer (ENIAC) was the first general-purpose Turing-complete computer. It was completed and put to work in 1945.

2.1.1.1. Inverse transform sampling

Let us assume that we want to generate samples from a distribution with a Lebesgue-integrable probability density function $f_X : [a, b] \rightarrow \mathbb{R}^{\geq 0}$, satisfying $\int_a^b f_X(x') dx' = 1$. The cumulative distribution function is defined as $F_X(x) := \int_a^x f_X(x') dx'$. The random variable X has probability density function f_X if

$$\Pr[x_0 \leq X \leq x_1] = \int_{x_0}^{x_1} f_X(x') dx'. \quad (2.2)$$

Here, $\Pr[x_0 \leq X \leq x_1]$ denotes the probability that a random sample falls in the interval $[x_0, x_1]$. Denote by U a sample from the uniform distribution $\mathcal{U}(0, 1)$. The distribution of $F_X^{-1}(U)$ is then identical to the distribution of X and we are able to generate samples from F_X by sampling $U \sim \mathcal{U}(0, 1)$ and evaluating $Y = F_X^{-1}(U)$. See e.g., [89] for the simple proof.

This method does however require the cumulative distribution function and its inverse to be available and cheap to evaluate. Usually, this can be achieved by precomputing F_X^{-1} numerically and tabulating the values to generate a lookup table.

2.1.1.2. Rejection sampling

Another method that generates samples from a density f_X —requiring neither the evaluation of F_X , nor of F_X^{-1} —is rejection sampling. What is necessary, however, is a function that serves as an upper bound for f_X from which samples can be generated efficiently. The situation is visualized in Figure 2.1. As an example, the distribution from which we wish to sample is given by $f_X(x) = c \cdot (x - 1)^2 \cdot (x + 1)^2 \cdot \mathcal{N}_{0,1}(x)$ with $\mathcal{N}_{\mu,\sigma^2}(x)$ as the normal distribution with mean μ and variance σ^2 . The constant c normalizes f_X in the sense that $\int_{-\infty}^{\infty} f_X(x') dx' = 1$. An upper bound to $f_X(x)$ is $3 \cdot f_Y(x)$, with $f_Y(x) = \mathcal{N}_{0,9}(x)$.

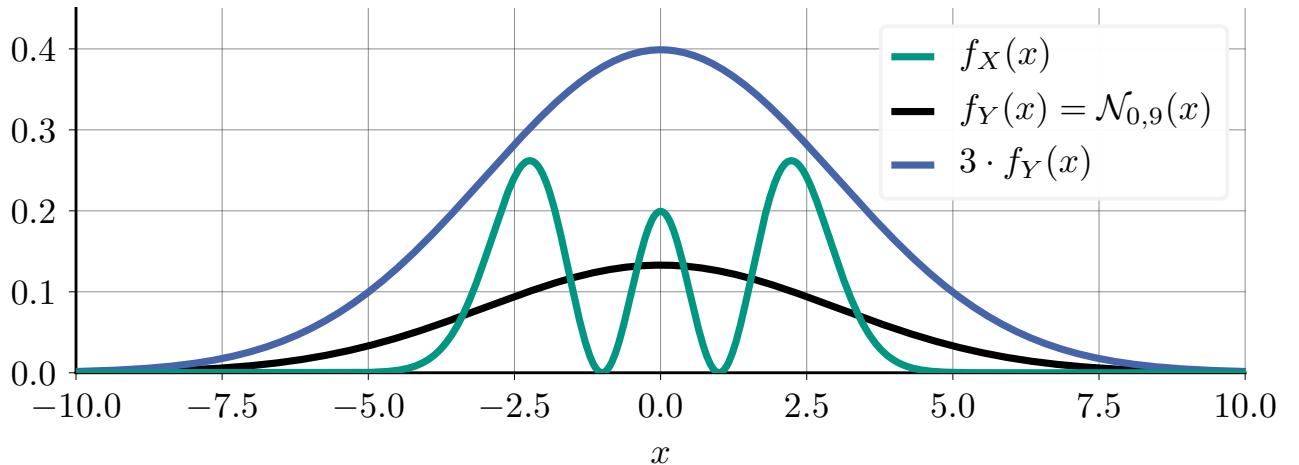


Figure 2.1.: To generate samples from f_X via rejection sampling, we use $3 \cdot f_Y$ as an upper bound. Here, f_Y is a normal distribution with mean $\mu = 0$ and variance $\sigma^2 = 9$.

To generalize, assume that our proposal distribution is $f_Y(x)$ and $M \cdot f_Y(x)$ is an upper bound to $f_X(x)$. Clearly, there exist multiple choices for M . In applications it is important to choose M such that $M \cdot f_Y$ is a tight upper bound to f_X (i.e., the situation in Figure 2.1 could be

optimized by choosing a constant slightly smaller than $M = 3$). An implementation² of rejection sampling is then given in Algorithm 2.1.

Algorithm 2.1 Rejection sampling.

```

1: function rejectionsampling( $f_X, f_Y, Y, M$ )
2:   while True do
3:      $y \leftarrow Y()$                                 Variable  $y$  is a sample from distribution  $Y$ .
4:      $u \leftarrow \mathcal{U}()$                             Variable  $u$  is a sample from  $\mathcal{U}(0, 1)$ .
5:     if  $u \cdot M \cdot f_Y(y) < f_X(y)$  then
6:       return  $y$                                 Variable  $y$  is also a sample from distribution  $X$ .

```

In Figure 2.2, we see the output of the rejection sampling algorithm. As indicated by the orange dashes at the bottom, the x values of the accepted points (in orange) are obeying the density f_X . However, the high number of rejected samples (in purple) means that it might potentially take a long time to generate the required number of samples. The number of rejected points equals 657, the number of accepted points is 343; this approximately agrees with the predicted acceptance rate of $1/M$.

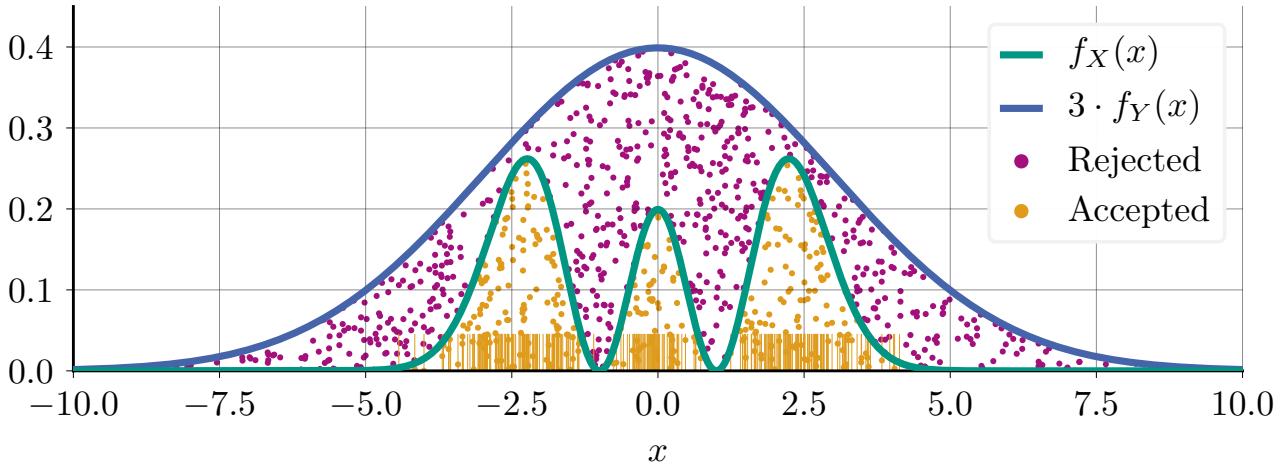


Figure 2.2.: Output of the rejection sampling algorithm. Purple points are being rejected. For the orange points, the corresponding x values are distributed according to f_X , illustrated by the orange dashes at the bottom. We observe that many points are being rejected, indicating that f_Y is no optimal choice.

It remains to show that Algorithm 2.1 does indeed generate samples that are distributed as prescribed by the density f_X .

Proof. We have to prove that the distribution of Y given that $u \cdot M \cdot f_Y(y) < f_X(y)$ is the distribution of X , i.e., $\Pr[y \leq y' \mid u \cdot M \cdot f_Y(y) < f_X(y)] = F_X(y')$. Note that

$$\Pr[u \cdot M \cdot f_Y(y) < f_X(y) \mid y = y'] = \frac{f_X(y')}{M \cdot f_Y(y')} \quad (2.3)$$

and thus

$$\Pr[u \cdot M \cdot f_Y(y) < f_X(y)] = \int_{-\infty}^{\infty} \frac{f_X(y')}{M \cdot f_Y(y')} f_Y(y') dy' = \frac{1}{M}. \quad (2.4)$$

²Algorithms presented in this thesis will obey a Python-ish style; instead of curly parentheses or closing `end` statements, we use indentations. Comments are right-aligned in gray.

A simple computation then completes the proof:

$$\Pr[y \leq y' | u \cdot M \cdot f_Y(y) < f_X(y)] \quad (2.5a)$$

$$= \Pr[u \cdot M \cdot f_Y(y) < f_X(y) | y \leq y'] \cdot \frac{\Pr[y \leq y']}{\Pr[u \cdot M \cdot f_Y(y) < f_X(y)]} \quad (2.5b)$$

$$= \Pr[u \cdot M \cdot f_Y(y) < f_X(y) | y \leq y'] \cdot \frac{F_Y(y')}{1/M} \quad (2.5c)$$

$$= \frac{\Pr[u \cdot M \cdot f_Y(y) < f_X(y), y \leq y']}{F_Y(y')} \cdot \frac{F_Y(y')}{1/M} \quad (2.5d)$$

$$= \int_{-\infty}^{y'} \Pr[u \cdot M \cdot f_Y(z) < f_X(z) | z \leq y'] f_Y(z) dz \cdot M \quad (2.5e)$$

$$= \int_{-\infty}^{y'} \frac{f_X(z)}{M \cdot f_Y(z)} f_Y(z) dz \cdot M \quad (2.5f)$$

$$= F_X(y'). \quad (2.5g)$$

□

Having discussed the main ingredients of the Monte Carlo method, we are now going to demonstrate how it can be used as a numerical integrator and a solution method for the transport equation.

2.1.1.3. Monte Carlo as a numerical integrator

Let $f : V \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$ be a Lebesgue-integrable function. The finite volume of V is given by $|V| = \int_V 1 d\mathbf{x}$. With I we denote the linear integral operator

$$I(f) := \int_V f(\mathbf{x}) d\mathbf{x}. \quad (2.6)$$

We wish to approximate $I(f)$ via Monte Carlo. For a random process, where \mathbf{x} has density $\rho(\mathbf{x})$, we define the expected value of $f(\mathbf{x})$ as

$$E[f] = \int_V f(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} \quad (2.7)$$

and the variance of $f(\mathbf{x})$ as

$$\text{Var}[f] = \int_V (f(\mathbf{x}) - E[f])^2 \rho(\mathbf{x}) d\mathbf{x}. \quad (2.8)$$

Let $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}$ be independent and identically distributed (iid) samples from $\mathcal{U}(V)$ and define

$$I^N(f) := \frac{|V|}{N} \sum_{i=1}^N f(\mathbf{x}^{(i)}) \quad (2.9)$$

as the numerical approximation of $I(f)$ via Monte Carlo. The expected value of $I^N(f)$ is exactly $I(f)$ since $E[I^N(f)] = \frac{|V|}{N} \sum_{i=1}^N E[f(\mathbf{x}^{(i)})] = \frac{|V|}{N} \sum_{i=1}^N \int_V f(\mathbf{x}^{(i)}) \frac{1}{|V|} d\mathbf{x}^{(i)} = I(f)$. Likewise, the resulting error $\varepsilon^N(f) := I(f) - I^N(f)$ is itself a random variable.

Using both the law of large numbers and the central limit theorem, the well known convergence result for Monte Carlo can be derived, i.e.,

$$\sqrt{E[(\varepsilon^N(f))^2]} = \sqrt{\frac{\text{Var}[f]}{N}}. \quad (2.10)$$

We can draw several conclusions from (2.10):

- The expected error decreases with order $N^{-1/2}$ as the number of samples N increases.
- The expected order of convergence is independent of the dimension.
- The expected error itself, however, might depend on f via $\text{Var}[f]$.

There exists a vast body of literature discussing the Monte Carlo method in great detail and several important aspects of Monte Carlo have been omitted so far (e.g., Quasi Monte Carlo [75], Markov Chain Monte Carlo [40], Zero-Variance Monte Carlo [7]). However, that is not to say we are not equipped with the necessary tools to finally discuss how the Monte Carlo method can be used to solve transport equations.

2.1.2. Monte Carlo for transport equations

The complexity of Monte Carlo simulations can increase almost arbitrarily, based upon the physical processes that should be incorporated into the model. These include, but are not limited to, the pair production in photon transport or the long-range interactions when simulating plasma.

Here, we will restrict ourselves to the description of time-independent, energy-dependent transport of uncharged particles, focusing on the important building blocks of the Monte Carlo method without including too much physical complexity (which ultimately differs from application to application).

Assume a domain $V \subset \mathbb{R}^d$ with $d \in \{2, 3\}$, boundary ∂V , and known cross sections $\sigma_s(\mathbf{x}, E)$ and $\sigma_a(\mathbf{x}, E)$ for every point $\mathbf{x} \in V$ and positive energies E . Particles enter the domain through boundary conditions $\psi^{bc}(\mathbf{x}, \Omega, E)$ for $\mathbf{x} \in \partial V$ and $\Omega \cdot \mathbf{n}(\mathbf{x}) < 0$. Here, $\mathbf{n}(\mathbf{x})$ is the outward-pointing normal vector at \mathbf{x} . For every (\mathbf{x}_0, Ω_0) , the cross sections along $\mathbf{x}_0 + s \Omega_0$ are computed with a ray-tracing algorithm up to the point $\mathbf{x}_0 + \tilde{s} \Omega_0$ where the particle would leave the domain again. Particles lose energy in two ways: (i) Via soft stopping interaction where particles lose energy based upon the distance that they travel through a given material [11]. Randomness is included in this process as well since the amount of energy that particles lose is sampled from a distribution. (ii) In case of a interactions, particles lose energy again (sampled from a different distribution).

The setting described here could, for example, describe radiation therapy. To treat cancer, radiation targets the cancerous tissue with the goal of damaging its DNA, leading to the cells' death. It is obvious that an accurate estimate of the absorbed energy is necessary. Too little energy will leave cancerous tissue intact, whereas too much energy will harm the surrounding, healthy tissue unnecessarily. Cross sections vary as a result of the body composition—the chance for particles to scatter in the lungs is smaller than in fat—and can be obtained via a CT scan a priori.

If we are now interested in the energy doses $D(V)$ that describes the amount of energy absorbed in voxel $v \subset V$, we can use the fact that Monte Carlo is a numerical integrator. A particle's

trajectory t , from *birth* at $(\mathbf{x}_b, \Omega_b, E_b)$ to *death* at $(\mathbf{x}_d, \Omega_d, E_d)$, is a random sample that might increase the amount of absorbed energy in v (if the trajectory passes through v). Henceforth, with $d_v(t_i)$ denoting the amount of energy that the i -th particle with trajectory t_i loses in v , we approximate

$$D(v) \approx D^N(v) := \frac{1}{N} \sum_{i=1}^N d_v(t_i). \quad (2.11)$$

The computational complexity lies in the computation of samples t_i . A particle has to be traced through the computational domain, path lengths and new directions need to be sampled at every collision, and energy needs to be tallied. For complicated distributions, inverse sampling (Section 2.1.1.1) might not be possible and samples need to be generated via rejection sampling (Section 2.1.1.2), rendering the process significantly more expensive.

2.1.3. Advantages and disadvantages of the Monte Carlo method

Let us summarize the advantages and disadvantages of the Monte Carlo method as a way to approximately solve the transport equation.

Advantages

The Monte Carlo method is *embarrassingly parallel*. Trajectories can be computed simultaneously and independently of another with marginal communication overhead, rendering the method predestined for parallel computers.

Since ray-tracing is not limited to a voxel-based geometry, arbitrary complex geometries can be represented exactly, albeit at increased computational cost.

Because the variance is an exact proxy for the expected error, Monte Carlo has a built-in error estimator. This error estimator can be used to terminate the generation of new trajectories once a prescribed accuracy is obtained.

The convergence order of Monte Carlo is $N^{-1/2}$, regardless of the dimensionality of the problem.

Disadvantages

The convergence order of Monte Carlo is *only* $N^{-1/2}$, regardless of the dimensionality of the problem. Listed as an advantage in the case of many dimensions, convergence order 1/2 is slow for a small to moderate number of dimensions.

Moreover, even though the *order* of the error is independent of the dimension, the error itself is not. Due to subregions of the domain that particles are unlikely to reach, the variance of the dose might be significant and is certainly not independent of the problem's dimensionality [95]. Almost always, variance reduction techniques need to be used to reduce the variance in domains with strong material heterogeneities.

2.2. Discrete ordinates method

The discrete ordinates (S_N) method is usually ascribed to Chandrasekhar [28] and work by Wick [100], dating back to the mid 1940s. Carlson and Lee discussed numerical quadratures for

the transport equation in 1961 [23]. Noteworthy, too, is Chandrasekhar's 1969 book *Radiative Transfer* [29]. In the preface of this book, he writes that the motivation and historic relevance of radiative transfer stems from astrophysics and investigations performed by Rayleigh in 1871. Nevertheless, his work impacts nuclear engineers and mathematicians as well. Convergence properties (under mild assumptions) were derived by Madsen in 1971 [65] and Anselone and Gibbs in 1974 [5].

At its core, the S_N method restricts the movements of particles to a fixed set of directions. The angular-dependent transport equation reduces to a set of angular-independent transport equations, coupled via scattering. The left-hand side of each equation is a simple advection equation that can be solved forward in time.

Ultimately, the spherical quadrature's job is to make the approximation

$$\int_{\mathbb{S}^2} \psi(t, \mathbf{x}, \boldsymbol{\Omega}') d\boldsymbol{\Omega}' \approx \sum_{q'=1}^{n_q} w_{q'} \psi(t, \mathbf{x}, \boldsymbol{\Omega}_{q'}) \quad (2.12)$$

as accurate as possible. For example, using isotropic scattering, this is the scattering operator on the transport equation's right-hand side. Henceforth, we will first discuss spherical quadratures before analyzing how the resulting system of coupled equations can be time-integrated.

2.2.1. Angular discretization

The fundamental idea of the S_N method is to approximate a high-dimensional equation by a lower-dimensional system of equations via discretization of the angular variable. This avoids solving the full linear transport equation (1.13) that reads

$$\begin{aligned} \partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_a \psi(t, \mathbf{x}, \boldsymbol{\Omega}) \\ = \sigma_s \int_{\mathbb{S}^2} s(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') (\psi(t, \mathbf{x}, \boldsymbol{\Omega}') - \psi(t, \mathbf{x}, \boldsymbol{\Omega})) d\boldsymbol{\Omega}' + q(t, \mathbf{x}, \boldsymbol{\Omega}). \end{aligned}$$

Since we are discussing a discretization in angle, the energy dependency is omitted. A spherical quadrature consists of a finite set of ordinates $\{\boldsymbol{\Omega}_1, \dots, \boldsymbol{\Omega}_{n_q}\} \subset \mathbb{S}^2$ and quadrature weights $\{w_1, \dots, w_{n_q}\}$, such that (2.12) is as accurate as possible.

The discrete ordinates method demands (1.13) to be satisfied only at the set of ordinates $\{\boldsymbol{\Omega}_1, \dots, \boldsymbol{\Omega}_{n_q}\}$ and deploys (2.12) to circumvent the spherical integral on the right-hand side. The S_N approximation to the linear transport equation is then

$$\begin{aligned} \partial_t \psi_q(t, \mathbf{x}) + \boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}} \psi_q(t, \mathbf{x}) + \sigma_a \psi_q(t, \mathbf{x}) \\ = \sigma_s \sum_{q'=1}^{n_q} w_{q'} s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_{q'}) (\psi_{q'}(t, \mathbf{x}) - \psi_q(t, \mathbf{x})) + q(t, \mathbf{x}, \boldsymbol{\Omega}_q), \end{aligned} \quad (2.13)$$

for $q = 1, \dots, n_q$ and with $\psi_q(t, \mathbf{x}) := \psi(t, \mathbf{x}, \boldsymbol{\Omega}_q)$. The left-hand side of (2.13) is a set of n_q linear advection equations that could be solved independently if it were not for the right-hand side. The coupling is a result of the discretized scattering operator. We can rewrite (2.13) more compactly as

$$L\psi(t, \mathbf{x}) + \sigma_a \psi(t, \mathbf{x}) = \sigma_s (S^+ - S^-) \psi(t, \mathbf{x}) + q(t, \mathbf{x}), \quad (2.14)$$

with $\psi(t, \mathbf{x}) = (\psi_1(t, \mathbf{x}), \dots, \psi_{n_q}(t, \mathbf{x}))^T$, the linear transport operator $L : \mathbb{R}^{n_q} \rightarrow \mathbb{R}^{n_q}$ that satisfies $L_{q,q'} = \delta_{q,q'} (\partial_t + \boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}})$, and the linear in- and out-scattering operators $S^+ : \mathbb{R}^{n_q} \rightarrow \mathbb{R}^{n_q}$ satisfying $S_{q,q'}^+ = w_{q'} s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_{q'})$ and $S^- : \mathbb{R}^{n_q} \rightarrow \mathbb{R}^{n_q}$ with $S_{q,q'}^- = \delta_{q,q'} \sum_{p=1}^{n_q} w_p s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_p)$, respectively. The source $\mathbf{q}(t, \mathbf{x})$ is defined analogously to the angular flux vector $\psi(t, \mathbf{x})$. Since the spherical quadrature does not necessarily guarantee $\sum_{q'=1}^{n_q} w_{q'} s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_{q'}) = 1$ (which is the case for the exact spherical integration), we can move out-scattering to the left-hand side before applying the S_N discretization to obtain

$$L\psi(t, \mathbf{x}) + \sigma_t \psi(t, \mathbf{x}) = \sigma_s S^+ \psi(t, \mathbf{x}) + \mathbf{q}(t, \mathbf{x}). \quad (2.15)$$

Recalling (1.18), in the case of the Henyey-Greenstein scattering kernel, we can expand s_g in terms of spherical harmonics as

$$s_g(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) = \frac{1 - g^2}{4\pi (1 - 2g \boldsymbol{\Omega}' \cdot \boldsymbol{\Omega} + g^2)^{3/2}} = \sum_{n=0}^{\infty} g^n \sum_{m=-n}^n Y_n^m(\boldsymbol{\Omega}) \overline{Y_n^m(\boldsymbol{\Omega}')}$$

Truncating the first sum at N , we get an approximation to the Henyey-Greenstein scattering kernel with a finite number of expansion coefficients

$$s_g(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \approx s_g^N(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) := \sum_{n=0}^N g^n \sum_{m=-n}^n Y_n^m(\boldsymbol{\Omega}) \overline{Y_n^m(\boldsymbol{\Omega}')}. \quad (2.16)$$

For a fixed value of N , this approximation is more accurate when $|g|$ is close to zero (scattering is almost isotropic) and less accurate when $|g|$ is close to one (scattering is peaked forward or backward). If we substitute the approximation (2.16) into (2.15), the in-scattering matrix S^+ has entries

$$S_{q,q'}^+ = w_{q'} \sum_{n=0}^N g^n \sum_{m=-n}^n Y_n^m(\boldsymbol{\Omega}_q) \overline{Y_n^m(\boldsymbol{\Omega}_{q'})}. \quad (2.17)$$

Define the matrices

$$O \in \mathbb{R}^{n_q \times N^2}, \quad O_{q,\text{idx}(n,m)} = Y_n^m(\boldsymbol{\Omega}_q), \quad (2.18a)$$

$$\Sigma \in \mathbb{R}^{N^2 \times N^2}, \quad \Sigma_{\text{idx}(n,m),\text{idx}(n,m)} = g^n, \quad (2.18b)$$

$$M \in \mathbb{R}^{N^2 \times n_q}, \quad M_{\text{idx}(n,m),q'} = \overline{Y_n^m(\boldsymbol{\Omega}_{q'})}, \quad (2.18c)$$

with $\text{idx}(n, m) = n \cdot (N + 1) + m + 1$, $0 \leq n \leq N$, and $-n \leq m \leq n$. Then the approximated in-scattering matrix S^+ with entries given by (2.17) can be decomposed into $S^+ = O\Sigma M$.

We will discuss two methods to solve the coupled system of equations in Sections 2.2.4 and 2.2.5. Additionally, we will elaborate how the decomposition $S^+ = O\Sigma M$ can be used to speed up the computation. The choice of the quadrature is crucial for the S_N method to work and will be discussed in the next section.

Remark. The truncation order N has to be high when scattering is highly peaked to have an accurate approximation. Thus, the decomposition is usually only used when the scattering is close to isotropic, since then the expansion coefficients g^n decay sufficiently fast.

Remark. The expansion $S^+ = O\Sigma M$ can similarly be formulated for different choices of the scattering kernel s . However, in the case of the Henyey-Greenstein phase function, the decomposition is easy to derive analytically. This is one of the arguments for choosing the Henyey-Greenstein phase function to model scattering.

Weight function	Interval	Orthogonal polynomials
$w(x) = 1$	$[-1, 1]$	Legendre
$w(x) = 1/\sqrt{1-x^2}$	$(-1, 1)$	Chebyshev (1 st kind)
$w(x) = \sqrt{1-x^2}$	$[-1, 1]$	Chebyshev (2 nd kind)
$w(x) = e^{-x}$	$[0, \infty)$	Laguerre
$w(x) = e^{-x^2}$	$(-\infty, \infty)$	Hermite

Table 2.1.: Table of weight functions and corresponding orthogonal polynomials.

2.2.2. Quadrature rules in one dimension

Because several quadrature rules are based upon one-dimensional quadrature rules, we will start by summarizing some of the classical results regarding numerical integration in one spatial dimension before continuing with spherical integration. Additionally, one-dimensional quadrature rules are necessary for applying the S_N method to the transport equation in slab geometry.

In general, one-dimensional quadratures try to approximate the bounded integral

$$\int_a^b f(x)w(x) dx, \quad (2.19)$$

using a sum of n_q finite terms via

$$\sum_{q=1}^{n_q} w_q f(x_q). \quad (2.20)$$

To do this as accurately as possible, both the quadrature points and the quadrature weights need to be chosen carefully. While it is obvious that a quadrature with n_q quadrature points can integrate (2.19) for f being a polynomial up to degree $n_q - 1$, Gaussian quadratures are special in the sense that they correctly integrate polynomials up to degree $2n_q - 1$. Gaussian quadratures use orthogonal polynomials that, for a given weight function w , are defined by the relation

$$\langle P_i, P_j \rangle_w := \int_a^b P_i(x)P_j(x)w(x) dx = \delta_{i,j}. \quad (2.21)$$

For different weight functions w , some orthogonal polynomials are given in Table 2.1.

Definition 2.2.1 (Gaussian quadratures). *A Gaussian quadrature is a quadrature rule that approximates the integral $\int_a^b f(x)w(x) dx$ by the finite sum $\sum_{q=1}^{n_q} w_q f(x_q)$. The quadrature points x_1, \dots, x_{n_q} are the roots of the n_q -th orthogonal polynomial P_{n_q} , defined via (2.21). The corresponding quadrature weights are defined by*

$$w_q = \frac{1}{F'(x_q)} \int_a^b \frac{F(x)w(x)}{x - x_q} dx, \quad (2.22)$$

with

$$F(x) = \prod_{q=1}^{n_q} (x - x_q). \quad (2.23)$$

Theorem 2.2.1 (Optimality of Gaussian quadratures). *A Gaussian quadrature is optimal in the sense that it approximates $\int_a^b \pi(x)w(x) dx$ for polynomials $\pi(x)$ up to including degree $2n_q - 1$ (but generally not higher) exactly, using only n_q quadrature points and n_q quadrature weights.*

Proof. See, e.g., Chandrasekhar [29]. □

Remark. Although Gaussian quadratures are optimal, other quadratures might be superior in certain applications [94].

2.2.3. Spherical quadrature rules

All quadrature sets try to achieve (2.12) sufficiently well. However, not all of them are necessarily suitable for the S_N method. Several properties are desirable:

1. In the limit of $n_q \rightarrow \infty$, (2.12) should become an equality.
2. Quadrature weights and nodes should be easily computable; even up to a high number.
3. Quadrature points should be spaced equidistantly on the sphere for the method to be invariant under rotation. (At least for 90° rotations around the x -, y -, and z -axis this is often a desirable property.)
4. Quadrature weights should have a small variance. This is a result of the quadrature points being spaced equidistantly.
5. It is often demanded that the quadrature correctly integrates spherical harmonics or Legendre polynomials up to a certain degree. This potentially contradicts the third point. For example, we can use specific knowledge about the first N spherical harmonics that we want to integrate correctly to *intelligently* place quadrature points, sacrificing rotational invariance.
6. The quadrature should allow for (possibly adaptive) refinement strategies.
7. Quadrature weights should be positive. This might be in conflict with property five, but is desirable to ensure positivity of the S_N method.

The first two properties are self-evident. The third property can be explained when considering, to name only a simple example, an anisotropic, highly forward-peaked source in an almost void domain. In this advective setting, particles will predominantly move along straight lines in accordance with the source. If, however, the dominant direction is unknown a priori, no direction should be preferred by the quadrature set. The best (and sometimes only) guess is to space quadrature points equidistantly. Point four relates to the previous point since a lack of preference in direction immediately dictates a lack of preference in the weight associated to that direction. The fifth point is closely related to (2.17). Finally, (adaptive) refinement in the angular discretization allows to leverage multi-level strategies or to dynamically increase the angular accuracy in regions of interest [90, 31].

Next, we are going to provide examples of common spherical quadrature sets and investigate both their theoretical and practical properties.

Computing $\int_0^{2\pi} \int_0^\pi Y_j^i(\theta, \phi) \overline{Y_l^k(\theta, \phi)} \sin(\theta) d\theta d\phi$ with the level-symmetric quadrature of order 20.

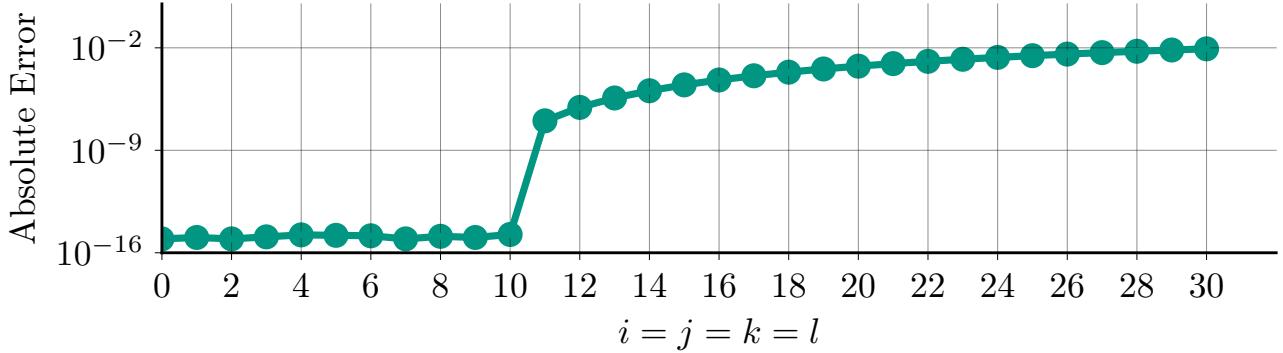


Figure 2.3.: For a fixed quadrature order N , the level-symmetric quadrature computes the integral exactly (up to machine precision) if $i + k \leq N$.

Example 2.2.1 (Level-symmetric quadrature). The level-symmetric quadrature set dates back to the early and mid 1960s [23, 60, 24]. It is a spherical quadrature set that is invariant under 90° rotations around the x -, y -, and z -axis³. For a fixed, even number N , we pick $N/2$ points along the x -axis $x_1, \dots, x_{N/2}$. These points are replicated along the other two axes, i.e., $x_i = y_i = z_i$ for $i = 1, \dots, N/2$. The set of quadrature points is then $\{(x_i, y_j, z_k)^T \mid i + j + k = N/2 + 2\}$. If $p_1 := (x_i, y_j, z_k)^T \in \mathbb{S}^2$ is a point of the level-symmetric quadrature, then for $p_2 := (x_i, y_{j+1}, z_{k'})^T$ to be a point of the level-symmetric quadrature as well, we have to require $k' = k - 1$ due to symmetry. From $\|p_1\|_2 = \|p_2\|_2 = 1$ follows

$$y_{j+1}^2 - y_j^2 = z_{k+1}^2 - z_k^2 \quad (2.24a)$$

$$\Rightarrow x_{j+1}^2 - x_j^2 = x_{k+1}^2 - x_k^2, \quad (2.24b)$$

for all feasible values of j and k , implying $x_{j+1}^2 = x_j^2 + C$ for $j = 1, \dots, N/2 - 1$. The variable C is determined by enforcing that $x_1^2 + y_1^2 + z_{N/2}^2 = 1$ which follows from the fact that $(x_1, y_1, z_{N/2})$ is part of the quadrature set. Lastly, we choose the quadrature weights in such a way that they correctly integrate the Legendre polynomials in each of the variables [64]. Following through with this requirement, individual quadrature weights turn negative for $N \geq 22$. Figures 2.3 and 2.4 demonstrate that $\int_0^\pi \int_0^{2\pi} Y_j^i(\theta, \phi) \overline{Y_l^k(\theta, \phi)} \sin(\phi) d\theta d\phi = \delta_{i=k} \delta_{j=l}$ is integrated correctly (with machine precision) up to a certain degree.

Example 2.2.2 (Tensorized Gauss-Legendre). The tensorized (or product) Gauss-Legendre quadrature stems from the idea of rewriting a spherical integral in spherical coordinates, i.e.,

$$\int_{\mathbb{S}^2} f(\Omega) d\Omega = \int_0^{2\pi} \int_0^\pi \hat{f}(\theta, \phi) \sin(\theta) d\theta d\phi. \quad (2.25)$$

Abusing notation, we drop the superscript hat in the integrand from now onward. The integral can then be approximated via

$$Q^{n_q}[f] := \frac{\pi}{n_q} \sum_{j=1}^{2n_q} \sum_{i=1}^{n_q/2} w_i f(\theta_i, \phi_j), \quad (2.26)$$

³This property is common and allows us to only provide quadrature points for one of the eight octants since all other quadrature points and weights follow from symmetry.

Computing $\int_0^{2\pi} \int_0^\pi Y_8^8(\theta, \phi) \overline{Y_8^8(\theta, \phi)} \sin(\theta) d\theta d\phi$ with the level-symmetric quadrature of varying order.

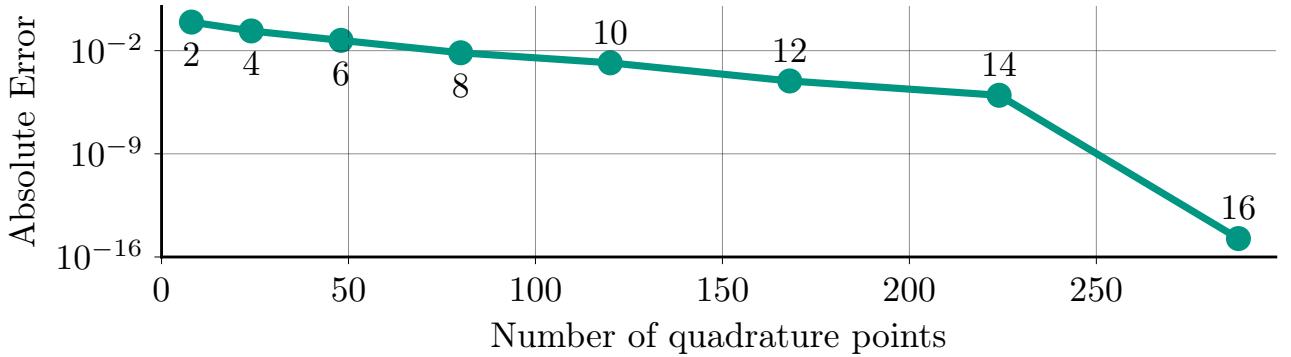


Figure 2.4.: The level-symmetric quadrature becomes more accurate the higher the number of quadrature points gets. Eventually, the integral is computed exactly (up to machine precision). The number of quadrature points grows quadratically with the order.

Computing $\int_0^{2\pi} \int_0^\pi Y_j^i(\theta, \phi) \overline{Y_l^k(\theta, \phi)} \sin(\theta) d\theta d\phi$ with the Gauss-Legendre quadrature of order 20.

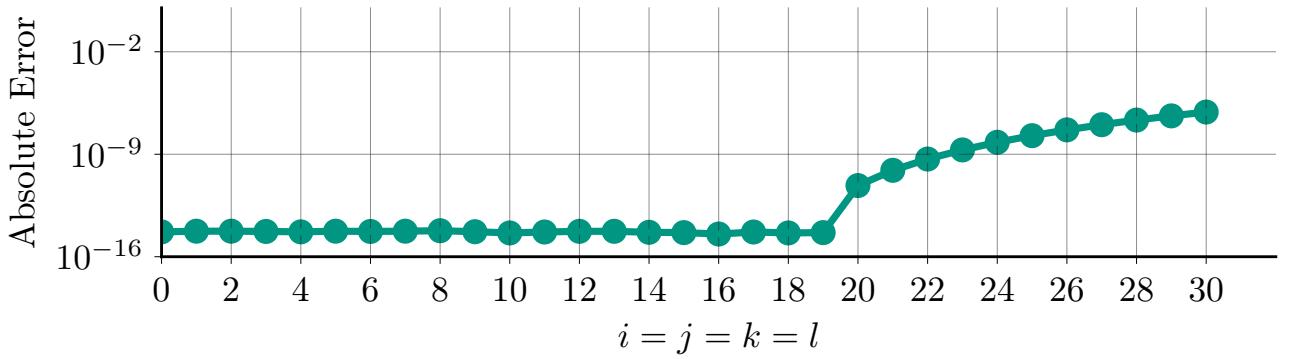


Figure 2.5.: For a fixed quadrature order N , the level-symmetric quadrature computes the integral exactly (up to machine precision) if $i + k \leq 2N - 1$.

where the ϕ_j are spaced equidistantly along the unit circle, and the θ_i and w_i are, respectively, the quadrature points and weights of the one-dimensional Gauss-Legendre quadrature. The degree of the aforementioned quadrature is $2n_q - 1$ [8], demonstrated in Figure 2.5. The quadrature is *tensorized* in the sense that the spherical quadrature points are products of two one-dimensional quadratures—and so are the quadrature weights. That the Gauss-Legendre quadrature is not designed under consideration of uniform quadrature weights is exemplified in Figure 2.6 where we see that the ratio between the maximal quadrature weight and the minimal quadrature weight diverges. A more detailed discussion of the Gauss-Legendre quadrature can be found in the work by Atkinson [8] and the literature therein.

Example 2.2.3 (Octahedron- and icosahedron-based quadratures with connectivity). In Camminady et al. [22], we present highly uniform quadratures that are purely geometric and result from a triangulation of an octahedron or, alternatively, an icosahedron—the Platonic solids with eight or twenty triangular faces, respectively. Platonic solids are regular, convex polyhe-

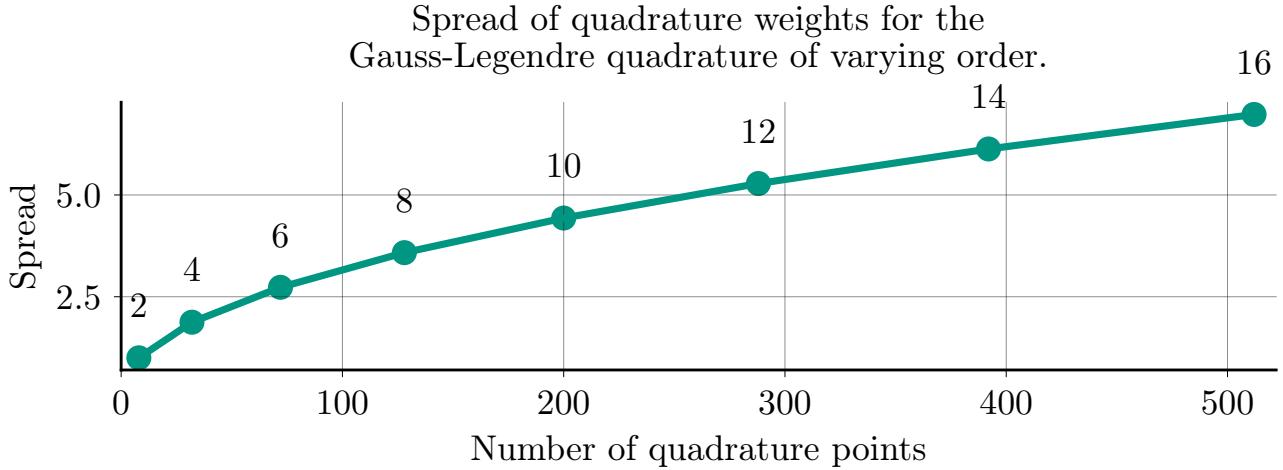


Figure 2.6.: The spread of the quadrature weights, i.e., the ratio between the maximal weight and the minimal weight, diverges for the Gauss-Legendre quadrature. This also holds true in the three-dimensional case, when using a tensorized quadrature that is based on the Gauss-Legendre points and weights. Nevertheless, the Gauss-Legendre quadrature's capability to satisfy the fifth point on the list of desirable properties makes it one of the most commonly used quadratures.

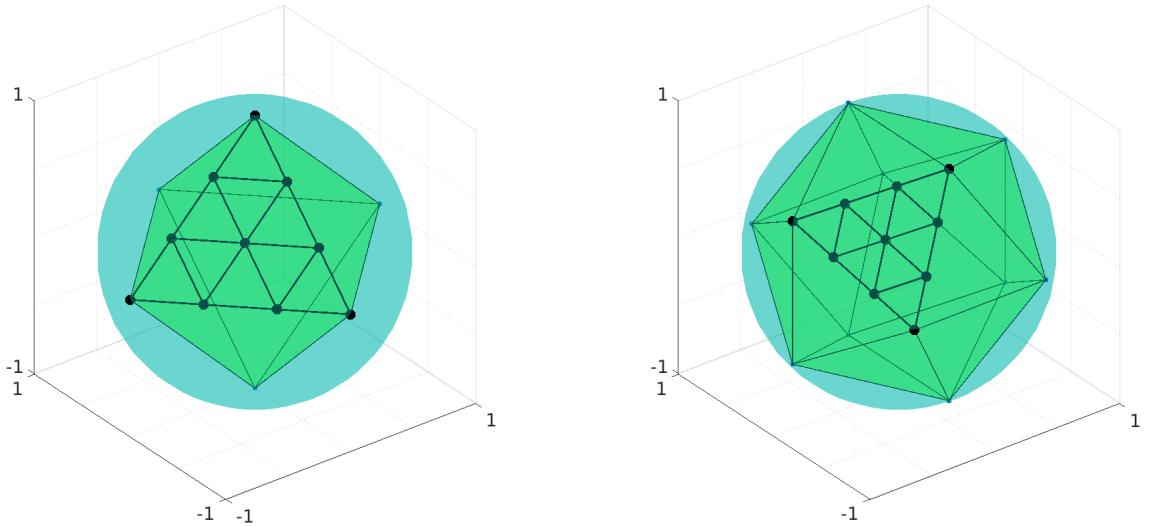
dra with identical faces, all having the same area. Projecting from these platonic solids onto the unit sphere yields highly uniformly distributed points. This idea has been used within climate forecasting [85] where a uniform discretization of the earth's atmosphere is desirable.

The triangulation can be performed by linear interpolation (lerp) or spherical linear interpolation (slerp), demonstrated for the lerp version in Fig. 2.7a for the octahedron, and in Fig. 2.7b for the icosahedron. Linear interpolation places points with equidistant spacing in planar geometry, whereas spherical linear interpolation places the points equidistantly on the sphere. Given two points $p_0, p_1 \in \mathbb{R}^3$, we perform spherical linear interpolation via

$$\text{slerp}(p_0, p_1, t) = \frac{\sin((1-t)\Omega)}{\sin(\Omega)}p_0 + \frac{\sin(t\Omega)}{\sin(\Omega)}p_1,$$

where $\cos(\Omega) = p_0 \cdot p_1$. Consequently, we obtain a total of four quadrature sets: O_N^l , O_N^s , I_N^l , and I_N^s . Here, O and I distinguish between the octahedron and the icosahedron version, and l and s between the lerp and slerp version, respectively. Since the difference in the construction between the octahedron and the icosahedron version is minor, we focus on the octahedron version. Similar to the T_N quadrature [93], the planar area is the equilateral triangle in three dimension with vertices $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$, presented in Figure 2.8a. Refining the triangulation in the planar setting and then projecting each vertex onto the sphere yields the quadrature points for one of the eight octants, shown in Figure 2.8b.

The quadrature weights correspond to the area associated with each quadrature point. This area is the hexagon that is defined by connecting the midpoints of the six triangles that every vertex is associated with. As an example, the quadrature weight for the centered point in Figure 2.8b is given by the area of the green hexagon on the unit sphere. In contrast, the T_N method takes the triangle midpoints as the quadrature points and the associated triangle area as the quadrature weight. The slerp version of the quadrature can be constructed analogously by performing the linear interpolation, i.e. the refinement of the triangulation, on the sphere instead of in the planar geometry.



(a) Octahedron within the unit sphere, used to generate the O_N^l quadrature.

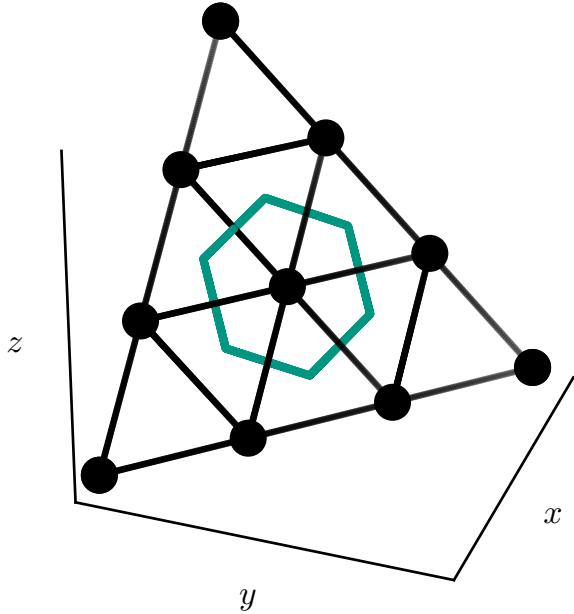
(b) Icosahedron within the unit sphere, used to generate the I_N^l quadrature

Figure 2.7.: Generation of the quadrature sets. One face of the octahedron and icosahedron have been further refined. Vertices will be projected onto the unit sphere and taken as quadrature points.

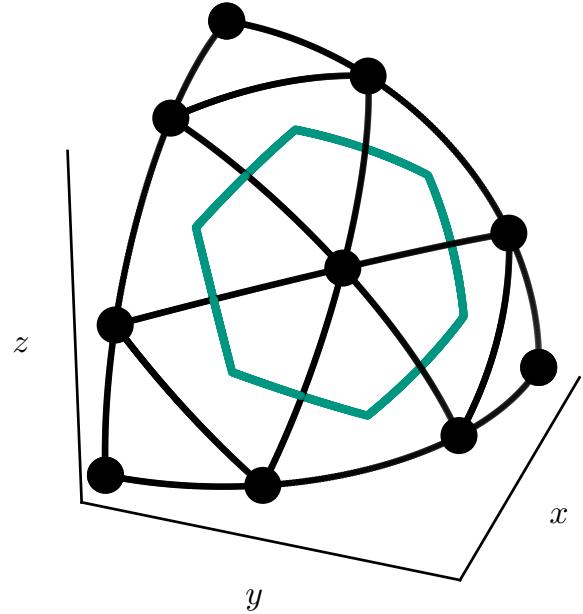
To obtain the icosahedron versions of the quadratures, the planar geometry is replaced by one face of an icosahedron.

For the aforementioned quadratures, properties of the quadrature weights are presented in Figure 2.10. As expected, the I_N versions have lower variance in the quadrature weights and smaller ratios between the maximal and minimal quadrature weights. Additionally, the slerp versions have smaller ratios and variance than the lerp versions. Combining these two effects, the I_N^l quadrature behaves similar to the O_N^s quadrature, whereas the O_N^l has the highest ratios and variance and the I_N^s quadrature the lowest ratios and variance.

Finally, Figure 2.9 shows the distribution of quadrature weights on the unit sphere for all four quadrature sets. Notably, the slerp versions are less symmetric than their lerp counterparts. This is due to the fact that the slerp interpolation does not treat all three sides of an equilateral triangle in the same way, resulting in a loss of perfect symmetry. Furthermore, since the quadrature points at the vertices of the platonic solid only neighbor four (in the octahedron case) or five (in the icosahedron case) triangles as opposed to six triangles for every other quadrature point, their quadrature weight is significantly lower.



(a) For the lerp version, the triangulation happens in planar geometry.



(b) Quadrature points result from projecting the planar points onto the unit sphere.

Figure 2.8.: Triangulation for order = 4 with the corresponding quadrature points in the plane and on the surface of the unit sphere, together with the connectivity and the quadrature weight for the center point in green. Quadrature points and weights for the seven other octants result from symmetry. Figure 2.8a demonstrates the lerp procedure. For the slerp quadratures, the points in Figure 2.8b would have to be spaced equally.

2.2.3.1. Comparison and source code

Various spherical quadratures are implemented in the Python package `SphericalQuadpy`, available at <https://github.com/camminady/sphericalquadpy> under the MIT license. The library provides quadrature points and weights for different orders and different quadrature types. A comparison to other state of the art quadratures has been published [22], indicating the octahedron and icosahedron quadratures' high degree of uniformity (with respect to the distribution of quadrature points in the sphere). However, different from other quadratures, the icosahedron and octahedron quadratures were not generated with the goal to integrate certain types of functions exactly. The comparison includes quadratures that were generated with the idea to ensure positivity of quadrature weights [93]. Other, more advanced ideas additionally emphasize high order convergence when integrating spherical harmonics and are based on finite elements [15] or discontinuous finite elements [49, 61].

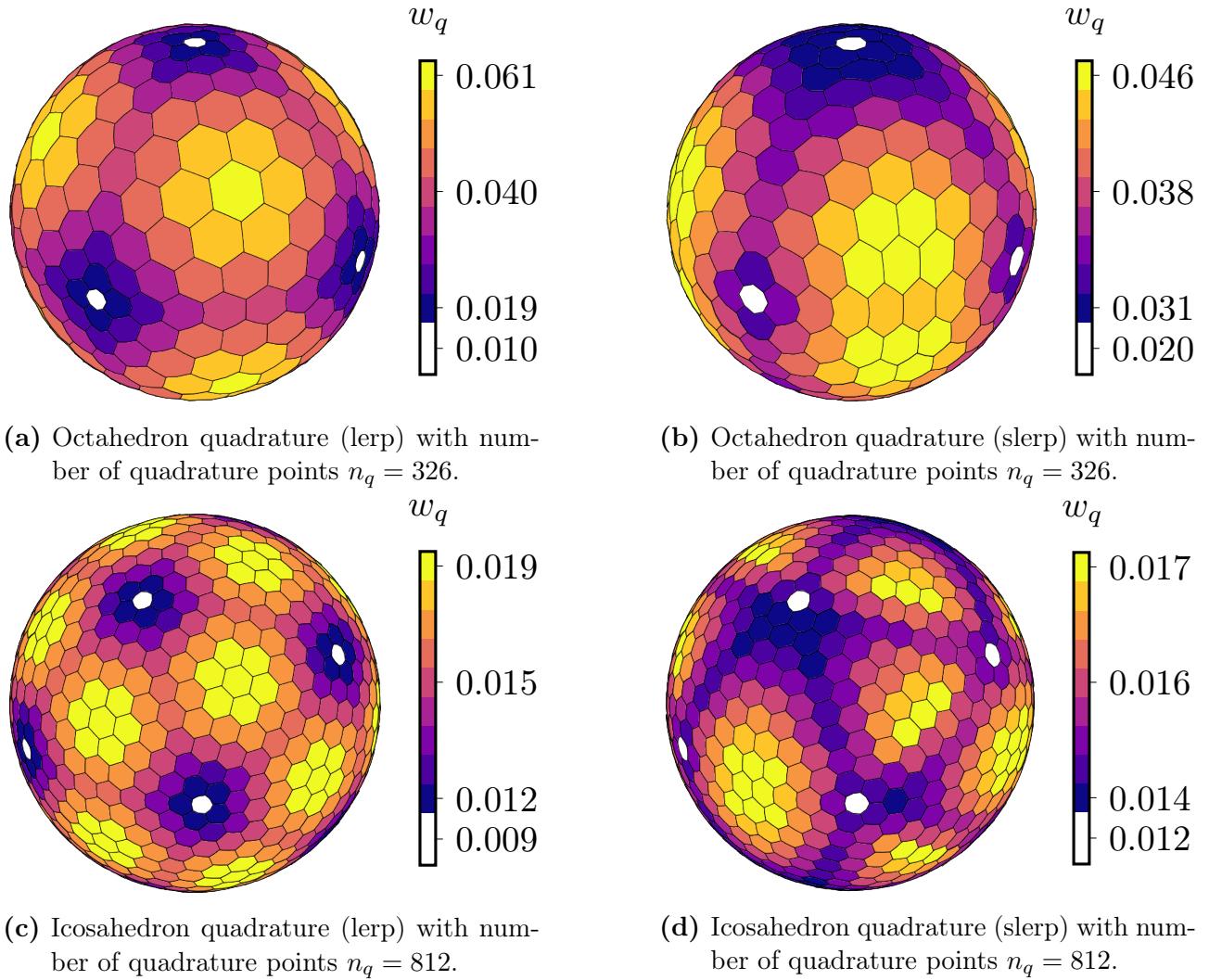


Figure 2.9.: Octahedron and icosahedron quadrature. Due to the implementation of the spherical linear interpolation, the slerp versions of the quadratures have less symmetry than the lerp versions. They are, however, not as spread in their weight distribution. The smallest quadrature weights are associated with the vertices of the respective platonic solids. Since their quadrature weights are significantly lower, they are excluded from the colorbar and highlighted in white to avoid distortion.

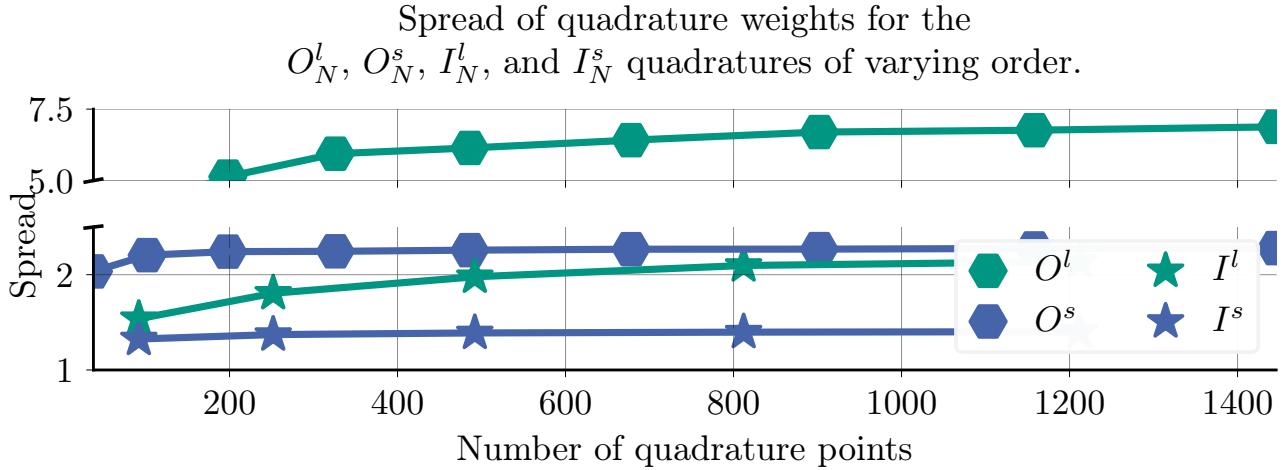


Figure 2.10.: The spread of the quadrature weights, i.e., the ratio between the maximal weight and the minimal weight, diverges for the octahedron and icosahedron quadratures. Note the gap in the y -axis, inserted to make it possible to include all four graphs.

2.2.4. Finite volume schemes

Spherical quadrature sets are used for the angular discretization of the transport equation. It remains to discretize the space and time dimension. Let us recall the S_N discretization (2.13), given by

$$\begin{aligned} \partial_t \psi_q(t, \mathbf{x}) + \boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}} \psi_q(t, \mathbf{x}) + \sigma_a \psi_q(t, \mathbf{x}) \\ = \sigma_s \sum_{q'=1}^{n_q} w_{q'} s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_{q'}) (\psi_{q'}(t, \mathbf{x}) - \psi_q(t, \mathbf{x})) + q(t, \mathbf{x}, \boldsymbol{\Omega}_q). \end{aligned}$$

Besides the coupling term on the right-hand side, this is a simple linear advection equation which we can solve using the finite volume method, explained in the following for order one. For a Cartesian mesh, define $C_{i,j,k} := [x_i, x_{i+1}] \times [y_j, y_{j+1}] \times [z_k, z_{k+1}]$ with volume $|C_{i,j,k}| := \Delta x_i \cdot \Delta y_j \cdot \Delta z_k$, where $\Delta \xi_i := \xi_{i+1} - \xi_i$ for $\xi \in \{x, y, z\}$. The indices i , j , and k run from 1 to n_x , n_y , and n_z , respectively. Time is discretized equidistantly such that $t^n := \Delta t \cdot n$ for some $\Delta t > 0$ with $n = 0, \dots, n_t$. Quintessentially, the finite volume method approximates the solution of the transport equation by cell-averaged quantities for which an update rule is based on the fluxes over the cell edges. These cell-averaged quantities are

$$\psi_{q;i,j,k}^n \approx \frac{1}{|C_{i,j,k}|} \int_{C_{i,j,k}} \psi_q(t^n, \mathbf{x}) d\mathbf{x}, \quad (2.27)$$

and the update rule for Cartesian grids can be written as

$$\begin{aligned} \psi_{q;i,j,k}^{n+1} = \psi_{q;i,j,k}^n - \frac{\Delta t}{|C_{i,j,k}|} \left(f_{q;i+1/2,j,k} - f_{q;i-1/2,j,k} + f_{q;i,j+1/2,k} - f_{q;i,j-1/2,k} + f_{q;i,j,k+1/2} - f_{q;i,j,k-1/2} \right) \\ - \Delta t \left(\sigma_a \psi_{q;i,j,k}^n - \sigma_s \sum_{q'=1}^{n_q} w_{q'} s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_{q'}) (\psi_{q';i,j,k}^n - \psi_{q;i,j,k}^n) - q_{q;i,j,k}^n \right), \end{aligned} \quad (2.28)$$

where $f_{q;i+1/2,j,k}$ represents the flux between cells $C_{i,j,k}$ and $C_{i+1,j,k}$, given a fixed direction $\boldsymbol{\Omega}_q$. All other terms in (2.28) can be interpreted analogously. If the outward facing normal vector

for cell $C_{i,j,k}$ at the edge touching cell $C_{i+1,j,k}$ is denoted by \mathbf{n} , the upwind flux formulation for a fixed direction Ω_q is given by

$$f_{q;i+1/2,j,k} = \Delta x_i \langle \mathbf{n}, \Omega_q \rangle \cdot \begin{cases} \psi_{q;i,j,k}^n & \text{if } \langle \mathbf{n}, \Omega_q \rangle \geq 0, \\ \psi_{q;i+1,j,k}^n & \text{if } \langle \mathbf{n}, \Omega_q \rangle < 0. \end{cases} \quad (2.29)$$

Since this method is explicit, the time steps can become so small that the associated computational costs render this time-stepping scheme inapplicable.

2.2.5. Source iteration and transport sweeps

Iterative methods for the numerical solution of transport processes often make use of source iteration to solve the transport equation. To illustrate the core idea of source iteration and transport sweeps, we restrict ourselves to mono-energetic, time-independent, and isotropic transport. The equation we wish to solve—in angularly discretized form—is then

$$\Omega_q \cdot \nabla_{\mathbf{x}} \psi_q(\mathbf{x}) + \sigma_t \psi_q(\mathbf{x}) = \sigma_s \sum_{q'=1}^{n_q} w_{q'} \frac{1}{4\pi} \psi_{q'}(\mathbf{x}) + q(\mathbf{x}, \Omega_q), \quad (2.30)$$

for all $q = 1, \dots, n_q$, abbreviated as

$$L_q \psi_q = S^+(\psi) + q_q, \quad (2.31)$$

with the streaming operator $L_q = \Omega_q \cdot \nabla_{\mathbf{x}} + \sigma_t$, in-scattering $S^+(\psi) = \sigma_s \sum_{q'=1}^{n_q} w_{q'} \frac{1}{4\pi} \psi_{q'}(\mathbf{x})$, and source $q_q = q(\mathbf{x}, \Omega_q)$. We may also write $L\psi = S^+(\psi) + \mathbf{q}$ to group the n_q equations of type (2.31) together. Source iteration then iterates on

$$L\psi^{(l+1)} = S^+(\psi^{(l)}) + \mathbf{q}, \quad l \geq 0, \quad (2.32)$$

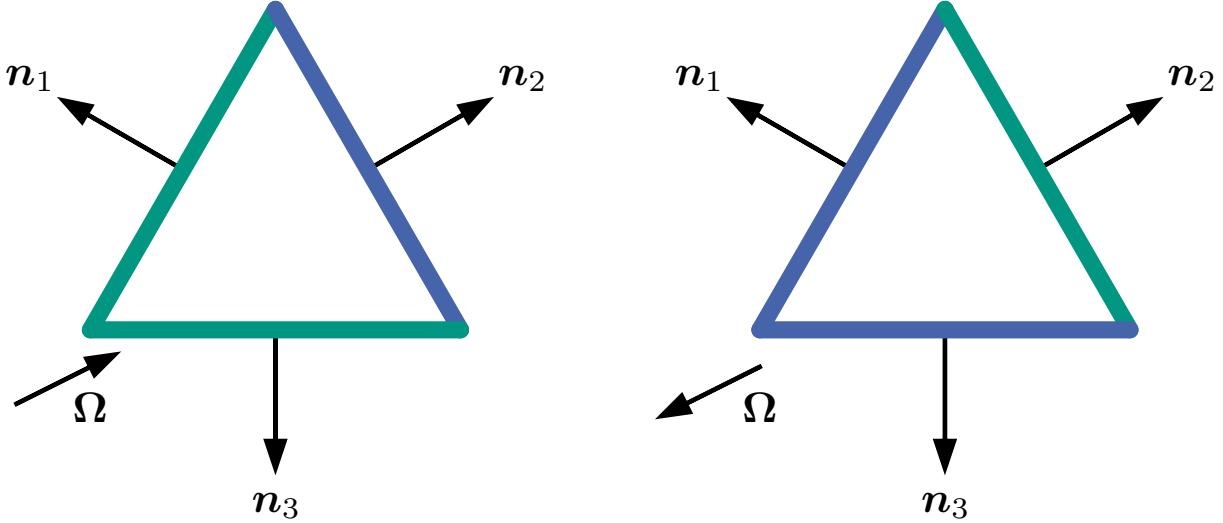
with some initial $\psi^{(0)}$. This fixed-point iteration is repeated until the solution is approximated reasonably well, i.e., $\|\psi^{(l)} - \psi^{\text{exact}}\| \leq \varepsilon/(1 - C)$ for some norm $\|\cdot\|$, Lipschitz-constant C , and prescribed tolerance ε . Physically, $\psi^{(l)}$ represents the contributions of particles that have scattered at most l times, which consequently outlines the problem of source iteration in regions of marginal absorption and high scattering: Convergence will be relatively slow. A comprehensive overview of the possible acceleration strategies is summarized in the review paper of Adams and Larsen [2]. However, instead of discussing ways to accelerate the convergence of (2.31), we will discuss how to actually solve one iteration of (2.31). For a fixed l , (2.31) becomes a system of n_q equations with a given right-hand side, given by

$$L_q \psi_q^{(l+1)} = S^+(\psi^{(l)}) + q_q =: r_q^{(l)}. \quad (2.33)$$

It remains to show how to solve this equation, which can be rewritten for the two-dimensional case as

$$\Omega_x \partial_x \psi(\mathbf{x}) + \Omega_y \partial_y \psi(\mathbf{x}) + \sigma_t \psi(\mathbf{x}) = r(\mathbf{x}). \quad (2.34)$$

We omitted the direction-index q because we assume fixed values for Ω_x and Ω_y and follow the analysis from Lewis and Miller [64]. Introducing a triangulation of the spatial domain that



- (a) Two inflow edges (green) and one outflow edge (blue) since $\mathbf{n}_1 \cdot \mathbf{\Omega} < 0$ and $\mathbf{n}_3 \cdot \mathbf{\Omega} < 0$.
(b) One inflow edge (green) and two outflow edges (blue) since only $\mathbf{n}_2 \cdot \mathbf{\Omega} < 0$.

Figure 2.11.: Two possible situations for sweeping through a domain of triangles: There are either two inflow edges (green) and one outflow edge (blue), or one inflow edge and two outflow edges.

consists of triangular cells C_i with boundary ∂C_i and outward pointing normal vector \mathbf{n} , we can integrate (2.34) over C_i and use the divergence theorem to obtain the equality

$$\oint_{\partial C_i} \mathbf{\Omega} \cdot \mathbf{n}(\mathbf{s}) \psi(\mathbf{s}) d\mathbf{s} + \sigma_t \int_{C_i} \psi(\mathbf{x}) d\mathbf{x} = \int_{C_i} r(\mathbf{x}) d\mathbf{x}. \quad (2.35)$$

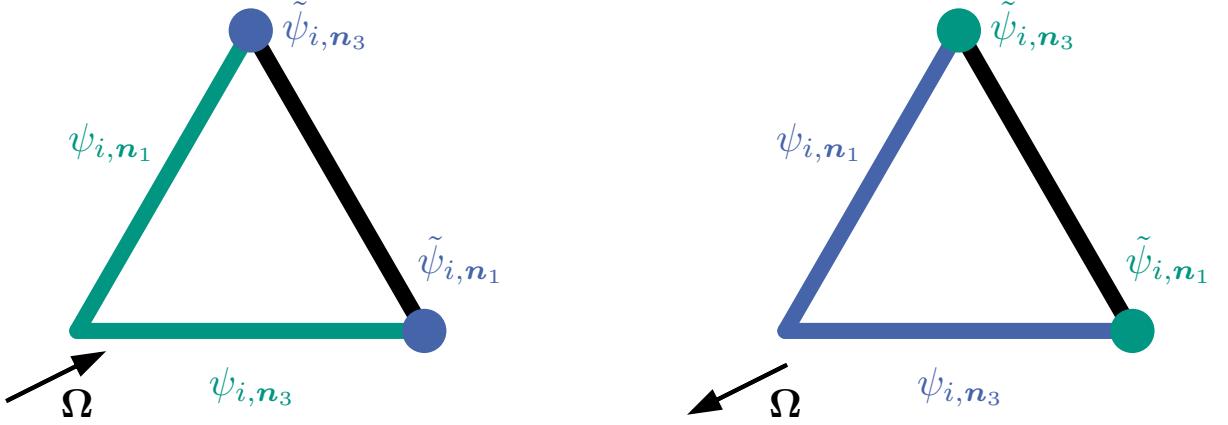
Acknowledging that the normal vectors are constant along the three edges and using cell-, as well as boundary-averaged quantities, we can rewrite the above equation as

$$\sum_{j=1}^3 \mathbf{\Omega} \cdot \mathbf{n}_j \frac{|C_{i,j}|}{|C_i|} \psi_{i,\mathbf{n}_j} + \sigma_t \psi_i = r_i, \quad (2.36)$$

where \mathbf{n}_j are the three normal vectors of cell C_i with area $|C_i|$. The boundary-averaged fluxes are given by

$$\psi_{i,\mathbf{n}_j} := \frac{1}{|C_{i,j}|} \oint_{C_{i,j}} \psi(\mathbf{s}) d\mathbf{s} \quad (2.37)$$

for each side of the triangle $C_{i,j}$ with length $|C_{i,j}|$. Lastly, ψ_i and r_i are the respective averaged cell integrals. There has been no spatial approximation made so far. Thus, (2.36) is exact. As discussed by Lewis and Miller [64] and visualized in Figure 2.12, two cases need to be distinguished: There are either two inflow edges and one outflow edge, or one inflow edge two outflow edges. We also assume the triangles to be equilateral ($C_{i,j} = L$) since we can simply map the equilateral triangle to an arbitrary triangle via a linear transformation (and vice versa). In Figure 2.12a, the ansatz $\psi_i = \frac{1}{3} \sum_{j=1}^3 \psi_{i,\mathbf{n}_j}$ then allows to uniquely solve (2.36) for the only unknown ψ_i via elimination of ψ_{i,\mathbf{n}_2} since values for ψ_{i,\mathbf{n}_1} and ψ_{i,\mathbf{n}_3} are known from the cells



(a) Values at the two green edges are known and we wish to solve for values at the blue vertices.

(b) Values at the two green vertices are known and we wish to solve for values at the blue edges.

Figure 2.12.: We augment the situation by one additional variable. For triangles with two inflow edges, we solve for flux values at the two outflow vertices. These two outflow vertices are inflow vertices for the adjacent triangle. Known variables are green, unknown variables blue.

that proceed cell i in the sweeping order. This is not possible for the second scenario, where, as shown in Figure 2.12b, only ψ_{i,n_2} is known and the system is therefore underdetermined.

Counterintuitive at first, we can overcome this problem when augmenting the situation by an additional variable, illustrated in Figure 2.11. We again distinguish two different situations. In Figure 2.11a, two edges are inflow edges and we know the respective averaged flux values. In this situation, the two unknowns are the flux values at the two vertices of the outflow edge, which are taken as known inputs for the situation in Figure 2.11b. This situation's unknowns are the averaged flux values along the two outflow edges. Due to the way that the triangles are connected, the two known variables are always given from a proceeding triangle.

Since we have an additional variable, we also need an additional equation. If we assume the flux to be linear in a given triangle, we can write the averaged fluxes as

$$\psi_{i,n_1} = \frac{1}{2} (\tilde{\psi}_{i,n_2} + \tilde{\psi}_{i,n_3}), \quad (2.38a)$$

$$\psi_{i,n_2} = \frac{1}{2} (\tilde{\psi}_{i,n_1} + \tilde{\psi}_{i,n_3}), \quad (2.38b)$$

$$\psi_{i,n_3} = \frac{1}{2} (\tilde{\psi}_{i,n_1} + \tilde{\psi}_{i,n_2}), \quad (2.38c)$$

where the flux at the vertex opposite of edge j is denoted by $\tilde{\psi}_{i,n_j}$. Together with $\psi_i = \frac{1}{3} \sum_{j=1}^3 \psi_{i,n_j}$, fluxes along edges and at vertices relate to the cell average via

$$2\psi_{i,n_j} = 3\psi_i - \tilde{\psi}_{i,n_j}. \quad (2.39)$$

Finally, we are able to write out the update rule for both cases. For two inflow edges and one outflow edge we can solve (2.36) for ψ_n via elimination of ψ_{i,\mathbf{n}_2} and obtain

$$\psi_i = \frac{1}{3w_2 + \sigma_t} (\psi_{i,\mathbf{n}_1}(w_2 - w_1) + \psi_{i,\mathbf{n}_3}(w_2 - w_3) + r_i), \quad (2.40)$$

where $w_j := \Omega \cdot \mathbf{n}_j \frac{|C_{i,j}|}{|C_i|}$. The fluxes at the two outflow vertices ψ_{i,\mathbf{n}_1} and ψ_{i,\mathbf{n}_3} are then computed by means of (2.39). Similarly, for the case of two outflow edges and one inflow edge we substitute $\psi_{i,\mathbf{n}_1} = \frac{3}{2}\psi_i - \frac{1}{2}\tilde{\psi}_{i,\mathbf{n}_1}$ and $\psi_{i,\mathbf{n}_3} = \frac{3}{2}\psi_i - \frac{1}{2}\tilde{\psi}_{i,\mathbf{n}_3}$ in (2.40) and rearrange terms to get

$$\psi_i = \frac{1}{3(w_1 + w_3) + 2\sigma_t} (\tilde{\psi}_{i,\mathbf{n}_1}(w_1 - w_2) + \tilde{\psi}_{i,\mathbf{n}_1}(w_3 - w_2) + 2r_i). \quad (2.41)$$

Fluxes of outflow edges can again be computed by means of (2.39).

So far we have implicitly assumed that we can iterate through the spatial domain in a way that obeys the dependency between cells. Phrased more precisely: If two connected cells C_i and C_j share edge $e_{i,j}$, and $e_{i,j}$ is an outflow edge for C_i and an inflow edge for C_j , we have to perform computations in C_i prior to computations in C_j . It remains to prove that this ordering does indeed exist for a regular, triangular mesh. We use the following theorem and two definitions from graph theory.

Definition 2.2.2 (Topological sorting). *A topological sorting of a directed graph $G = (V, E)$ is a linear ordering of vertices where V_i precedes V_j whenever there exists a directed edge $(V_i, V_j) \in E$.*

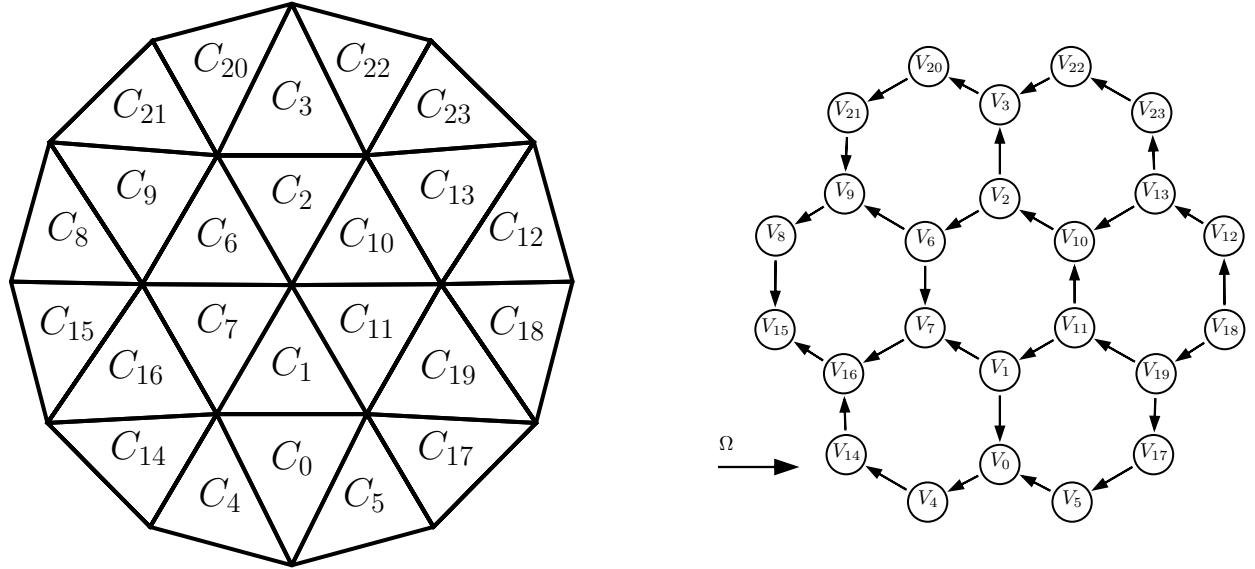
Definition 2.2.3 (Directed acyclic graph). *A graph $G = (V, E)$ is called a directed acyclic graph (DAG), if there exists no sequence of vertices $V_1, V_2, \dots, V_N, V_1$ where all two consecutive vertices V_i and V_{i+1} are connected by a directed edge $(V_i, V_{i+1}) \in E$.*

Theorem 2.2.2 (Topological sorting for DAGs). *Every directed acyclic graph has at least one topological sorting.*

Proof. See the algorithm by Kahn [52] that creates a topological sorting with an asymptotic runtime in $\mathcal{O}(|V| + |E|)$. □

Sweeping is therefore possible, if—given a fixed direction Ω —the induced dependency graph $G = (V, E)$ is acyclic.

Theorem 2.2.3 (Sweeping is possible for a triangular mesh). *Consider a domain that is discretized by a set of cells $\{C_i\}_{i=1,\dots,I}$ where each cell C_i is a triangle and we do not allow for hanging nodes, exemplified in Figure 2.13a. Furthermore, we have a fixed direction Ω that prescribes the flow of information. Under these conditions, sweeping is possible.*



(a) A computational domain, subdivided into spatial cells.

(b) Given a fixed direction Ω , we can translate the spatial domain into a graph.

Figure 2.13.: For a given spatial discretization, a direction Ω induces a directed dependency graph $G = (V, E)$. A directed edge $e_{i,j} \in E$ from V_i to V_j implies that for computations to start in cell C_j , we must have finished computations in cell C_i .

Proof. We know that sweeping is possible, if and only if the induced dependency graph $G = (V, E)$ is acyclic. Now assume the dependency graph is not acyclic and provoke a contradiction.

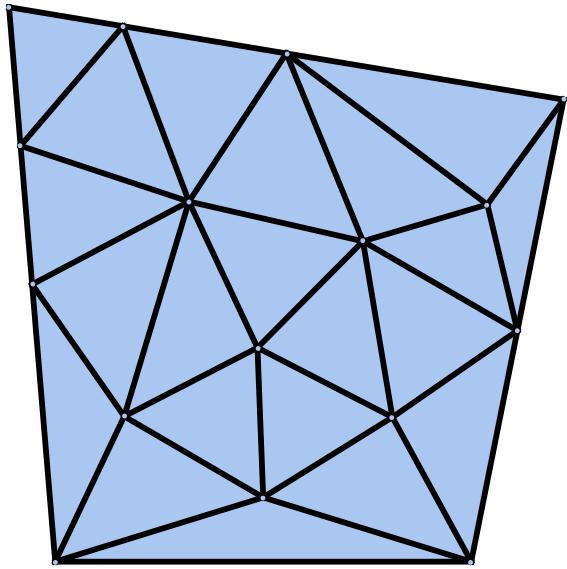
If the dependency graph G is not acyclic, there exists an ordered sequence of cells $\{C_{i_1}, C_{i_2}, \dots, C_{i_M}, C_{i_1}\}$, such that two consecutive cells share an edge $e_{i_j, i_{j+1}}$ and $\mathbf{n}_{i_j} \cdot \Omega > 0, \forall j = 1, \dots, M$. Without loss of generality, let $i_j = j$ and assume that we pass through the cells in a counterclockwise manner as shown in Figure 2.14b for the spatial domain of Figure 2.14a.

Now label the angles between \mathbf{n}_i and \mathbf{n}_{i+1} by α_i , shown in Figure 2.15a. A counterclockwise turn corresponds to $\alpha_i > 0$ and a clockwise turn to $\alpha_i < 0$. We know that $-\pi < \alpha_i < \pi$, because we consider regular triangles. Since we perform a full counterclockwise turn, $\sum_{i=1}^M \alpha_i = 2\pi$. Denote the angle between \mathbf{n}_1 and Ω^\perp by θ as sketched in Figure 2.15b, with $0 < \theta < \pi$ and Ω^\perp the vector normal to Ω . Let R_{α_i} be the rotation matrix that encodes rotating with magnitude α_i around the z -axis. Then

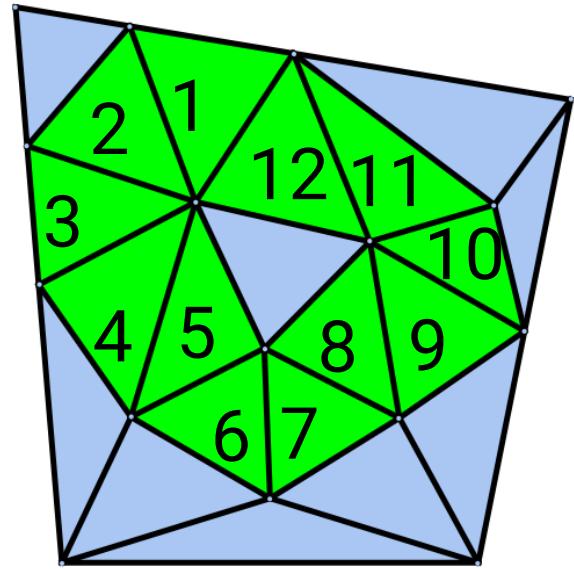
$$\mathbf{n}_{i+1} = R_{\alpha_i} \mathbf{n}_i = \prod_{k=1}^i R_{\alpha_k} \mathbf{n}_1 = R_{\sum_{k=1}^i \alpha_k} \mathbf{n}_1.$$

If we turn \mathbf{n}_1 (counterclockwise) by more than θ but less than $\theta + \pi$, then $\mathbf{n}_1 \cdot \Omega < 0$. However, there exist i^* such that $\sum_{k=1}^{i^*-1} \alpha_k \leq \theta$, but $\theta < \sum_{k=1}^{i^*} \alpha_k < \theta + \pi < 2\pi$, as $-\pi < \alpha_{i^*} < \pi$. Then \mathbf{n}_{i^*+1} is \mathbf{n}_1 turned (counterclockwise) by more than θ , but less than $\theta + \pi$. Therefore $\mathbf{n}_{i^*+1} \cdot \Omega < 0$ which contradicts the assumption and finishes the proof. This proof is available as a preprint [19].

□

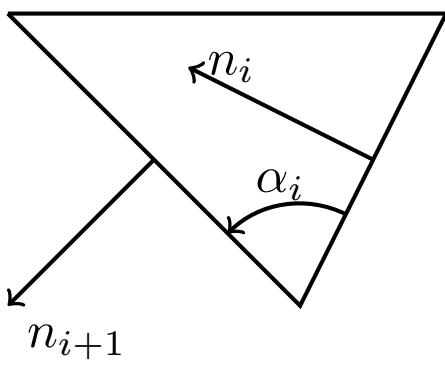


(a) A spatial domain that is discretized by triangles.

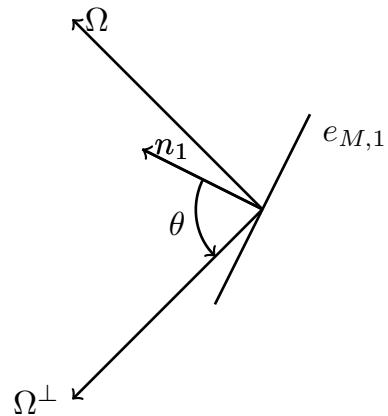


(b) A possible sequence of cells that form a circle.

Figure 2.14.: For sweeping to work, there can not be a circular dependency in the mesh.



(a) Nomenclature for inflow and outflow edges.



(b) The normal \mathbf{n}_1 , with Ω and Ω^\perp , as well as the angle θ between \mathbf{n}_1 and Ω^\perp .

Figure 2.15.: Sketches for the proof of Theorem 2.2.3.

In general, allowing arbitrary convex quadrilaterals for the triangulation of a computational domain does not imply the absence of circular dependencies. This is sketched in Figure 2.16. Quadrilaterals can be arranged in a plane, perpendicular to the z -axis such that they form a circular dependency for a direction of flow along the z -axis.

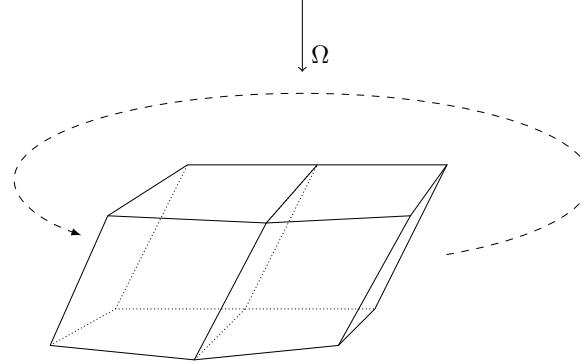


Figure 2.16.: Quadrilaterals can be arranged in a circle such that no topological ordering can be obtained for $\Omega = (0, 0, -1)^T$.

2.2.5.1. Source iteration for time dependent problems

It is straightforward to include time in (2.30), resulting in

$$\partial_t \psi_q(t, \mathbf{x}) + \boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}} \psi_q(t, \mathbf{x}) + \sigma_t \psi_q(t, \mathbf{x}) = \sigma_s \sum_{q'=1}^{n_q} w_{q'} \frac{1}{4\pi} \psi_{q'}(t, \mathbf{x}) + q(t, \mathbf{x}, \boldsymbol{\Omega}_q), \quad (2.42)$$

or—when time is discretized implicitly—in

$$\boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}} \psi_q^{n+1}(\mathbf{x}) + \left(\sigma_t + \frac{1}{\Delta t} \right) \psi_q^{n+1}(\mathbf{x}) = \sigma_s \sum_{q'=1}^{n_q} w_{q'} \frac{1}{4\pi} \psi_{q'}^{n+1}(\mathbf{x}) + q^{n+1}(\mathbf{x}, \boldsymbol{\Omega}_q) + \frac{1}{\Delta t} \psi_q^n(\mathbf{x}). \quad (2.43)$$

The superscript $n + 1$ denotes the unknown quantities evaluated at time $t^{n+1} = (n + 1) \cdot \Delta t$ and $\psi_q^n(\mathbf{x})$ is known from the previous time step. We can again abbreviate this formulation, this time via

$$\tilde{L}_q \psi_q^{n+1} = S^+(\boldsymbol{\psi}^{n+1}) + q_q^{n+1}, \quad (2.44)$$

where $\tilde{L}_q := \boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}} + \left(\sigma_t + \frac{1}{\Delta t} \right)$ and $q_q^{n+1} := q^{n+1}(\mathbf{x}, \boldsymbol{\Omega}_q) + \frac{1}{\Delta t} \psi_q^n(\mathbf{x})$. In-scattering is again denoted by S^+ . This formulation is equivalent to (2.33) with different cross section and source. We can therefore use source iteration and transport sweeps in the exact same manner, just repeatedly to march forward in time. Source iteration at time step $n + 1$ can be initialized with the angular flux of the previous step, i.e., $\psi^{n+1(0)} := \psi^{n(l)}$ where l denotes the number of transport sweeps at time step n .

2.2.6. Advantages and disadvantages of the discrete ordinates method

We start by discussing the advantages of the S_N method first, followed by its disadvantages.

Advantages

Unlike the Monte Carlo method, the discrete ordinates method is deterministic and does not suffer from statistical noise and the slow $1/\sqrt{N}$ convergence rate.

The S_N method also preserves positivity of the angular flux, which moment methods are generally not expected to do due to Gibbs-like oscillations. However, this can only be ensured for positive quadrature weights and Cartesian meshes.

The fact that the choice of the underlying quadrature set is up to the user can be considered a double-edged sword: Poor choices may reinforce numerical artifacts whereas apt, problem-dependent choices can increase the solution quality without an increase in computational cost or memory.

Adaptivity can be used in time and space (due to its relation to advection equations), as well as in angle [90, 31]. For example, in regions with near-vacuum properties, i.e., absent scattering, the set of ordinates can be aligned with the particles' initial directions of travel.

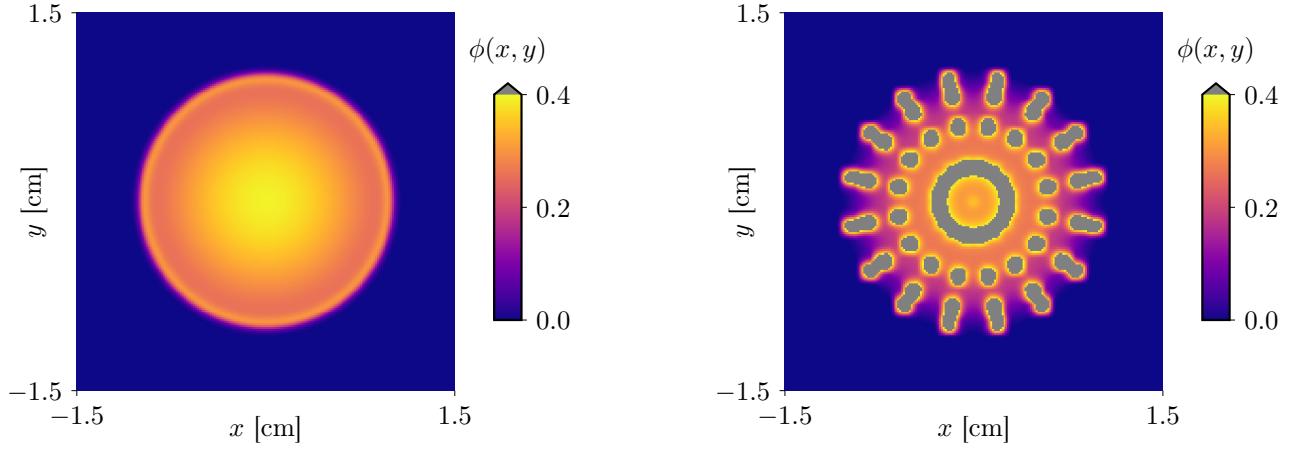
While the Monte Carlo method is embarrassingly parallel, the discrete ordinates method is harder to parallelize. However, parallelization in angle is relatively easy since, given a precomputed right-hand side, source iteration has no interdependence between the different directions.

Disadvantages

On the other hand, certain disadvantages prevail. Space, time, and angle need to be discretized⁴ in the discrete ordinates method, whereas all three variables are continuous in the Monte Carlo method. Different from space and time discretization, the discretization of angle introduces an error that is not blatantly obvious. So called *ray effects* [60] result from restricting transport to a finite set of angular directions, illustrated in Figure 2.17 for the line-source test case. Here, particles are emitted isotropically in the center of a domain at $t = 0$ and move away from the origin as time progresses. We will discuss this test case more thoroughly later on and it will serve as an important benchmark when analyzing modified S_N methods. In a nutshell, this test case examines whether or not a numerical method is able to reproduce a radially symmetric solution where the scalar flux is a function of the distance to the origin only. Clearly, this is not the case for the discrete ordinates method since density fluctuations are undeniable. Refining the angular resolution does resolve these fluctuations at the cost of increased run time and memory consumption. Two methods that mitigate ray effects without adding more directions, called rS_N and $as-S_N$, will be discussed in Sections 3.2 and 3.3, respectively.

Three further problems stem from the usage of source iteration and transport sweeps as the solution algorithm. Parallelizing transport sweeps by domain decomposition, difficult due to the dependency between cells, is a topic of current research and becomes especially crucial for highly performant codes on large-scale clusters. Furthermore, in addition to the cross sections, the isotropy of the scattering kernel influences the convergence of source iteration. A scattering kernel that models predominantly forward-peaked scattering requires fewer transport sweeps than an isotropic scattering kernel. This is no surprise, however, since forward-peaked scattering means that particle mostly advect through the spatial domain without change of direction. Lastly, source iteration might be too slow for typical reactor physics problems and acceleration strategies need to be used to overcome the slow convergence.

⁴Energy, too, is being discretized. We focus on mono-energetic transport throughout this thesis and therefore omit the consequences of discretizing energy.



(a) Reference solution for the scalar flux of the line-source problem. The solution solely depends upon the distance to the origin, not the angle. The scalar flux is bounded from above (approximately) 0.4.

(b) Solution for the scalar flux of the line-source problem using the discrete ordinates method. Symmetry is clearly broken. Violations of the upper bound are masked in gray.

Figure 2.17.: Ray effects for the line-source problem occur due to the finite set of angular directions.

2.3. Moment methods

The P_N method is a Galerkin-type method that expands the angular dependency in the transport equation in terms of spherical harmonics. The spherical harmonics are a complete set of orthogonal functions on the unit sphere. There exist different definitions of the spherical harmonics that vary in the respective normalization constants. One possible definition is

$$Y_n^m(\theta, \phi) := \sqrt{\frac{2n+1}{4\pi} \frac{(n-m)!}{(n+m)!}} e^{im\theta} P_n^m(\cos(\phi)), \quad (2.45)$$

with $|m| \leq n$, the azimuthal angle $\theta \in [0, 2\pi)$, the polar angle $\phi \in [0, \pi]$, and P_n^m as the associated Legendre functions. This choice ensures

$$\langle Y_n^m, Y_{n'}^{m'} \rangle_{\mathbb{S}^2} := \int_0^{2\pi} \int_0^\pi Y_n^m(\theta, \phi) \overline{Y_{n'}^{m'}(\theta, \phi)} \sin(\phi) d\phi d\theta = \delta_{n,n'} \delta_{m,m'}, \quad (2.46)$$

i.e., orthonormality with respect to integration over the unit sphere. Since the azimuthal angle θ and the polar angle ϕ uniquely define a point Ω on the unit sphere via the relation $\Omega = (\cos(\theta) \sin(\phi), \sin(\theta) \sin(\phi), \cos(\phi))^T$, we abbreviate $Y_n^m(\theta, \phi) = Y_n^m(\Omega)$. The real and imaginary parts of the spherical harmonics up to $m = 3$ are visualized in Figure 2.18 and Figure 2.19, respectively.

The spherical harmonics relate to the Legendre polynomials via the *addition theorem*. For $\Omega, \Omega' \in \mathbb{S}^2$ and $l \in \mathbb{N}^{\geq 0}$ the equality

$$P_n(\Omega \cdot \Omega') = \frac{4\pi}{2n+1} \sum_{|m| \leq n} Y_n^m(\Omega) \overline{Y_n^m(\Omega')} \quad (2.47)$$

allows to express the Legendre polynomials in form of the spherical harmonics, given an argument that is the dot product of two vectors living on the unit sphere. Since the scattering kernel is frequently evaluated at the dot product between the in- and out-scattering directions, the *addition theorem* can be applied to write

$$s(\Omega \cdot \Omega') = \sum_{n=0}^{\infty} s_n \sum_{|m| \leq n} Y_n^m(\Omega) \overline{Y_n^m(\Omega')}, \quad (2.48)$$

with expansion coefficients s_n .

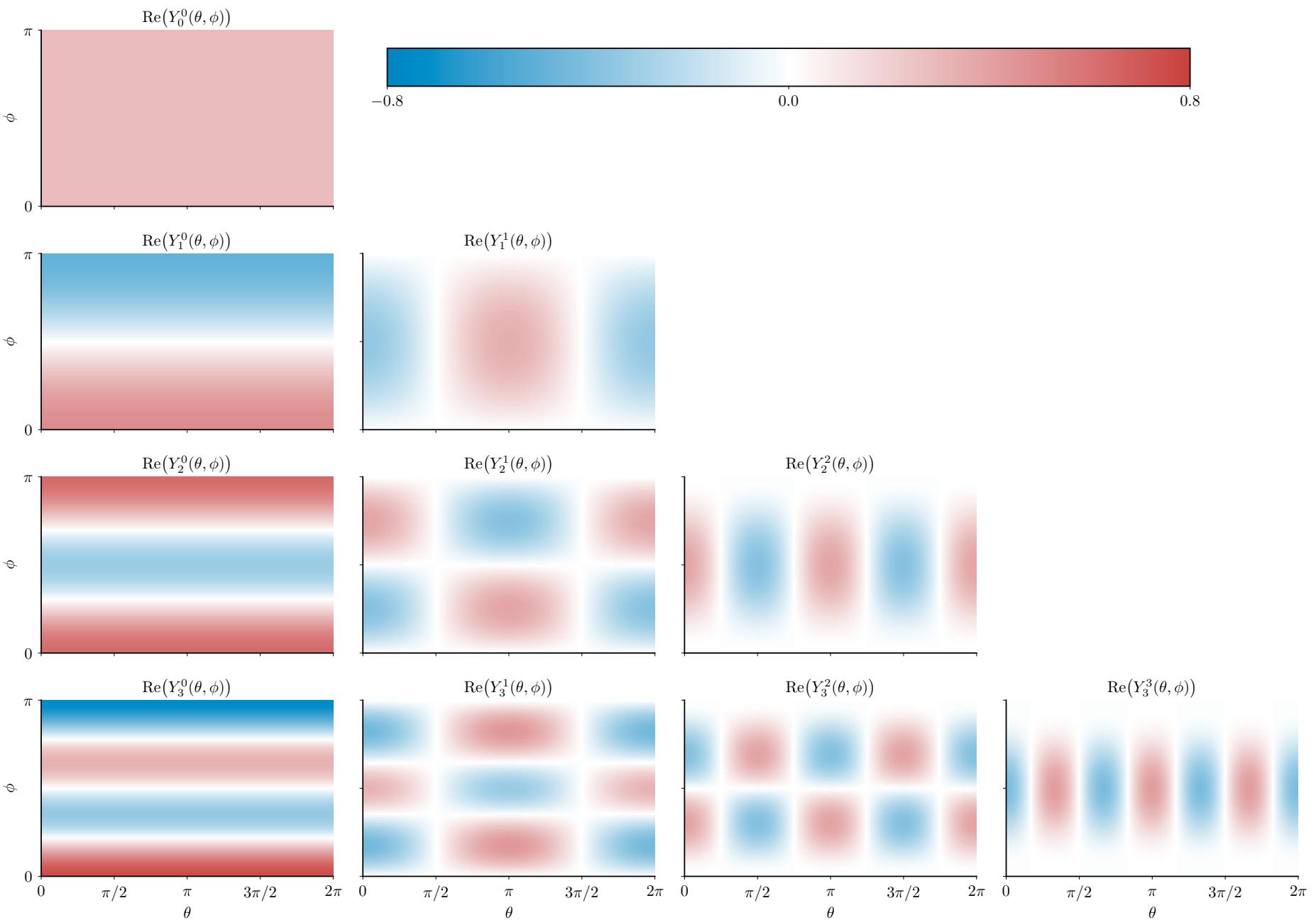


Figure 2.18.: Real parts of the spherical harmonics for different orders.

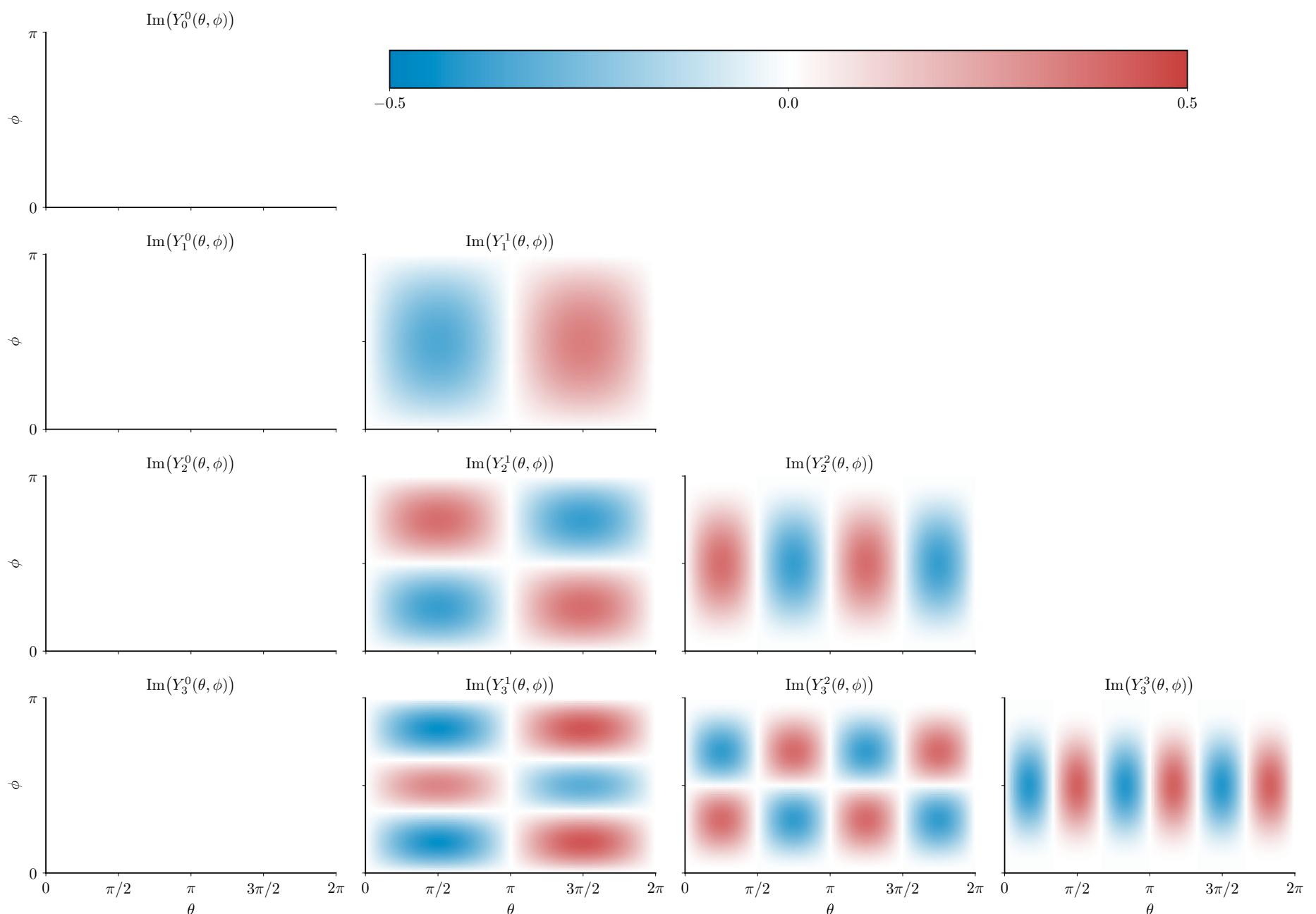


Figure 2.19.: Imaginary parts of the spherical harmonics for different orders.

2.3.1. \mathbf{P}_N method

Other terms of the transport equation can be expressed similarly to the scattering kernel in (2.48), i.e.,

$$\psi(t, \mathbf{x}, \boldsymbol{\Omega}) = \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} \sum_{|m| \leq n} \psi_n^m(t, \mathbf{x}) Y_n^m(\boldsymbol{\Omega}) \quad (2.49)$$

for the angular flux, and

$$q(t, \mathbf{x}, \boldsymbol{\Omega}) = \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} \sum_{|m| \leq n} q_n^m(t, \mathbf{x}) Y_n^m(\boldsymbol{\Omega}) \quad (2.50)$$

for the source term. Time- and space-dependent moments are denoted by $\psi_n^m(t, \mathbf{x})$. The expression for in-scattering, using orthogonality and the addition theorem, reduces to

$$\begin{aligned} & \int_{\mathbb{S}^2} s(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') \psi(t, \mathbf{x}, \boldsymbol{\Omega}') d\boldsymbol{\Omega}' \\ &= \int_{\mathbb{S}^2} \left(\sum_{n=0}^{\infty} s_n \sum_{|m| \leq n} Y_n^m(\boldsymbol{\Omega}) \overline{Y_n^m(\boldsymbol{\Omega}')} \right) \left(\sum_{n=0}^{\infty} \frac{2n+1}{4\pi} \sum_{|m| \leq n} \psi_n^m(t, \mathbf{x}) Y_n^m(\boldsymbol{\Omega}') \right) d\boldsymbol{\Omega}' \\ &= \sum_{n=0}^{\infty} s_n \sum_{|m| \leq n} \sum_{n'=0}^{\infty} \frac{2n'+1}{4\pi} \sum_{|m'| \leq n'} Y_n^m(\boldsymbol{\Omega}) \psi_{n'}^{m'}(t, \mathbf{x}) \int_{\mathbb{S}^2} \overline{Y_n^m(\boldsymbol{\Omega}')} Y_{n'}^{m'}(\boldsymbol{\Omega}') d\boldsymbol{\Omega}' \\ &= \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} \sum_{|m| \leq n} s_n \psi_n^m(t, \mathbf{x}) Y_n^m(\boldsymbol{\Omega}). \end{aligned} \quad (2.51)$$

Unsurprisingly, convolving scattering kernel and angular flux corresponds to multiplication of the respective expansion coefficients in moment space. No approximations have been made up until now. We define $\sigma_{t,n} := \sigma_t - s_n$ and rewrite the transport equation as

$$\sum_{n=0}^{\infty} \sum_{|m| \leq n} \frac{2n+1}{4\pi} Y_n^m(\boldsymbol{\Omega}) [\partial_t \psi_n^m(t, \mathbf{x}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_n^m(t, \mathbf{x}) + \sigma_{t,n} \psi_n^m(t, \mathbf{x}) - q_n^m(t, \mathbf{x})] = 0. \quad (2.52)$$

If we truncate the expansions (2.48), (2.49), and (2.50) at a finite value N , the equality in (2.52) no longer holds true. However, equality is recovered when the residual is projected onto any basis function of the ansatz space $\{\sum_{n=0}^N \sum_{|m| \leq n} c_n^m Y_n^m(\boldsymbol{\Omega}) \mid c_n^m \in \mathbb{R} \text{ for all feasible tuples } (n, m)\}$, i.e.,

$$\partial_t \psi_{n'}^{m'}(t, \mathbf{x}) + \sum_{n=0}^{\infty} \sum_{|m| \leq n} \frac{2n+1}{2n'+1} \left\langle \boldsymbol{\Omega} Y_n^m, Y_{n'}^{m'} \right\rangle_{\mathbb{S}^2} \cdot \nabla_{\mathbf{x}} \psi_n^m(t, \mathbf{x}) + \sigma_{t,n} \psi_{n'}^{m'}(t, \mathbf{x}) - q_{n'}^{m'}(t, \mathbf{x}) = 0, \quad (2.53)$$

for all $0 \leq n' \leq N$ and $|m'| \leq n'$. At first glance, the term $\langle \mathbf{\Omega} Y_n^m, Y_{n'}^{m'} \rangle$ looks troublesome as it implies a potentially dense tensor. However, we can use the relation

$$2\mathbf{\Omega} Y_n^m = \begin{pmatrix} -c_{n-1}^{m-1} & d_{n+1}^{m-1} & 0 & 0 & e_{n-1}^{m+1} & -f_{n+1}^{m+1} \\ ic_{n-1}^{m-1} & -id_{n+1}^{m-1} & 0 & 0 & ie_{n-1}^{m+1} & -if_{n+1}^{m+1} \\ 0 & 0 & 2a_{n-1}^m & 2b_{n+1}^m & 0 & 0 \end{pmatrix} \begin{pmatrix} \overline{Y_{n-1}^{m-1}} \\ \overline{Y_{n+1}^{m-1}} \\ \overline{Y_{n-1}^m} \\ \overline{Y_{n+1}^m} \\ \overline{Y_{n-1}^{m+1}} \\ \overline{Y_{n+1}^{m+1}} \end{pmatrix}, \quad (2.54)$$

which ultimately reduces (2.53) to the complex-valued P_N equations

$$\partial_t \psi(t, \mathbf{x}) + \sum_{i=1}^3 A_i (\nabla_{\mathbf{x}})_i \psi(t, \mathbf{x}) + \Sigma_t \psi(t, \mathbf{x}) - \mathbf{q}(t, \mathbf{x}) = 0, \quad (2.55)$$

where the A_i are sparse matrices, dependent on the coefficients from (2.54). These coefficients and more details on the real-valued version of (2.55) are provided in the appendix of Küpper's thesis [54]. The expansion coefficients of (2.49) are stored in vector format in $\psi(t, \mathbf{x})$.

2.3.2. Advantages and disadvantages of moment methods

Again, we discuss advantages and disadvantages.

Advantages

For smooth solutions, moment methods produce accurate solutions, even for a small number of expansion coefficients. Here, smoothness especially refers to the dependency on the angular variable $\mathbf{\Omega}$, since this is the dependency that we approximate through the moment expansion.

Ray effects do not occur and moment methods yield symmetrical approximate solutions when the true solution is symmetric.

Disadvantages

A major drawback of the P_N method is its tendency to produce oscillatory solutions. In regions of low densities, these oscillations may cause the density to become negative, sometimes resulting in a breakdown of the solution algorithm. However, ways exist to circumvent or at least mitigate these oscillations [69].

Other moment methods, like the minimal entropy (M_N) method, make a different ansatz for the representation of the solution in terms of its moments. At each time step, a constrained optimization problem is solved to ensure that a certain entropy η is minimized [63, 3, 4].

If the solution depends strongly on $\mathbf{\Omega}$, i.e., it is highly anisotropic, more and more moments are necessary to accurately approximate the angular flux. This is true for any Galerkin-type method and relates to the slow decay of expansion coefficients in this case.

3. Ray effects and their mitigation

It was not until the S_N method was applicable in two and three spatial dimensions that the problem of ray effects became fully eminent [59]. To this day, ray effects impede the quest for accurate and reliable, yet efficient transport calculations. The S_N method's raison d'être—restricting the propagation of the angular flux to a finite and therefore computationally manageable set of directions—inevitably destroys rotational invariance and causes oscillatory solution approximations. If undetected, these oscillations can cause the neutron flux to be significantly over- or underestimated; especially troublesome in source-detector problems with only little to no scattering [60]. Increasing the number of directions tends to reduce the oscillations' magnitude while simultaneously increasing their frequency, but, even for a large number of directions, ray effects remain present [58].

More sophisticated strategies to mitigate ray effects make use of biased quadrature sets, which reflect the importance of certain ordinates [1]. Tencer computed the angular flux for several, differently oriented quadrature sets and ray effects are then mitigated by averaging over all solutions [92]. Moreover, a method combining S_N and P_N to reduce ray effects has been introduced by Lathrop [58], with further refinements developed subsequently [51, 84, 71]. The idea is to use a mixture of collocation points as well as basis functions to represent the solution's angular dependency. Consequently, a system for the angular expansion coefficients with an increased coupling of the individual equations needs to be solved. The accuracy of these methods has been studied in a review paper [73] and it turns out that all methods still suffer from ray effects for a line-source inside void.

3.1. Understanding ray effects

Before discussing the two novel ray effect mitigation techniques in Sections 3.2 and 3.3, it is worth discussing the origin of ray effects more thoroughly. Assuming that we can solve both equations exactly, the transport equation

$$\partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_a \psi(t, \mathbf{x}, \boldsymbol{\Omega}) = Q(t, \mathbf{x}, \boldsymbol{\Omega}) \quad (3.1)$$

and its S_N counterpart

$$\partial_t \psi_q(t, \mathbf{x}) + \boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}} \psi_q(t, \mathbf{x}) + \sigma_a \psi_q(t, \mathbf{x}) = Q(t, \mathbf{x}, \boldsymbol{\Omega}_q) \quad (3.2)$$

will return the same values for the angular flux along directions $\boldsymbol{\Omega} = \boldsymbol{\Omega}_q$, i.e., $\psi(t, \mathbf{x}, \boldsymbol{\Omega}_q) = \psi_q(t, \mathbf{x})$. (Note the absence of scattering in the two equations above.)

Consider now the situation depicted in Figure 3.1. A circular, isotropic source (green) is placed at $(0, 0)$ with radius $r = 0.4$. The domain is purely absorbing with $\sigma_a = 1$ and $\sigma_s = 0$ throughout the full domain. The solution will be evaluated along the black line at $x = 1$ for different directions $\boldsymbol{\Omega} = (\cos(\alpha), \sin(\alpha))^T$, denoted by $\psi(y, \alpha)$. Additionally, we pay attention

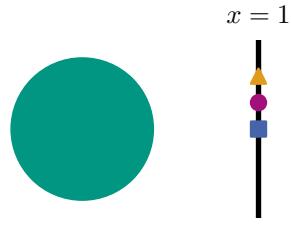


Figure 3.1.: Infinite medium test case with a circular source of radius $r = 0.4$ at $(0, 0)$. The domain is purely absorbing with $\sigma_a = 1$. We evaluate the angular and scalar flux along the black line at $x = 1$ and take a closer look at the three points $(1, 0.3)$, $(1, 0.15)$, and $(1, 0)$ in orange, purple, and blue, respectively. Scene drawn to scale.

to the angular fluxes at the points $(1, 0.3)$, $(1, 0.15)$, and $(1, 0)$. Figures 3.2 and 3.3 show both the angular and scalar flux along the line and the three points, evaluated with $n_q = 11$ and $n_q = 21$ ordinates for the half-space $\alpha \in [-\pi/2, \pi/2]$. Values for the angular flux are exact (up to machine precision). The scalar flux, however, oscillates significantly despite the high number of ordinates and the rather large spatial extend of the isotropic source.

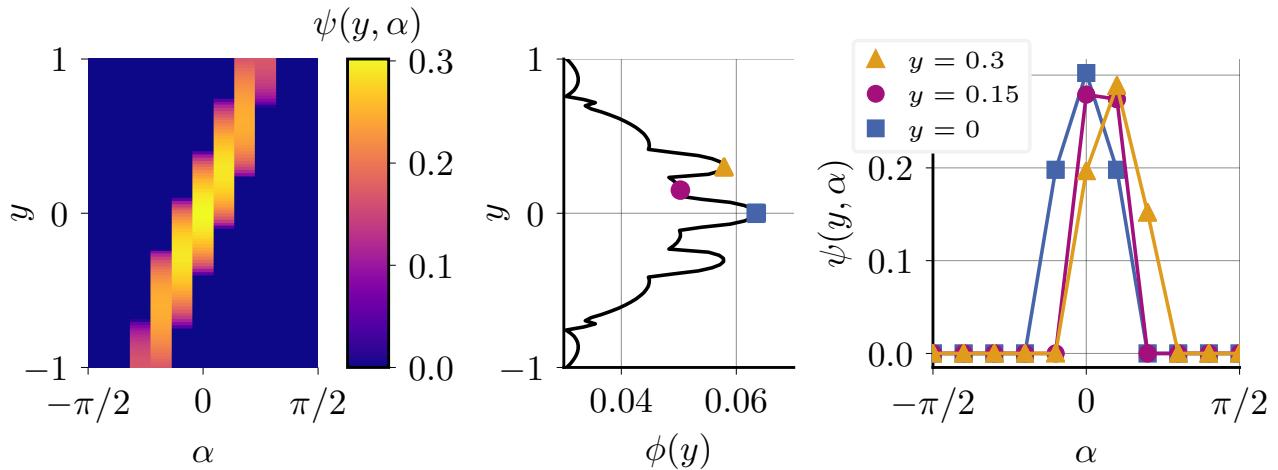


Figure 3.2.: Angular and scalar flux with $n_q = 11$ for the test case presented in Figure 3.1.

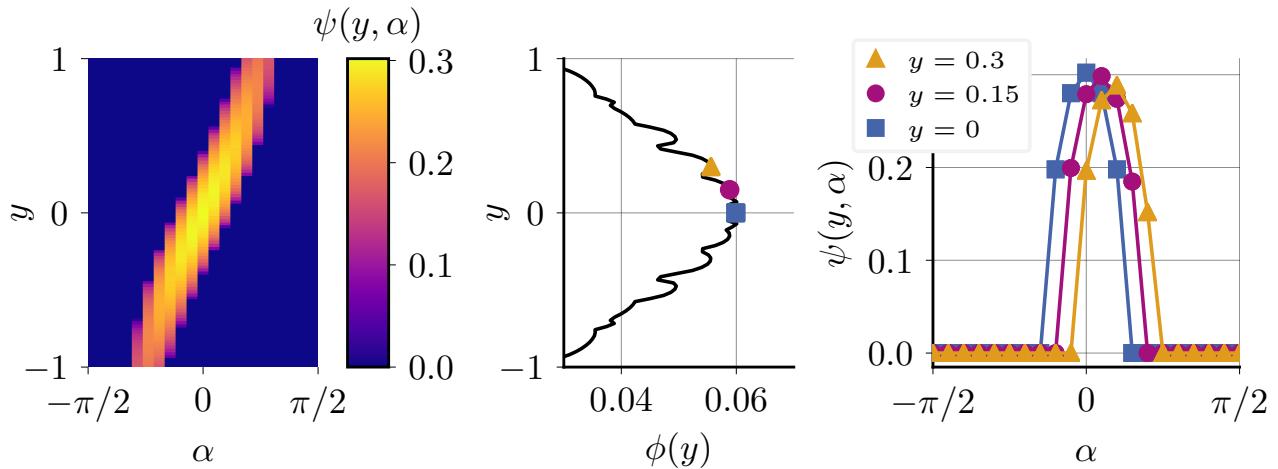


Figure 3.3.: Angular and scalar flux with $n_q = 21$ for the test case presented in Figure 3.1.

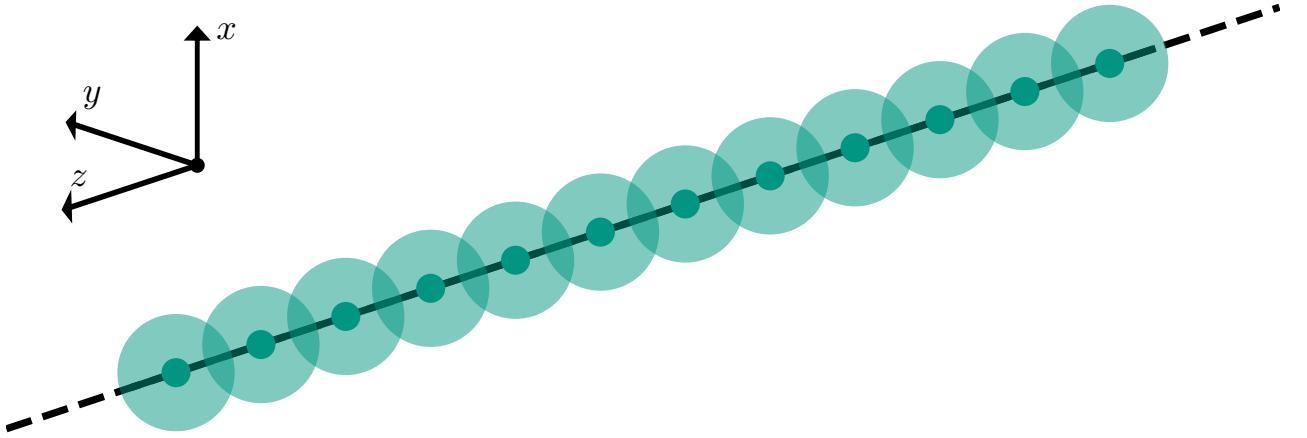


Figure 3.4.: For the line-source test case, particles are released isotropically at $t = 0$ along every point on the z -axis (depicted here for a finite number of points).

The rightmost column in Figures 3.2 and 3.3 then spotlights the origin of ray effects in this experiment: Though isotropic at the source, the angular flux is highly anisotropic at the measurement points. As a consequence, (i) the number of ordinates with non-zero angular flux that contribute to the scalar flux is small, and (ii) this number oscillates as we move along the black line. In Figure 3.2, for example, the number of ordinates with non-zero flux for $y = 0$ and $y = 0.3$ is three, whereas only two ordinates are non-zero for the intermediate point at $y = 0.15$, resulting in significant differences in the scalar flux as shown in the middle column.

A similar observation can be made for the line-source test case [39], which asks for the solution to the three-dimensional, time-dependent transport equation

$$\partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \psi(t, \mathbf{x}, \boldsymbol{\Omega}) = \frac{1}{4\pi} \int_{\mathbb{S}^2} \psi(t, \mathbf{x}, \boldsymbol{\Omega}') d\boldsymbol{\Omega}', \quad (3.3)$$

subject to initial condition

$$\psi(0, \mathbf{x}, \boldsymbol{\Omega}) = \psi_0(\mathbf{x}) := \delta(x, y). \quad (3.4)$$

The setup is depicted in Figure 3.4, with the isotropic initial condition illustrated as balls centered around the z -axis. Due to the nature of the initial condition, the three-dimensional problem yields the same results for every value of z , effectively rendering it a two-dimensional simulation which allows to redefine $\mathbf{x} = (x, y)^T$. An exact (up to evaluation of a double integral) solution for the scalar flux $\phi(t, r) = \phi(t, \|\mathbf{x}\|_2)$ exists and was already visualized in the last section in Figure 2.17a for $t = 1$. Computed beforehand, the scalar flux can replace the integral on the right-hand side of (3.3), yielding

$$\partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \psi(t, \mathbf{x}, \boldsymbol{\Omega}) = Q(t, \mathbf{x}) := \frac{1}{4\pi} \phi(t, \|\mathbf{x}\|_2). \quad (3.5)$$

Similar to the previous experiment, we obtain the exact angular flux via integration, i.e.,

$$\psi(t, \mathbf{x}, \boldsymbol{\Omega}) = e^{-t} \psi_0(\mathbf{x} - t\boldsymbol{\Omega}) + \int_0^t e^{-(t-\tau)} Q(t - \tau, \|\mathbf{x} - \tau\boldsymbol{\Omega}\|_2) d\tau. \quad (3.6)$$

It is infeasible to integrate the Dirac-pulse numerically. Instead, we replace $\psi_0(\mathbf{x})$ with $\psi_0^G(\mathbf{x})$, a narrow Gaussian that is centered at the origin with variance $\sigma^2 = 0.03^2$, similar to the

simulations by McClaren and Hauck [69]. Thus,

$$\psi^G(t, \mathbf{x}, \boldsymbol{\Omega}) = e^{-t} \psi_0^G(\mathbf{x} - t\boldsymbol{\Omega}) + \int_0^t e^{-(t-\tau)} Q(t-\tau, \|\mathbf{x} - \tau\boldsymbol{\Omega}\|_2) d\tau \quad (3.7)$$

approximates $\psi(t, \mathbf{x}, \boldsymbol{\Omega})$, which seems reasonable since the numerical S_N simulations for the line-source problem resolve the initial condition in the same manner. The two terms of the right-hand side of (3.7) are abbreviated as $A(t, \mathbf{x}, \boldsymbol{\Omega})$ and $B(t, \mathbf{x}, \boldsymbol{\Omega})$, denoting the contribution of the initial condition and the source term, respectively. For a fixed set of quadrature points and weights, we compute

$$\phi^A(t, \mathbf{x}) = \sum_{q'=1}^{n_q} w_{q'} A(t, \mathbf{x}, \boldsymbol{\Omega}_{q'}), \quad (3.8a)$$

$$\phi^B(t, \mathbf{x}) = \sum_{q'=1}^{n_q} w_{q'} B(t, \mathbf{x}, \boldsymbol{\Omega}_{q'}), \quad (3.8b)$$

$$\phi^G(t, \mathbf{x}) = \phi^A(t, \mathbf{x}) + \phi^B(t, \mathbf{x}). \quad (3.8c)$$

Computing $\phi^A(t, \mathbf{x})$ is trivial; computing $\phi^B(t, \mathbf{x})$ is not. It involves numerical integration of the source term, which itself requires the numerical evaluation of a double integral. Moreover, this process has to be repeated for $n_x \cdot n_y$ spatial cells and n_q ordinates, restricting the accuracy with which the integral in (3.7) can be evaluated. The component of the scalar flux that is due to the initial condition (ϕ^A) is visualized in Figure 3.5a. For an, admittedly low, accuracy of $\varepsilon_{\text{abs}} = 10^{-2}$, we see the contribution of the source (ϕ^B) in Figure 3.5b. Both images are created using $n_x = n_y = 200$ cells in space (though the four-fold symmetry is exploited) and $n_q = 36$ ordinates. The ordinates result from a tensorized quadrature that discretizes the azimuthal angle equidistantly ($2n_q$ points) and uses Gauss-Legendre roots for the polar direction ($n_q/2$ points). Numerical integrals are computed with the `Cubature.jl`¹ package that implements “one- and multi-dimensional adaptive integration routines for the Julia language, including support for vector-valued integrands and facilitation of parallel evaluation of integrands, based on the Cubature Package by Steven G. Johnson.”

Adding both terms, we obtain the scalar flux $\phi^G(1, \mathbf{x})$ that uses the exact angular flux but performs the spherical integration with 36 ordinates. The result is shown in Figure 3.6a, together with the scalar flux for a full S_6 computation that uses the exact same ordinates in Figure 3.6b. The two results look fairly similar, both in a quantitative and qualitative way. The only two visually noticeable differences are (i) a larger spike in the center of the domain for the S_6 computation and (ii) a concave curvature at the outermost region of the scalar flux for the S_6 computation as opposed to a convex curvature for $\phi^G(1, \mathbf{x})$.

These findings validate the conclusions that we drew in the first experiment: Since the angular flux is exact (up to errors due to the numerical integration) the sole origin of ray effects in the scalar flux is the inexact spherical integration. Ray effect mitigation techniques that reduce oscillations in the scalar flux therefore inevitably have to do so by circumventing the error that is due to the aggregation of the scalar flux from the angular flux. We have already seen that various methods exist that do exactly this. In the following, two new ray effect mitigation techniques—the rS_N and $as-S_N$ method—are presented.

¹Source code available at <https://github.com/JuliaMath/Cubature.jl> and authored by Steven G. Johnson.

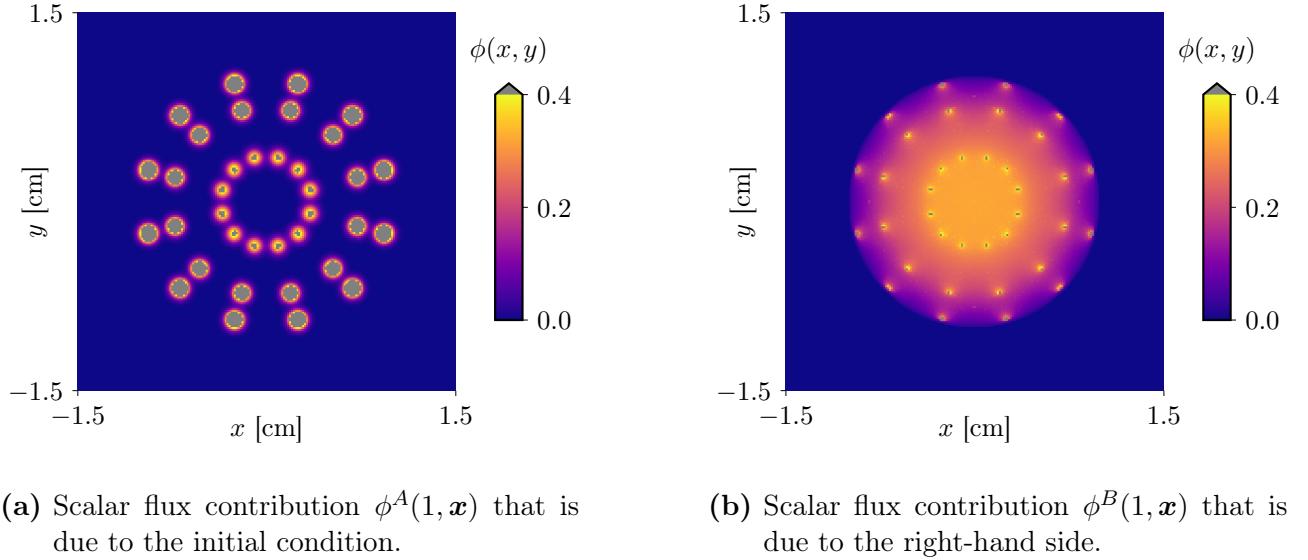


Figure 3.5.: The two contributions that make up the scalar flux $\phi^G(1, \mathbf{x})$ for the line-source test case using the exact (up to numerical integration) angular flux and 36 ordinates.

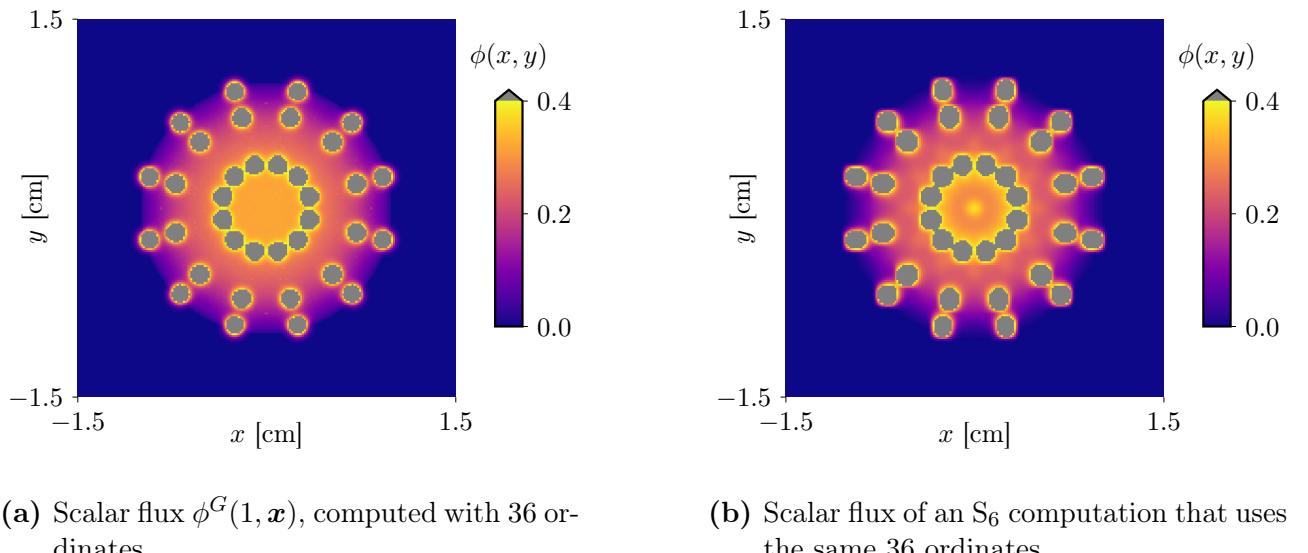


Figure 3.6.: Comparison between the scalar flux $\phi^G(1, \mathbf{x})$, based upon the exact angular flux, and the scalar flux that is the result of an S_6 computation.

3.2. rS_N method

Section 3.2 is, at parts, based upon the results published in Camminady et al. [21]. Findings therein have subsequently contributed to the thesis of Kusch [55], coauthor of Camminady et al. [21].

The rotated S_N (rS_N) method finds a solution for the following dilemma: While we have seen that an increase in the number of ordinates reduces the magnitude of oscillations, it inevitably increases both runtime and memory consumption. To overcome this problem, the rS_N method only *effectively* increases the number of ordinates, while simultaneously adding angular diffusion to smoothen the angular flux. It does so by rotating existing ordinates after each time step

to obtain a new set of directions. An interpolation step then yields the angular flux at these new ordinates. Both components—rotation and interpolation—can be implemented efficiently (increasing the runtime by roughly five to ten percent), while the simulation results are on par with S_N simulations of much larger N . Quintessentially, the rotation-and-interpolation steps add angular diffusion.

Recapitulating the S_N method in Algorithm 3.1, the minimally invasive nature of the rS_N method becomes clear when put alongside in Algorithm 3.2—both written in pseudo-code with differences highlighted in green. Arguably the most complex and time-consuming building block within the S_N method, the computation of the angular flux remains, unaltered. To allow for a faster interpolation of the angular flux, we require specific quadrature points and weights. Additionally, the rotation-and-interpolation step needs to be implemented. We will now discuss these modifications and analyze their effect on the S_N method later on, together with the discussion of numerical experiments.

Algorithm 3.1 The S_N method.

```

1: function  $S_N(\Delta t, t_{\text{end}}, \text{order}, n_x, n_y, \psi_0)$ 
2:    $t \leftarrow 0$ 
3:    $n_q \leftarrow \text{order}^2$ 
4:    $P \leftarrow \text{QPoints}(n_q) \in \mathbb{R}^{3 \times n_q}$ 
5:    $W \leftarrow \text{QWeights}(n_q) \in \mathbb{R}^{n_q}$ 
6:    $\psi \leftarrow \psi_0 \in \mathbb{R}^{n_q \times n_x \times n_y}$ 
7:   while  $t < t_{\text{end}}$  do
8:      $F \leftarrow \text{ComputeFlux}(\psi, P, W)$ 
9:      $\psi \leftarrow \psi + \Delta t \cdot F$ 
10:
11:
12:      $t \leftarrow t + \Delta t$ 
13:   return  $\psi$ 

```

Algorithm 3.2 The rS_N method.

```

1: function  $rS_N(\Delta t, t_{\text{end}}, \text{order}, n_x, n_y, \psi_0, \delta)$ 
2:    $t \leftarrow 0$ 
3:    $n_q \leftarrow 4 \cdot \text{order}^2 - 8 \cdot \text{order} + 6$ 
4:    $P \leftarrow \text{NewQPoints}(n_q) \in \mathbb{R}^{3 \times n_q}$ 
5:    $W \leftarrow \text{NewQWeights}(n_q) \in \mathbb{R}^{n_q}$ 
6:    $\psi \leftarrow \psi_0 \in \mathbb{R}^{n_q \times n_x \times n_y}$ 
7:   while  $t < t_{\text{end}}$  do
8:      $F \leftarrow \text{ComputeFlux}(\psi, P, W)$ 
9:      $\psi \leftarrow \psi + \Delta t \cdot F$ 
10:     $\alpha \leftarrow \delta \cdot \Delta t / n_q$ 
11:     $\psi, P \leftarrow \text{RotateInterpolate}(\psi, P, \alpha)$ 
12:     $t \leftarrow t + \Delta t$ 
13:  return  $\psi$ 

```

The rS_N method uses variants of the octahedron- or icosahedron-based quadrature that has been introduced in Example 2.2.3. Quadrature points result from triangulating faces of an octahedron or icosahedron and then projecting the vertices onto the unit sphere, depicted in Figure 2.8 for one face of an octahedron. The triangulation induces a connectivity that ensures six neighboring triangles and quadrature points (except for the poles, there the number is four) for every quadrature point on the unit sphere.

If we now rotate a quadrature point $\Omega_q \in \mathbb{S}^2$ with the rotation matrix

$$R_\alpha^n = \begin{pmatrix} n_x^2 \gamma + \cos(\alpha) & n_x n_y \gamma - n_z \sin(\alpha) & n_x n_z \gamma + n_y \sin(\alpha) \\ n_y n_x \gamma + n_z \sin(\alpha) & n_y^2 \gamma + \cos(\alpha) & n_y n_z \gamma - n_x \sin(\alpha) \\ n_z n_x \gamma - n_y \sin(\alpha) & n_z n_y \gamma + n_x \sin(\alpha) & n_z^2 \gamma + \cos(\alpha) \end{pmatrix} \quad (3.9)$$

around the axis $\mathbf{n} = (n_x, n_y, n_z)^T \in \mathbb{S}^2$ by the amount α and with $\gamma := (1 - \cos(\alpha))$, we obtain a new quadrature point $\tilde{\Omega}_q \in \mathbb{S}^2$. Regardless of the axis and the rotation magnitude, a rotated quadrature point will fall into a triangle of the unrotated quadrature. Using the spherical version of barycentric interpolation—explained for the planar case in Figure 3.7—we obtain the angular flux at the rotated ordinates. The rotation-and-interpolation step is implemented in Algorithm 3.3.

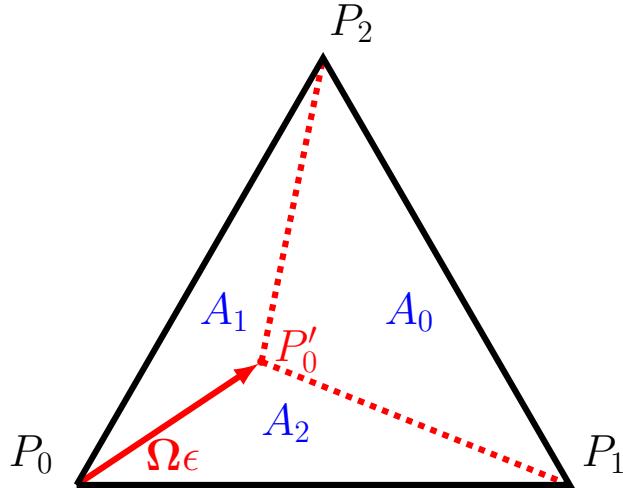
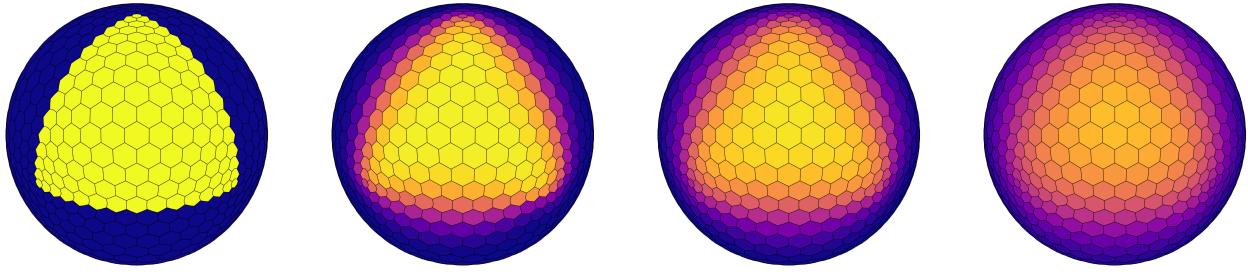


Figure 3.7.: Planar barycentric interpolation works by using the areas A_0, A_1 , and A_2 as interpolation weights to obtain a new function value at P'_0 based upon the function values at P_0, P_1 , and P_2 . In this example, the point P'_0 is the result of shifting P_0 by $\varepsilon \cdot \Omega$. For the spherical case, the corresponding transformation would be the application of the rotation matrix to point P_0 .

We make the following remarks: (i) Since the rotation magnitude is usually small, quadrature points mostly fall into neighboring triangles, allowing to rapidly identify the interpolation nodes for the rotated points. (ii) The procedure does not guarantee preservation of mass, which is only achieved by scaling the angular flux for each spatial cell individually based upon the scalar flux at the previous time step. (iii) We perform the same rotation in every spatial cell. Thus, the interpolation nodes and weights can be precomputed before the actual interpolation is executed in every spatial cell individually—potentially in parallel. (iv) Quadrature points keep their quadrature weight. (v) Lastly, rotating by $\alpha = \delta \cdot \Delta t / n_q$ ensures that the user only needs to choose the dimensionless quantity δ .

Before discussing the full transport test cases, we can see the diffusive effect that the rotation-and-interpolation step has in a simple experiment. For forth-and-back rotations around a random axis, and switching between δ and $-\delta$ in consecutive time steps, we have shown in Camminady et al. [21] that we add a discrete second-order derivative in the azimuthal angle to the right-hand side of the transport equation. Rotating by $\alpha = \delta \cdot \Delta t / n_q$ ensures a reasonable limit: The added diffusion vanishes when the number of ordinates increases, and increasing the number of spatial cells or time steps does not increase the amount of diffusion added to the equation. The analysis becomes more complicated for the three-dimensional case. This is due to several factors: (i) Not all triangles on the unit sphere have the same size. (ii) Though quadrature points are being rotated (mostly) into neighboring triangles, finding out the exact triangle is difficult analytically. (iii) Most importantly, we change the axis around which we rotate after each time step randomly. Thus the process has a random nature and needs to be interpreted from a statistical viewpoint as well. We have, however, observed that due to the large number of time steps, simulation results look almost indistinguishable when being run repeatedly. Figure 3.8 demonstrates the smearing out effect of an indicator function, applied to the first octant of the quadrature set. Rotating forth and back repeatedly causes the imprint of the indicator function to diffuse over the sphere.



(a) Before rotating. (b) After 5 rotations. (c) After 10 rotations. (d) After 20 rotations.

Figure 3.8.: Each quadrature point is assigned a value of $\psi = 1.0$ if it is positioned in the first octant and $\psi = 0.0$ otherwise. The associated area is colored accordingly. The rotation and interpolation procedure around a random axis is then applied 5, 10, and 20 times, respectively. In each time step, we rotate forth and back around the same axis by the same magnitude to keep the orientation of the quadrature points. The diffusive behavior is clearly visible.

Algorithm 3.3 The rotation-and-interpolation step.

```

1: function RotateInterpolate( $\psi, P, \alpha$ )
2:    $n_q, n_x, n_y = \text{size}(\psi)$ 
3:    $n = \text{getRandomAxis}() \in \mathbb{S}^2$                                The axis we are going to rotate around.
4:    $R = \text{getRotationMatrix}(n, \alpha) \in \mathbb{R}^3$                   See (3.9).
5:    $\hat{P} = R \cdot P \in \mathbb{R}^{3 \times n_q}$                          The rotated quadrature points.
6:    $W = \text{zeros}(n_q, n_q) \in \mathbb{R}^{n_q \times n_q}$                   The matrix  $W$  will store interpolation weights.
7:    $\hat{\psi} = \text{zeros}(n_q, n_x, n_y) \in \mathbb{R}^{n_q \times n_x \times n_y}$   The tensor  $\hat{\psi}$  will store the interpolated flux.
8:
9:   for  $q = 1, \dots, n_q$  do
10:     $\hat{p} = \hat{P}[:, q]$                                          Store in  $\hat{p}$  a quadrature point from the rotated quadrature.
11:     $i, j, k = \text{interpolateFrom}(P, \hat{p})$                   Compute the three vertices of the triangle that
12:     $w_i, w_j, w_k = \text{interpolationWeights}(P, \hat{p})$           the rotated quadrature point  $\hat{p}$  falls into.
13:     $W[q, i] = w_i, W[q, j] = w_j, W[q, k] = w_k$            The interpolation weights.
14:   return  $\hat{\psi}, \hat{P}$                                          Store the interpolation weights in  $W$ .
15:
16:   for  $i = 1, \dots, n_x$  do
17:     for  $j = 1, \dots, n_y$  do
18:        $\hat{\psi}[:, i, j] = W \cdot \psi[:, i, j]$                       Apply interpolation in each spatial cell.
19:        $\hat{\psi}[:, i, j] = \hat{\psi}[:, i, j] / \text{sum}(\hat{\psi}[:, i, j]) \cdot \text{sum}(\psi[:, i, j])$   Preserve mass.

```

3.2.1. rS_N and the line-source test case

Since the line-source² test case yields the same angular and scalar flux for every value z , the problem can be evaluated in the x - y -plane, more precisely the domain $[a, b] \times [a, b] = [-1.5 \text{ cm}, 1.5 \text{ cm}] \times [-1.5 \text{ cm}, 1.5 \text{ cm}]$. The domain is purely scattering with $\sigma_a = 0 \text{ cm}^{-1}$ and $\sigma_t = \sigma_s = 1 \text{ cm}^{-1}$, and we evaluate the solution at a final time $t = 1\text{s}$. For the initial condition

$$\psi(0, \mathbf{x}, \boldsymbol{\Omega}) = \frac{1}{4\pi} \delta(\mathbf{x}), \quad (3.10)$$

there exists a semi-analytical solution [38]. Numerically, we resolve the initial condition as a narrow Gaussian and offset everything from zero, i.e.,

$$\psi(0, \mathbf{x}, \boldsymbol{\Omega}) = \max \left\{ 0.001, \frac{1}{4\pi\sigma^2} \exp \left(-\frac{\|\mathbf{x}\|_2^2}{4\sigma^2} \right) \right\}, \quad (3.11)$$

with $\sigma^2 = 0.03^2$. We evaluate the scalar flux rather than the angular flux. Computed exactly, we should see a radially symmetric solution with a wavefront propagating away from the domain's origin. Solutions are identical along cuts in the domain that pass through the origin. Moreover, the solution is constant along circles around the domain's origin.

Figures 3.9, 3.10, and 3.11 showcase the results. The quadrature set is the octahedron (lerp) quadrature. Orders are taken as 4, 6, 8, or 10; resulting in 38, 102, 198, or 326 quadrature points, respectively. The rotation magnitude δ varies between 0, 4, 8, and 10. Consequently, $r_\delta S_N$ denotes a solution that was computed with the discrete ordinates method of order N and rotation magnitude δ . We used a second-order finite volume scheme with minmod-limiter to evolve the angular flux in time. Throughout all simulations, we use a grid of 100×100 spatial cells and a CFL number of 0.95. The scalar flux is visualized at the final time $t = 1\text{s}$.

Figure 3.9 shows—in the *eyeball norm*—that ray effects can be mitigated in two ways; either by an increase of quadrature points, or by an increase of the rotation magnitude. While an increase in the number of quadrature points causes a proportional increase in both memory and runtime, adding rotation increases the runtime by only 10%, regardless of the magnitude. We can also identify pairs with similar ray effects. For example, the $r_0 S_{10}$ method and the $r_4 S_8$ method yield similar results even though the number of quadrature points varies significantly. We can be more precise with the help of Figures 3.10 and 3.11. Here we see the solution along cuts through the domain and along concentric circles around the domain's origin. Not only does the magnitude of oscillations along cuts decrease when we add rotation, also the solutions for horizontal and vertical cuts tend to converge to the solution of diagonal cuts. Present in all simulations with rotation is the solution's diffusive character: The reference solution has a sharp decrease in the scalar flux at roughly $x = 1$, but the $r_\delta S_N$ simulations tend to smear out that edge. Looking at the $r_{10} S_4$ computation, the solution's propagation speed has arguably decreased when compared with the $r_0 S_4$ result. This behavior is less dominant for larger numbers of quadrature points. Looking at Figure 3.11, we see the scalar flux along concentric circles of radii 0.2, 0.6, 0.9, and 1.0. Since the reference solution is radially symmetric, we expect a constant value along these circles but see stark oscillations instead. Except for the $r = 1.0$ case, adding rotation shifts the scalar flux closer to the reference solution's scalar flux. For the $r = 1.0$ case, however, the solution is underestimated due to the aforementioned smearing out.

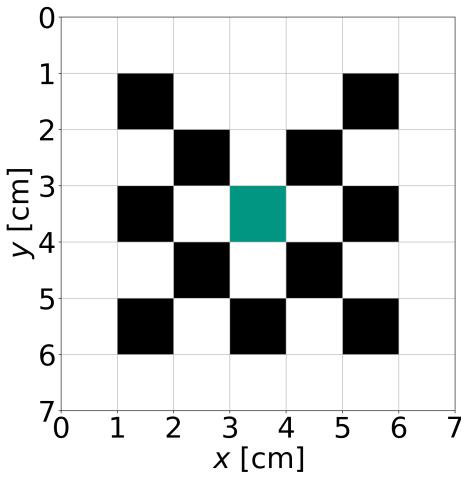
²The name *line-source test case* is arguably suboptimal since we are dealing with an initial condition rather than a source. We do, however, keep the name for historical reasons.

Acknowledging that the r_8S_8 and r_4S_{10} solutions are on par, the added rotation allows for similar solutions with two thirds the number of quadrature points. This directly translates to a reduction of both memory and runtime by about 30%.

To demonstrate that the added rotation is useful in problems that are not radially symmetric—the line-source test case is perhaps the most apt test case to investigate—we consider the lattice test case next.

3.2.2. rS_N and the lattice test case

Purpose of the lattice test case [16, 17] is to model—in a very simplified and idealized way—radiation in a reactor. Depicted in Figure 3.12, the heterogeneous domain consists of a source in the center (green), regions of pure scattering (white), and purely absorbing blocks (black). Cross sections and source strength are given in Table 3.1. The angular flux is initially zero as the source is the sole driver of the transport dynamics. Two versions of this test case exist; time-dependent and steady-state. We consider the time-dependent case and evaluate the solution after 3.2 seconds, chosen because particles are bound to reach the domain’s (vacuum) boundary by that time. Because the scalar flux spans several orders of magnitude, we consider the \log_{10} scalar flux, visualized in Figure 3.13, together with three isolines for $\log_{10}(\phi) = -3$, $\log_{10}(\phi) = -4$, and $\log_{10}(\phi) = -5$. The rotation magnitude δ again varies in the different columns as does the number of quadrature points in the different rows. Numerical parameters match those of the line-source test case, only the spatial resolution is refined as the domain $[0 \text{ cm}, 7 \text{ cm}]^2$ is discretized with $n_x = n_y = 280$ cells.



Color	σ_a in cm^{-1}	σ_s in cm^{-1}	Q in $\text{cm}^{-2}\text{s}^{-1}$
white	0	1	0
black	10	0	0
green	10	0	1

Table 3.1.: Cross sections and source for the lattice test case.

Figure 3.12.: Layout of the lattice test case.

Similar conclusions as in the line-source test case can be drawn for the lattice test case. Ray effects, most visible along the isolines, are mitigated by an increased number of ordinates or by added rotation. Ray effects are present, even for 326 quadrature points. Adding rotation again yields results that are on par with those achieved by a significant increase in quadrature points. Noticeable in the combination of high rotation and small number of ordinates is, again, the reduced propagation speed (e.g., $r_{10}S_4$).

Together with Section 3.2.1, these two test cases show the benefits and versatility of the rS_N method. Adding little computational overhead, the rS_N method allows for a $1.5 \times$ increase

in performance and memory efficiency. This holds true both for test cases that are radially symmetric and those that are not. While the implementation of the rS_N method is straightforward if an existing S_N code is available, there is some caveat. Our implementation solves the transport equation explicitly via means of a finite-volume scheme. It is, however, common to use source iteration instead; especially in high-performant, many-cores implementations. Since source iteration requires to march through the spatial cells in a specific order, and given that this order depends on the ordinates, rotating quadrature points inevitably requires to recompute this order after each rotation. This might be infeasible if the ordering is precomputed and serves as a basis to load-balance the computational work by distribution spatial cells to different cores.

Another question addresses the choice of the rotation magnitude: How to choose this parameter if no parameter study can be performed due to the unavailability of a reference solution? Two approaches might circumvent this problem. (i) We have not discussed the influence that changing the spatial resolution has on the solution's outcome. Since the amount by which we execute the rotation scales with the spatial resolution, the effect of added rotation is mostly independent of the number of spatial cells. A parameter study can therefore be performed on a coarse spatial grid for the same problem and the optimal δ will be used for the fine grid afterwards. (ii) We did observe that an optimal δ in the line-source test case was also a reasonably good δ for the lattice test case. Thus, if possible, a simplified version of the problem of interest can be used to perform a parameter study beforehand. The success of this approach arguably depends on the similarity of the two problems.

Lastly, the influence of quadrature sets has been omitted from the discussion so far. Qualitatively, the results are similar, regardless of whether we choose the octahedron (lerp or slerp) or icosahedron (lerp or slerp) quadrature. It is worth mentioning that the icosahedron quadrature does not possess a 90° rotational symmetry, although the quadrature points are spaced more evenly. We decided to use the octahedron (lerp) quadrature to exploit this symmetry, as it is present in both test cases. It is important to note that we are not restricted to only these four quadratures: Quadrature points from any quadrature set can be rotated around random axes. The only fundamental requirement is an available interpolation routine. Here, the octahedron and icosahedron quadratures benefit from the construction via triangulation in combination with the barycentric interpolation. However, other ways to execute the interpolation steps are possible. If the rS_N is to preserve positivity of the angular flux, the interpolation routine has to preserve positivity, too.

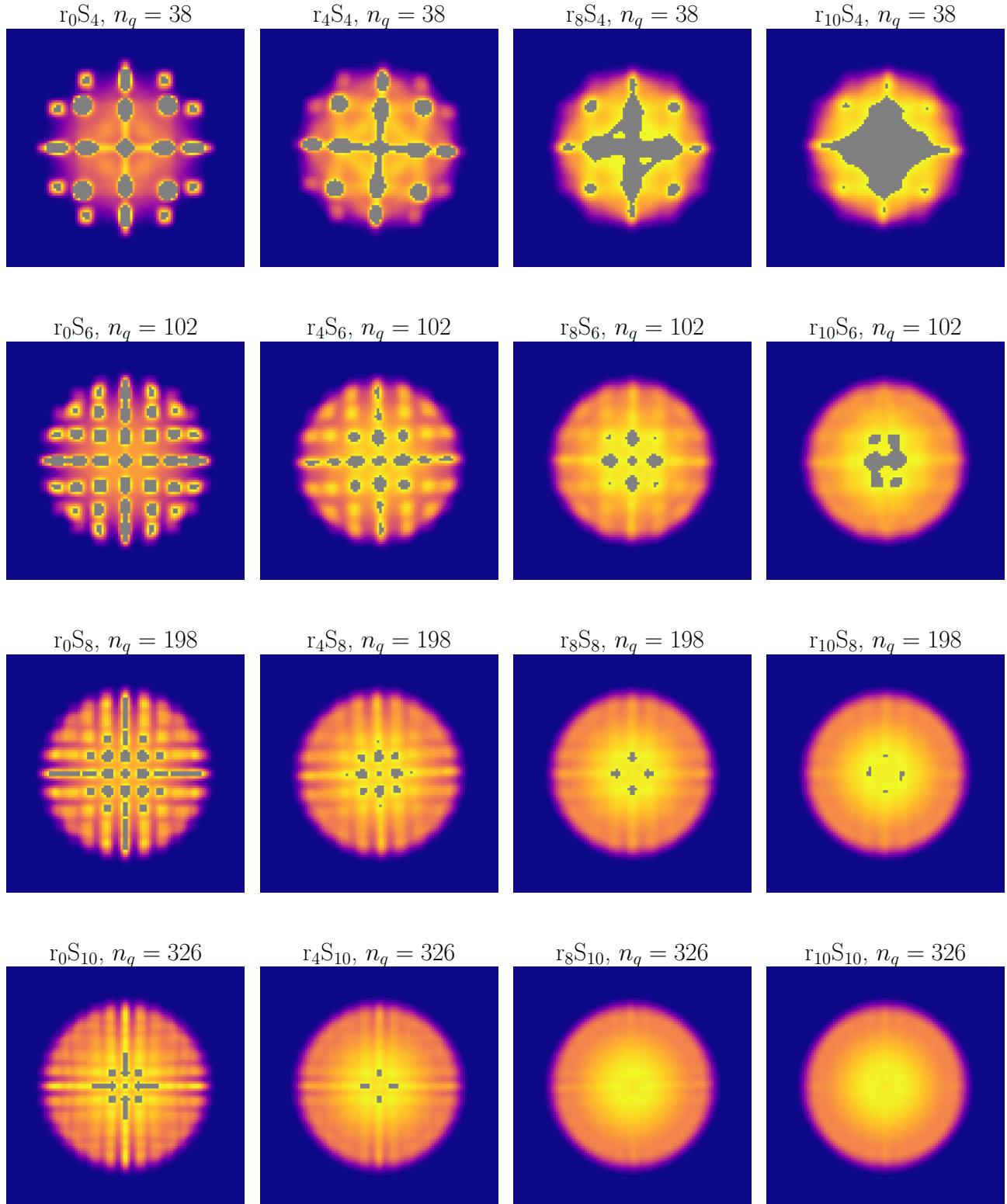


Figure 3.9.: Scalar flux for the line-source test case at $t = 1$ second using the octahedron (lerp) quadrature. The colorbar is omitted but identical to that in Figure 2.17a. The spatial domain is $[-1.5 \text{ cm}, 1.5 \text{ cm}]^2$ with $n_x = n_y = 100$. Rows vary in n_q and columns vary in δ , respectively.

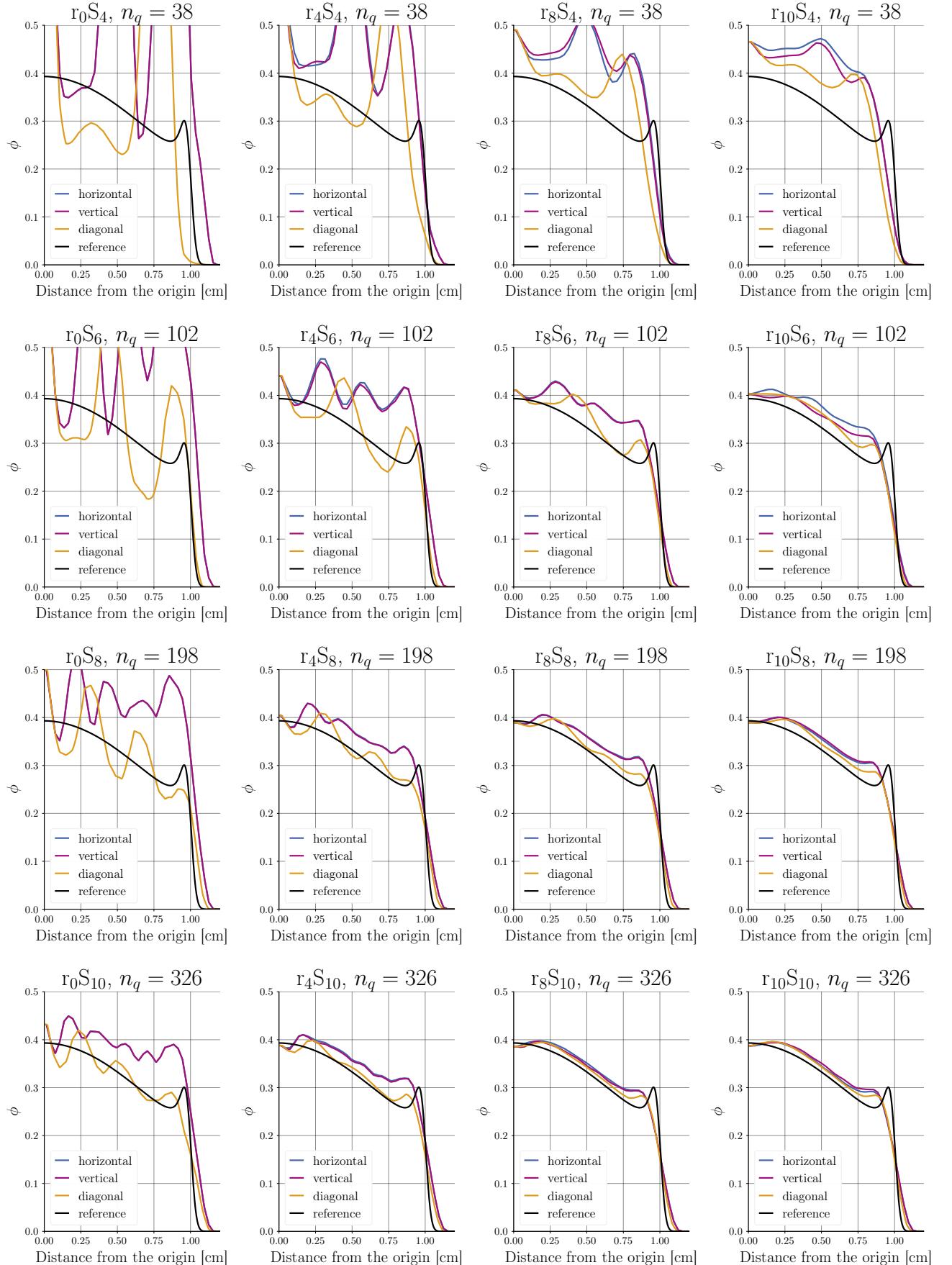


Figure 3.10.: Scalar flux for the line-source test case at $t = 1$ second using the octahedron (lerp) quadrature. The solution is visualized along three cuts: Horizontally (blue), vertically (purple), and diagonally (orange). The reference solution is given in black. The spatial domain is $[-1.5 \text{ cm}, 1.5 \text{ cm}]^2$ with $n_x = n_y = 100$. Rows vary in n_q and columns vary in δ , respectively.

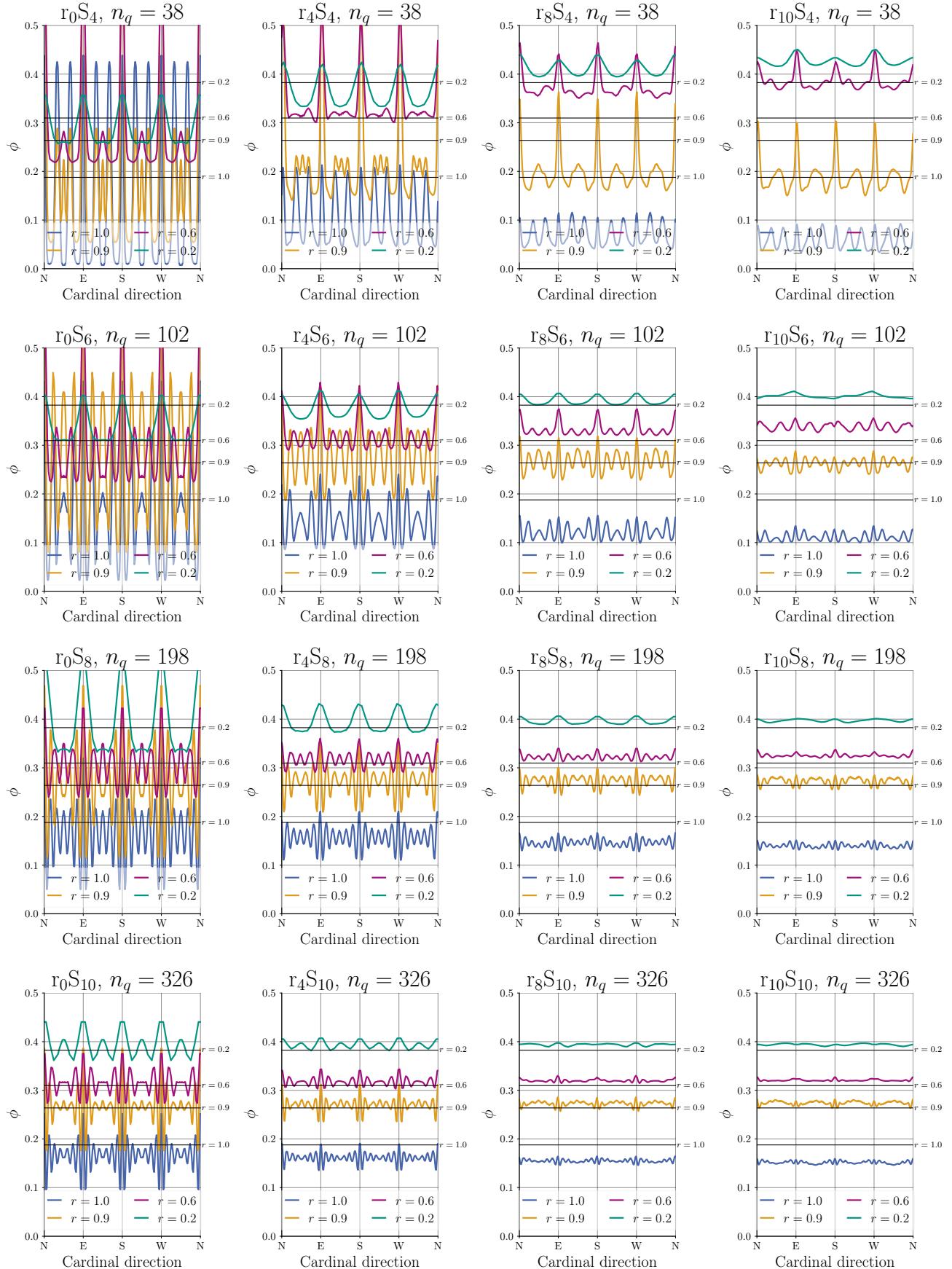


Figure 3.11.: Scalar flux for the line-source test case at $t = 1$ second using the octahedron (lerp) quadrature. The solution is visualized along four concentric circles: $r = 1.0$ (blue), $r = 0.9$ (orange), $r = 0.6$ (purple), and $r = 0.2$ (green). The reference solutions are given in black for all radii. The spatial domain is $[-1.5 \text{ cm}, 1.5 \text{ cm}]^2$ with $n_x = n_y = 100$. Rows vary in n_q and columns vary in δ , respectively.

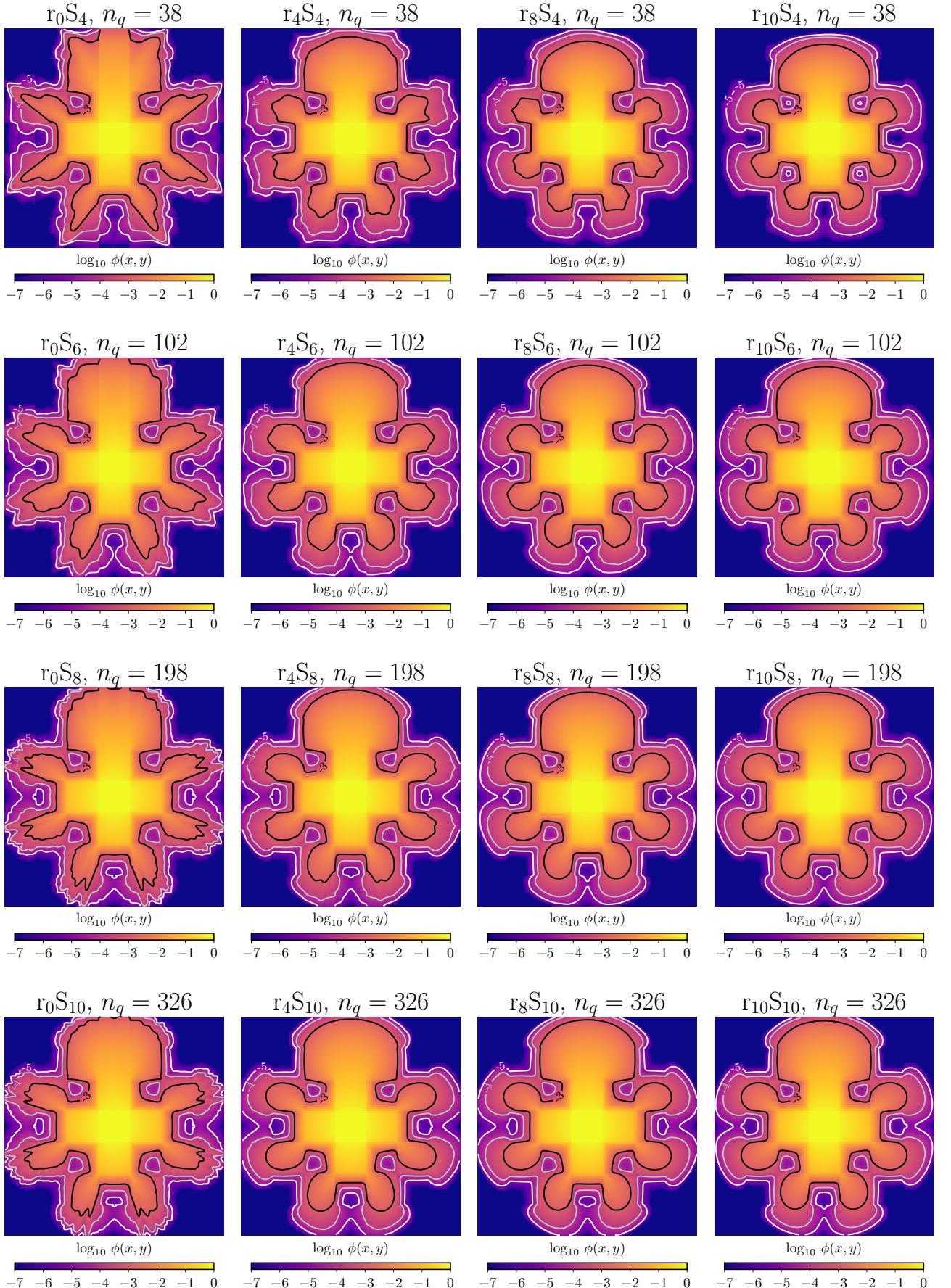
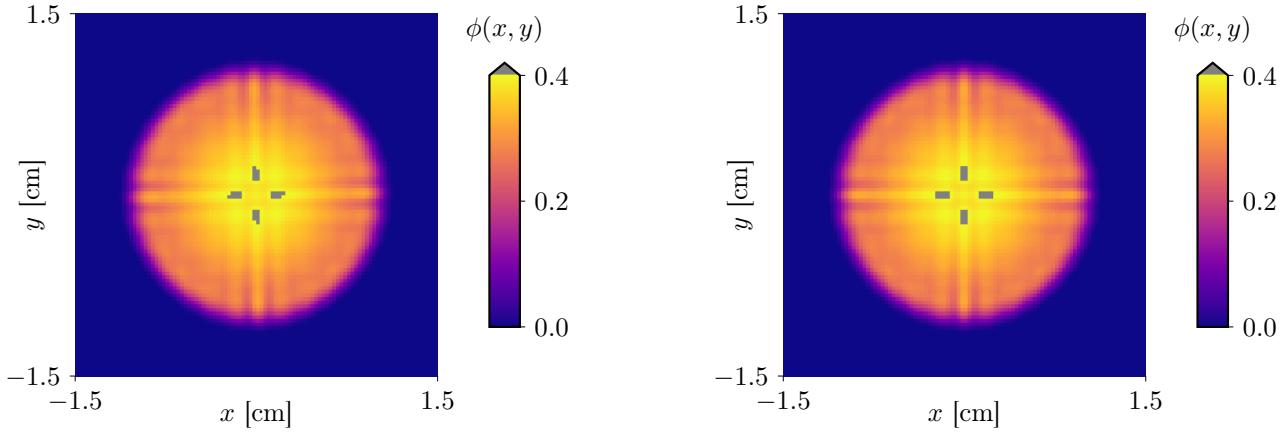


Figure 3.13.: Scalar flux (\log_{10}) for the lattice test case at $t = 3.2$ seconds using the octahedron (lerp) quadrature. The spatial domain is $[0 \text{ cm}, 7 \text{ cm}]^2$ with $n_x = n_y = 280$. Rows vary in n_q and columns vary in δ , respectively. Isolines are drawn to highlight the values -5 (white), -4 (gray), and -3 (black).

3.2.3. rS_N without change of ordinates



(a) Scalar flux for the standard rS_N method with $\delta = 4$.

(b) Scalar flux for the rS_N method with two opposite rotations of $\delta = 2$.

Figure 3.14.: Comparison of two ways to implement the rS_N method.

The change of ordinates that results from the rotation step can be problematic in optimized codes, but also for time-independent problems. Changing the ordinates during every transport sweep potentially interferes with the method's convergence. Additionally, to evaluate the difference of the solution before and after a transport sweep, an additional interpolation step is necessary.

To overcome this problem, we alter the rS_N method in the following way: Instead of performing one rotation and one interpolation between two transport sweeps (or at every time step), we perform *two* rotation and interpolation steps. We rotate the quadrature points around a random axis by $\delta/2$; interpolate function values at the new quadrature points; rotate back by $-\delta/2$ around the same axis to obtain the exact same quadrature points as before; and we then interpolate again, using the quadrature points and function values from the intermediate step. As a result, transport sweeps (or time stepping) always happen with the exact same set of quadrature points and weights. Nevertheless, we still add the same amount of angular diffusion, at least qualitatively. This is depicted in Figures 3.14a and 3.14b for the scalar flux of the line-source problem with the standard rS_N method and the forth-and-back modification, respectively.

3.3. $as-S_N$ method

Section 3.3 is, at parts, based upon the results in Frank et al. [37]. Findings therein have subsequently contributed to the thesis of Kusch [55], coauthor of Frank et al. [37].

Section 3.1 demonstrates that ray effects are essentially due to the aggregation of the scalar flux, i.e., the inability to compute the spherical integral reasonably well. Quintessentially, the rS_N method adds angular diffusion, facilitating to integrate the angular flux numerically. Recomputing the sweeping order anew after each time step might ultimately be an unbearable burden when it comes to the integration of the rS_N method into existing S_N codes; rendering

the r S_N difficult—if not impossible—to sell. Although this problem can seemingly be avoided when rotating forth and back, this modification of r S_N requires further research.

The artificial scattering S_N (as- S_N) method overcomes this hurdle while simultaneously mitigating ray effects equally well. At its core, the as- S_N method adds artificial scattering to the right-hand side of the transport equation in such a way that the following requirements are satisfied: (i) In the limit of infinite ordinates, the added term should vanish. (ii) Artificial scattering should be independent of the number of spatial cells and time steps. (iii) Despite adding a nonphysical term to the right-hand side, the solution quality should increase due to the mitigated ray effects. (iv) The ordinates remain unchanged throughout the simulation. Lastly, (v) the effect of artificial scattering should be investigatable both numerically and theoretically.

Starting with the linear transport equation

$$\begin{aligned} \partial_t \psi(t, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \sigma_t \psi(t, \mathbf{x}, \Omega) \\ = \sigma_s \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(t, \mathbf{x}, \Omega') d\Omega' + q(t, \mathbf{x}, \Omega), \end{aligned} \quad (3.12)$$

we add artificial scattering to the right-hand side to obtain

$$\begin{aligned} \partial_t \psi(t, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \Omega) + \sigma_t \psi(t, \mathbf{x}, \Omega) \\ = \sigma_s \int_{\mathbb{S}^2} s(\Omega \cdot \Omega') \psi(t, \mathbf{x}, \Omega') d\Omega' + q(t, \mathbf{x}, \Omega) \\ + \sigma_{as} \int_{\mathbb{S}^2} s_{\varepsilon}(\Omega \cdot \Omega') (\psi(t, \mathbf{x}, \Omega') - \psi(t, \mathbf{x}, \Omega)) d\Omega'. \end{aligned} \quad (3.13)$$

Choosing $s_{\varepsilon} : [-1, 1] \rightarrow \mathbb{R}$ as

$$s_{\varepsilon}(\mu) = \frac{2}{\sqrt{\pi} \varepsilon \text{Erf}(\frac{2}{\varepsilon})} \exp\left(-\frac{(1-\mu)^2}{\varepsilon^2}\right) \quad (3.14)$$

complies with the five requirements that we imposed. The error function satisfies $\text{Erf}(x) \rightarrow 1$ as $x \rightarrow \infty$. In general, s_{ε} can be any Dirac-like sequence, i.e.,

$$\int_{-1}^1 s_{\varepsilon}(\mu) d\mu = 1 \quad \text{and} \quad \int_{-1}^1 s_{\varepsilon}(\mu) f(\mu) d\mu \rightarrow f(0) \quad (3.15)$$

for any sufficiently smooth function f and $\varepsilon \rightarrow 0$. We paraphrase some of the remarks that were initially made in Frank et al. [37]:

- In- and out-scattering cancel if $\varepsilon \rightarrow 0$ and artificial scattering therefore vanishes. In compliance with (i), we therefore set $\varepsilon = \beta/n_q$ and let the user choose β .
- Particles are neither gained nor lost because the scattering kernel s_{ε} is mass preserving.
- Similar to the P_{N-1} -equivalent S_N method [58, 86], we add a fictitious term to the standard transport equation. However, the kernel in the as- S_N method is highly forward-peaked.
- The as- S_N method can be solved efficiently, both via implicit or explicit schemes.

More information on the implicit implementation of the as- S_N method can be found in the original paper [37] and the thesis by Kusch [55]. Worth mentioning, however, is the fact that the physical scattering term and the artificial scattering term are treated differently. When physical scattering is mostly isotropic, an expansion in spherical harmonics is used. Contrary to that, artificial scattering is highly forward-peaked and thus well-suited for sweeping methods. If this distinction is omitted, implementing the as- S_N method into existing S_N codes is simply realized by changing the scattering kernel.

3.3.1. Angular discretization

Our goal is to include artificial scattering in the S_N equations in (2.13). By simply approximating the artificial scattering term in (3.13) with the chosen quadrature rule, we obtain the as- S_N equations

$$\begin{aligned} \partial_t \psi_q(t, \mathbf{x}) + \boldsymbol{\Omega}_q \cdot \nabla_{\mathbf{x}} \psi_q(t, \mathbf{x}) + \sigma_t(\mathbf{x}) \psi_q(t, \mathbf{x}) + \sigma_{\text{as}}(\mathbf{x}) \psi_q(t, \mathbf{x}) \\ = \sigma_s(\mathbf{x}) \sum_{p=1}^{n_q} w_p \cdot c_q \cdot s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_p) \psi_p(t, \mathbf{x}) \\ + \sigma_{\text{as}}(\mathbf{x}) \sum_{p=1}^{n_q} w_p \cdot c_q^{(\varepsilon)} \cdot s_{\varepsilon}(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_p) \psi_p(t, \mathbf{x}) \\ + q_q(t, \mathbf{x}). \end{aligned} \quad (3.16)$$

Here, $c_q := 1 / \sum_p w_p \cdot s(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_p)$ and $c_q^{(\varepsilon)} := 1 / \sum_p w_p \cdot s_{\varepsilon}(\boldsymbol{\Omega}_q \cdot \boldsymbol{\Omega}_p)$ are normalization factors. While, on the continuous level, these factors are the same for every direction, we obtain a dependency on the chosen ordinates due to the non-uniform discretization in angle. These normalization factors are needed to obtain a simple expression for the out-scattering terms. Moving these terms to the left-hand side of (3.16) stabilizes the source iteration that is used in the implicit method.

It remains to pick an adequate quadrature set. When applying artificial scattering, the solution smears out along the directions in the quadrature set. To ensure an evenly spread artificial scattering effect, a quadrature with a highly uniform ordinate distribution should be chosen. Here, we decide to use the icosahedron (slerp) quadrature due to the nearly uniform distribution of quadrature points. Other quadratures were explored too, but the best results were achieved with the icosahedron quadrature, used in all following computations.

3.3.2. Modified equation and asymptotic analysis

According to Pomraning [81], the Fokker-Planck operator can be a legitimate description of highly peaked scattering. This is true if (i) the scattering kernel $s_{\varepsilon}(\mu)$ is a Dirac sequence, and (ii) the transport coefficients $p_{\varepsilon,i} := \int_{-1}^{+1} (1 - \mu)^i s_{\varepsilon}(\mu) d\mu$ are of order $\mathcal{O}(\varepsilon^i)$. The resulting modified equation then reads

$$\begin{aligned} \partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + (\sigma_a + \sigma_s) \psi(t, \mathbf{x}, \boldsymbol{\Omega}) \\ = \sigma_s \cdot \phi(t, \mathbf{x}) + \pi \cdot p_{\varepsilon,1} \cdot \sigma_{\text{as}} \cdot \Delta_{\boldsymbol{\Omega}} \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \mathcal{O}(\varepsilon^2), \end{aligned} \quad (3.17)$$

where $\Delta_{\boldsymbol{\Omega}}$ is the Laplace operator in spherical coordinates. We have already shown (i). To verify (ii), let $y = (1 - \mu)/\varepsilon$. Then

$$p_{\varepsilon,i} = \int_{2/\varepsilon}^0 (\varepsilon y)^i \frac{2}{\sqrt{\pi} \varepsilon \text{Erf}(\frac{2}{\varepsilon})} e^{-y^2} (-\varepsilon) dy \quad (3.18a)$$

$$= \frac{2}{\sqrt{\pi} \varepsilon \text{Erf}(\frac{2}{\varepsilon})} \varepsilon^i \int_0^{2/\varepsilon} y^i e^{-y^2} dy \quad (3.18b)$$

$$= \frac{2}{\sqrt{\pi} \varepsilon \text{Erf}(\frac{2}{\varepsilon})} \varepsilon^i \left[\Gamma\left(\frac{1+i}{2}\right) - \Gamma\left(\frac{1+i}{2}, \frac{4}{\varepsilon^2}\right) \right] \quad (3.18c)$$

$$= \mathcal{O}(\varepsilon^i), \quad (3.18d)$$

where $\Gamma(\cdot)$ and $\Gamma(\cdot, \cdot)$ denote the gamma function and the upper incomplete gamma function, respectively. Since (ii) implies that $p_{\varepsilon,1} = \mathcal{O}(\varepsilon)$, the operator vanishes if we let $\varepsilon \rightarrow 0$. We set $\varepsilon = \beta/n_q$ in the discrete case so that the angular diffusion vanishes if the number of ordinates n_q tends to infinity. This analysis shows that the product $\sigma_{as} \cdot \beta$ controls the strength of the added angular diffusion.

The asymptotic analysis that follows is a one-to-one translation of the original analysis [81] and substitutes the new scattering kernel. We omit the source term and shorten notation to get

$$\underbrace{\partial_t \psi(t, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla \psi + \sigma_a \psi}_{=: L\psi} = \sigma_s \int_{\mathbb{S}^2} \frac{1}{4\pi} (\psi(\boldsymbol{\Omega}') - \psi(\boldsymbol{\Omega})) d\boldsymbol{\Omega}' + \sigma_{as} \int_{\mathbb{S}^2} s_\varepsilon(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') (\psi(\boldsymbol{\Omega}') - \psi(\boldsymbol{\Omega})) d\boldsymbol{\Omega}', \quad (3.19)$$

which we further rewrite as

$$L\psi + (\sigma_a + \sigma_s)\psi = \sigma_s \phi + \sigma_{as} \int_{\mathbb{S}^2} s_\varepsilon(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') (\psi(\boldsymbol{\Omega}') - \psi(\boldsymbol{\Omega})) d\boldsymbol{\Omega}'. \quad (3.20)$$

Next, define $c_\varepsilon = \int_{\mathbb{S}^2} s_\varepsilon(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') d\boldsymbol{\Omega}'$ to get

$$L\psi + (\sigma_a + \sigma_s + \sigma_{as}c_\varepsilon)\psi = \sigma_s \phi + \sigma_{as} \int_{\mathbb{S}^2} s_\varepsilon(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') \psi(\boldsymbol{\Omega}') d\boldsymbol{\Omega}'. \quad (3.21)$$

Writing the scattering kernel in terms of Legendre polynomials with coefficients $s_{\varepsilon,n} = 2\pi \int_{-1}^1 s_\varepsilon(\mu) P_n(\mu) d\mu$, we obtain

$$\underbrace{L\psi + (\sigma_a + \sigma_s + \sigma_{as}c_\varepsilon)\psi}_{=: \text{LHS}} = \sigma_s \phi + \sigma_{as} \int_{\mathbb{S}^2} \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} s_{\varepsilon,n} P_n(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') \psi(\boldsymbol{\Omega}') d\boldsymbol{\Omega}'. \quad (3.22)$$

We can express $P_n(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}')$ and $\psi(\boldsymbol{\Omega})$ in terms of spherical harmonics as

$$P_n(\boldsymbol{\Omega} \cdot \boldsymbol{\Omega}') = \sum_{l=-n}^n a_{n,l} Y_{n,l}(\boldsymbol{\Omega}) \overline{Y_{n,l}(\boldsymbol{\Omega}')} \quad (3.23)$$

and

$$\psi(\boldsymbol{\Omega}) = \sum_{n=0}^{\infty} \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} \psi_{n,m} Y_{n,m}(\boldsymbol{\Omega}). \quad (3.24)$$

Combining the last three equations yields

$$\text{LHS} = \sigma_s \phi + \sigma_{as} \int_{\mathbb{S}^2} \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} s_{\varepsilon,n} \sum_{l=-n}^n a_{n,l} Y_{n,l}(\boldsymbol{\Omega}) \overline{Y_{n,l}(\boldsymbol{\Omega}')} \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} \psi_{n,m} Y_{n,m}(\boldsymbol{\Omega}') d\boldsymbol{\Omega}'. \quad (3.25)$$

Using orthogonality $\int_{\mathbb{S}^2} Y_{n,m}(\boldsymbol{\Omega}) \overline{Y_{k,l}}(\boldsymbol{\Omega}) = \frac{4\pi}{2n+1} \frac{1}{a_{n,m}} \delta_{nk,ml}$ then yields

$$\text{LHS} = \sigma_s \phi + \sigma_{as} \int_{\mathbb{S}^2} \sum_{n=0}^{\infty} \frac{2n+1}{2} s_{\varepsilon,n} \sum_{l=-n}^n a_{n,l} Y_{n,l}(\boldsymbol{\Omega}) \overline{Y_{n,l}(\boldsymbol{\Omega}')} \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} \psi_{n,m} Y_{n,m}(\boldsymbol{\Omega}') d\boldsymbol{\Omega}' \quad (3.26a)$$

$$= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} s_{\varepsilon,n} \sum_{l=-n}^n a_{n,l} Y_{n,l}(\boldsymbol{\Omega}) \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} \psi_{n,m} \int_{\mathbb{S}^2} Y_{n,m}(\boldsymbol{\Omega}') \overline{Y_{n,l}(\boldsymbol{\Omega}')} d\boldsymbol{\Omega}' \quad (3.26b)$$

$$= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} s_{\varepsilon,n} \sum_{l=-n}^n a_{n,l} Y_{n,l}(\boldsymbol{\Omega}) \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} \psi_{n,m} \frac{4\pi}{2n+1} \frac{1}{a_{n,m}} \delta_{nn,ml} \quad (3.26c)$$

$$= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} s_{\varepsilon,n} \sum_{m=-n}^n a_{n,m} Y_{n,m}(\boldsymbol{\Omega}) \frac{2n+1}{4\pi} a_{n,m} \psi_{n,m} \frac{4\pi}{2n+1} \frac{1}{a_{n,m}} \quad (3.26d)$$

$$= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} s_{\varepsilon,n} \sum_{m=-n}^n a_{n,m} \psi_{n,m} Y_{n,m}(\boldsymbol{\Omega}) \quad (3.26e)$$

$$= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} Y_{n,m}(\boldsymbol{\Omega}) \psi_{n,m} 2\pi \int_{-1}^1 s_{\varepsilon}(\mu) P_n(\mu) d\mu \quad (3.26f)$$

$$= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{2} \int_{-1}^1 s_{\varepsilon}(\mu) P_n(\mu) d\mu \sum_{m=-n}^n a_{n,m} Y_{n,m}(\boldsymbol{\Omega}) \psi_{n,m}. \quad (3.26g)$$

Summarizing the above derivation, we obtain

$$L\psi + (\sigma_a + \sigma_s + \sigma_{as} c_{\varepsilon})\psi = \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{2} \int_{-1}^1 s_{\varepsilon}(\mu) P_n(\mu) d\mu \sum_{m=-n}^n a_{n,m} \psi_{n,m} Y_{n,m}(\boldsymbol{\Omega}). \quad (3.27)$$

Next, we define $\hat{s}_{\varepsilon}(\mu) = s_{\varepsilon}(\mu)/c_{\varepsilon}$, substitute it into the above equation, and drop the hat to get

$$L\psi + (\sigma_a + \sigma_s + \sigma_{as})\psi = \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{2} \int_{-1}^1 s_{\varepsilon}(\mu) P_n(\mu) d\mu \sum_{m=-n}^n a_{n,m} \psi_{n,m} Y_{n,m}(\boldsymbol{\Omega}). \quad (3.28)$$

Let now $\sigma_{as} = \hat{\sigma}_{as}/\Delta$ with $\hat{\sigma}_{as} = \mathcal{O}(1)$. The fast variable y is given by $y = (1 - \mu)/\delta$ and $\varepsilon = \delta$. Here, $\delta, \Delta \ll 1$. Then $s_\varepsilon(\mu) = \hat{s}_\varepsilon(y)$ and we obtain

$$\begin{aligned} \text{LHS} &= \sigma_s \phi + \frac{\hat{\sigma}_{as}}{\Delta} \sum_{n=0}^{\infty} \frac{2n+1}{2} \underbrace{\int_{-1}^1 \hat{s}_\varepsilon \left(\frac{1-\mu}{\delta} \right) P_n(\mu) d\mu}_{= \delta \int_0^{2/\delta} \hat{s}_\varepsilon(y) P_n(1 - \delta y) dy} \underbrace{\sum_{m=-n}^n a_{n,m} \psi_{n,m} Y_{n,m}(\Omega)}_{= \delta \int_0^{2/\delta} \hat{s}_\varepsilon(y) (P_n(1) - \delta y P'_n(1) + \mathcal{O}(\delta)^2) dy} \\ &\quad (3.29a) \end{aligned}$$

$$= \sigma_s \phi + \hat{\sigma}_{as} \frac{\delta}{\Delta} \sum_{n=0}^{\infty} \frac{2n+1}{2} \int_0^{2/\delta} \hat{s}_\varepsilon(y) (P_n(1) - \delta y P'_n(1) + \mathcal{O}(\delta)^2) dy \underbrace{\sum_{m=-n}^n a_{n,m} \psi_{n,m} Y_{n,m}(\Omega)}_C. \quad (3.29b)$$

Together with $P_n(1) = 1$ and $P'_n(1) = n(n+1)/2$ and changing back according to $y = (1 - \mu)/\delta$ results in

$$\begin{aligned} \text{LHS} &= \sigma_s \phi + \hat{\sigma}_{as} \frac{1}{\Delta} \sum_{n=0}^{\infty} \frac{2n+1}{2} \int_{-1}^1 \hat{s}_\varepsilon \left(\frac{1-\mu}{\delta} \right) d\mu \cdot C \\ &\quad - \hat{\sigma}_{as} \frac{n(n+1)}{2} \frac{1}{\Delta} \sum_{n=0}^{\infty} \frac{2n+1}{2} \int_{-1}^1 \hat{s}_\varepsilon \left(\frac{1-\mu}{\delta} \right) (1-\mu) d\mu \cdot C \\ &\quad + \mathcal{O} \left(\frac{\delta^2}{\Delta} \right). \end{aligned} \quad (3.30)$$

Reversing the fast-to-slow transformation yields

$$\begin{aligned} \text{LHS} - \mathcal{O} \left(\frac{\delta^2}{\Delta} \right) &= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \frac{2n+1}{2} \overbrace{\int_{-1}^1 s_\varepsilon(\mu) d\mu}^{=1/(2\pi)} \cdot C - \sigma_{as} \frac{n(n+1)}{2} \sum_{n=0}^{\infty} \frac{2n+1}{2} \int_{-1}^1 s_\varepsilon(\mu) (1-\mu) d\mu \cdot C. \end{aligned} \quad (3.31)$$

Here,

$$\sigma_{as} = \mathcal{O} \left(\frac{1}{\Delta} \right), \quad (3.32a)$$

$$\sigma_{as} \int_{-1}^1 s_\varepsilon(\mu) (1-\mu) d\mu = \mathcal{O} \left(\frac{\delta}{\Delta} \right). \quad (3.32b)$$

The first line is obvious. The second line follows from the fact that $(1-\mu) = \mathcal{O}(\delta)$, resulting in

$$\begin{aligned} L\psi + (\sigma_a + \sigma_s + \sigma_{as})\psi &= \sigma_s \phi + \sigma_{as} \sum_{n=0}^{\infty} \sum_{m=-n}^n \frac{2n+1}{4\pi} a_{n,m} \psi_{n,m} Y_{n,m}(\Omega) \cdot \left[1 - n(n+1)\pi \int_{-1}^1 s_\varepsilon(\mu) (1-\mu) d\mu \right] + \mathcal{O} \left(\frac{\delta^2}{\Delta} \right). \end{aligned} \quad (3.33)$$

For the next step, we follow the computations (52) and (53) in Pomraning's work [81] to obtain

$$\begin{aligned}
L\psi + (\sigma_a + \sigma_s)\psi &= \sigma_s\phi + \underbrace{\sigma_{as}\pi \int_{-1}^1 s_\varepsilon(\mu)(1-\mu)d\mu}_{= \mathcal{O}(\sigma_{as} \cdot \varepsilon)} \cdot \left[\frac{\partial}{\partial\mu}(1-\mu^2)\frac{\partial}{\partial\mu} + \frac{1}{1-\mu^2}\frac{\partial^2}{\partial\phi^2} \right] \psi + \mathcal{O}(\sigma_{as} \cdot \varepsilon^2),
\end{aligned} \tag{3.34}$$

which are the scaling properties that we expect to see and also observe throughout the numerical experiments.

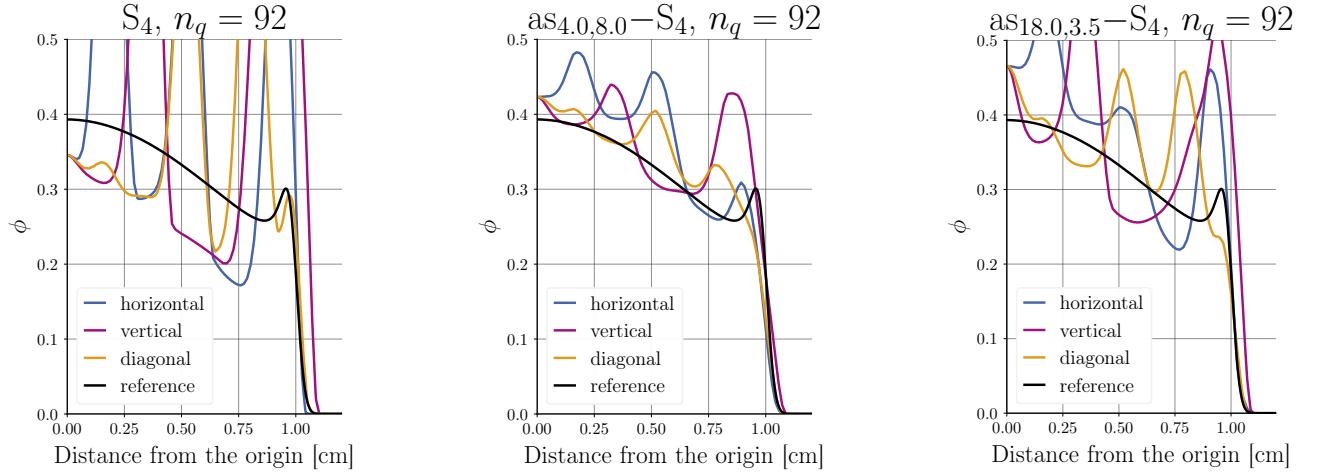
Before discussing the results of both the line-source and lattice test case, it is worth mentioning where and how the theoretical setting of the asymptotic analysis deviates from the actual setting of the numerical experiments. Most importantly, the asymptotic analysis works on the continuous level (in angle) while the as- S_N implementation is already discretized (in angle). Next, when trying to quantify the effect of artificial diffusion, we do so by proxying the L^2 error with respect to a reference solution, rather than the artificial diffusion itself. Lastly, while the effect of angular diffusion is characterized by the product $\sigma_{as} \cdot \varepsilon$, the $\mathcal{O}(\sigma_{as} \cdot \varepsilon^2)$ term might, for certain parameter pairs, be non-negligible to an extend where the difference between the as- S_N and the S_N method can not exclusively be attributed to the angular diffusion.

3.3.3. as- S_N and the line-source test case

Comparing the as- S_N method with the r S_N method, an arguable drawback is the necessity to choose two parameters (i.e., σ_{as} and β) instead of just one (i.e., δ). However, the results from the asymptotic analysis show that (under certain assumptions) the added artificial diffusion is controlled by the product of σ_{as} and β rather than by each parameter individually. This is mostly in agreement with the two parameter studies that are summarized in Figures 3.16 and 3.17, computed for the coarse setting $n_q \times n_x \times n_y = 12 \times 50 \times 50$ and the fine setting $n_q \times n_x \times n_y = 92 \times 200 \times 200$, respectively. In both settings, we (i) run as- S_N simulations for the line-source test case with different combinations of σ_{as} and β , (ii) compute the resulting L^2 error with respect to the semi-analytical reference solution, and (iii) divide that error by the error obtained from a standard S_N simulation. Each cell in the resulting heatmaps in Figures 3.16 and 3.17 is then color-coded by this *baseline normalized error*. Values below (above) unity indicate that the error has decreased (increased) when running the as- S_N method with that specific parameter pair. The optimal parameter pair is highlighted in yellow and all pairs close to optimal in green.

In the coarse case, the optimal parameter choice of $\beta = 3.5$ and $\sigma_{as} = 18$ yields an error reduction down to approximately 43%, measured in the L^2 norm over the full spatial domain at the final time $t = 1s$. Parameter pairs close to optimal, i.e., where the error does not exceed the optimal error by more than 10%, mostly fall in the region $20 \leq \beta \cdot \sigma_{as} \leq 80$, $\beta \geq 3$. For $120 \leq \beta \cdot \sigma_{as}$, the baseline normalized error will be above unity. This corresponds to cases in which too much artificial scattering was added.

It is obviously infeasible to perform a full parameter study on the same fine grid that the actual computation takes place on. Consequently, we may ask ourselves how well the coarse setting performs as a surrogate for the fine setting. Comparing the heatmap in Figure 3.16 with the one in Figure 3.17, the success of this strategy is evident. While the optimal parameter pair in



(a) S_4 with 92 ordinates and no artificial scattering, computed with the icosahedron quadrature set.

(b) Using artificial scattering with the optimal parameter set reduces the error down to about 23%.

(c) Using the optimal parameter set computed in the coarse setting still reduces the error down to about 43%.

Figure 3.15.: For the as- S_N method, two parameters (σ_{as} and β) need to be chosen. All three images show line-source cuts with $n_q \times n_x \times n_y = 92 \times 200 \times 200$. Left, artificial scattering is not used. Center, $\sigma_{\text{as}} = 4.0$ and $\beta = 8.0$, which corresponds to the optimal parameter found in the parameter study in Figure 3.17 on the same grid. Right, $\sigma_{\text{as}} = 18.0$ and $\beta = 3.5$, which are the optimal parameters from the coarse parameter study ($n_q \times n_x \times n_y = 12 \times 50 \times 50$).

the fine setting reduces the error down to 23%, the optimal parameter pair in the coarse setting still reduces the error down to 42% when applied in the fine setting. This is also visualized in Figure 3.15. The regions of optimal parameter configurations tend to overlap when comparing both settings, although the fine setting has a wider valley of optimal parameter pairs.

Computing an as- S_N simulation with $n_q \times n_x \times n_y = 92 \times 200 \times 200$ is approximately 490 ($\approx 92/12 \times 4 \times 4 \times 4$, since also $n_t = \mathcal{O}(n_x, n_y)$) times as costly as for $n_q \times n_x \times n_y = 12 \times 50 \times 50$. Therefore, performing *all* 306 simulations for the coarse heatmap, followed by one simulation in the fine setting with the optimal parameter pair increases the runtime by 62% while the error is decreased down to 42%. A similar return on investment would not be achieved by simply increasing the number of ordinates in the S_N method.

Lastly, the discussion of an efficient implementation of the as- S_N method has been omitted since all computations were done explicitly. Because the physical and artificial scattering can be combined into a new scattering kernel, the as- S_N method yields no increase in runtime.

Parameter study for the as- S_N method and the line-source test case ($n_x = n_y = 50$ and $n_q = 12$)

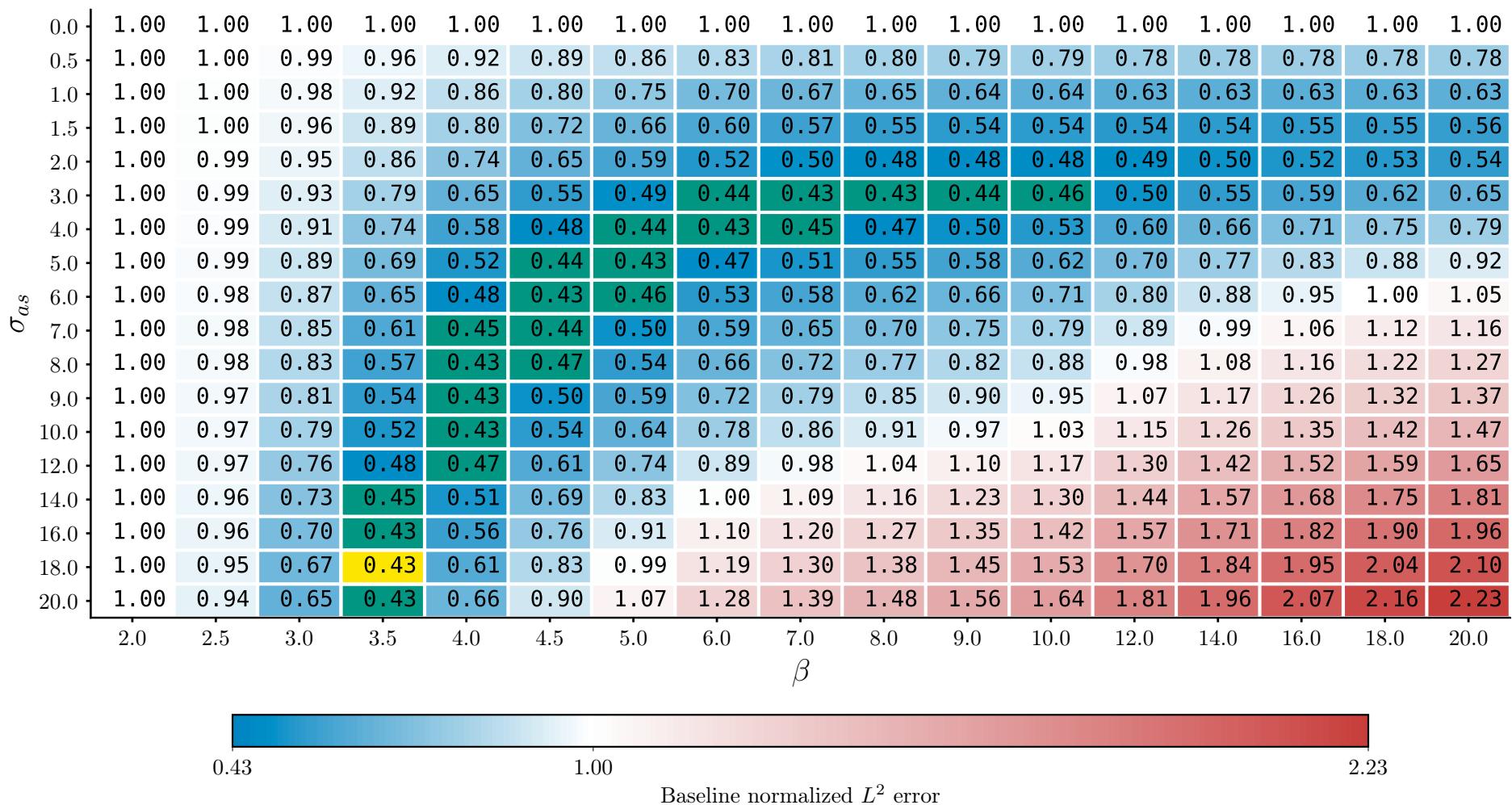


Figure 3.16.: Parameter study for σ_{as} and β on the grid $n_q \times n_x \times n_y = 12 \times 50 \times 50$ in an explicit calculation. For every set of parameters, we compute the L^2 error of the scalar flux ϕ with respect to the semi-analytical reference solution on the same spatial grid. The number in each field of the heatmap is then the baseline normalized error, i.e., the L^2 error obtained for that specific parameter configuration divided by the error obtained without artificial scattering. The optimal parameter configuration, highlighted in yellow, reduces the error down to 43%. Close-to-optimal parameter pairs are highlighted in green.

Parameter study for the as- S_N method and the line-source test case ($n_x = n_y = 200$ and $n_q = 92$)

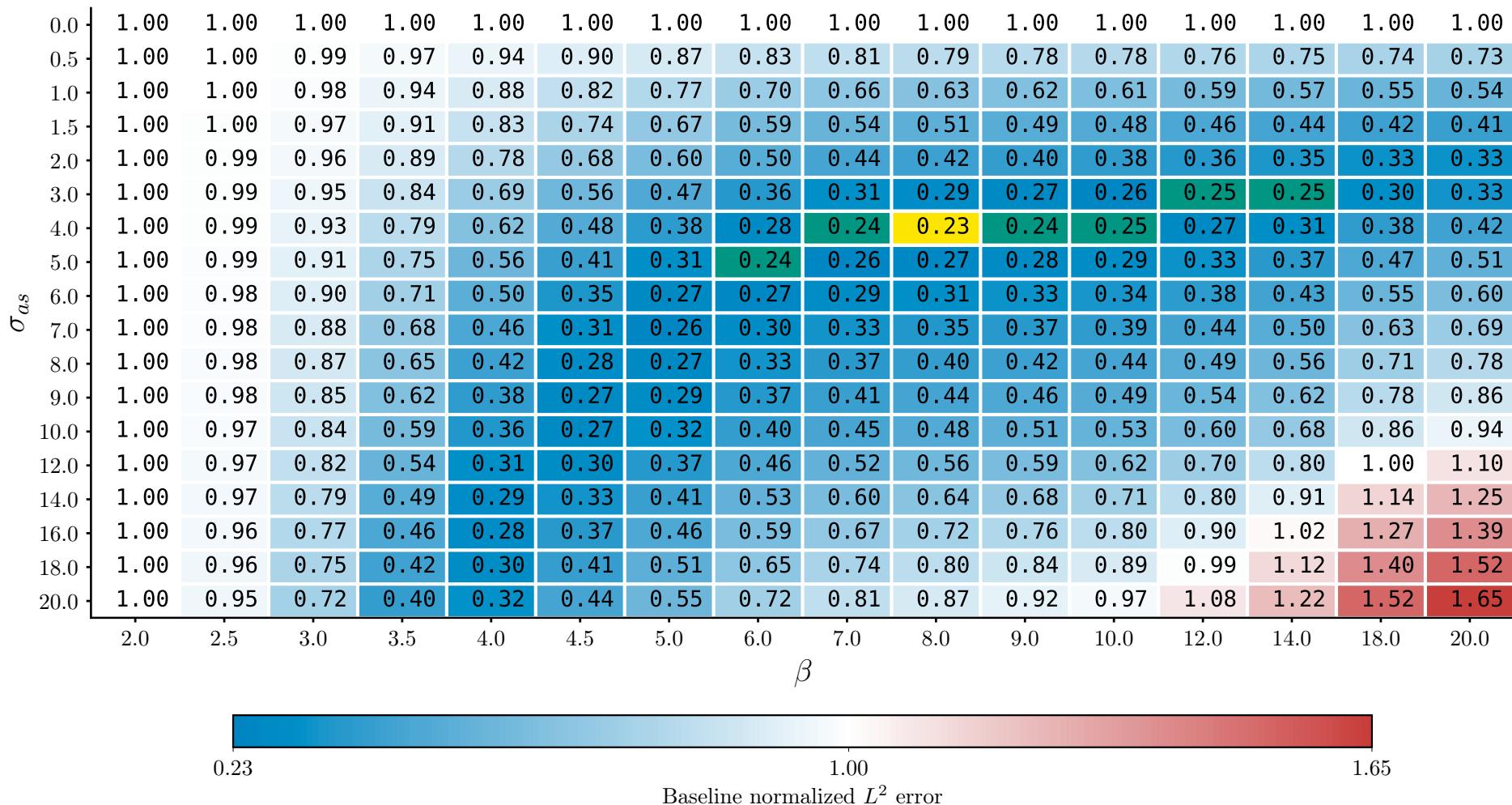


Figure 3.17.: Parameter study for σ_{as} and β on the grid $n_q \times n_x \times n_y = 92 \times 200 \times 200$ in an explicit calculation. For every set of parameters, we compute the L^2 error of the scalar flux ϕ with respect to the semi-analytical reference solution on the same spatial grid. The number in each field of the heatmap is then the baseline normalized error, i.e., the L^2 error obtained for that specific parameter configuration divided by the error obtained without artificial scattering. The optimal parameter configuration, highlighted in yellow, reduces the error down to 23%. Close-to-optimal parameter pairs are highlighted in green.

3.3.4. as- S_N and the lattice test case

The quantitative benefits of artificial scattering are, unsurprisingly, less pronounced in the lattice test case. This is predominantly due to the fact that ray effects occur on neglectable scales, orders of magnitudes smaller than the mean scalar flux. Nevertheless, we will observe an error reduction when switching from S_N to as- S_N .

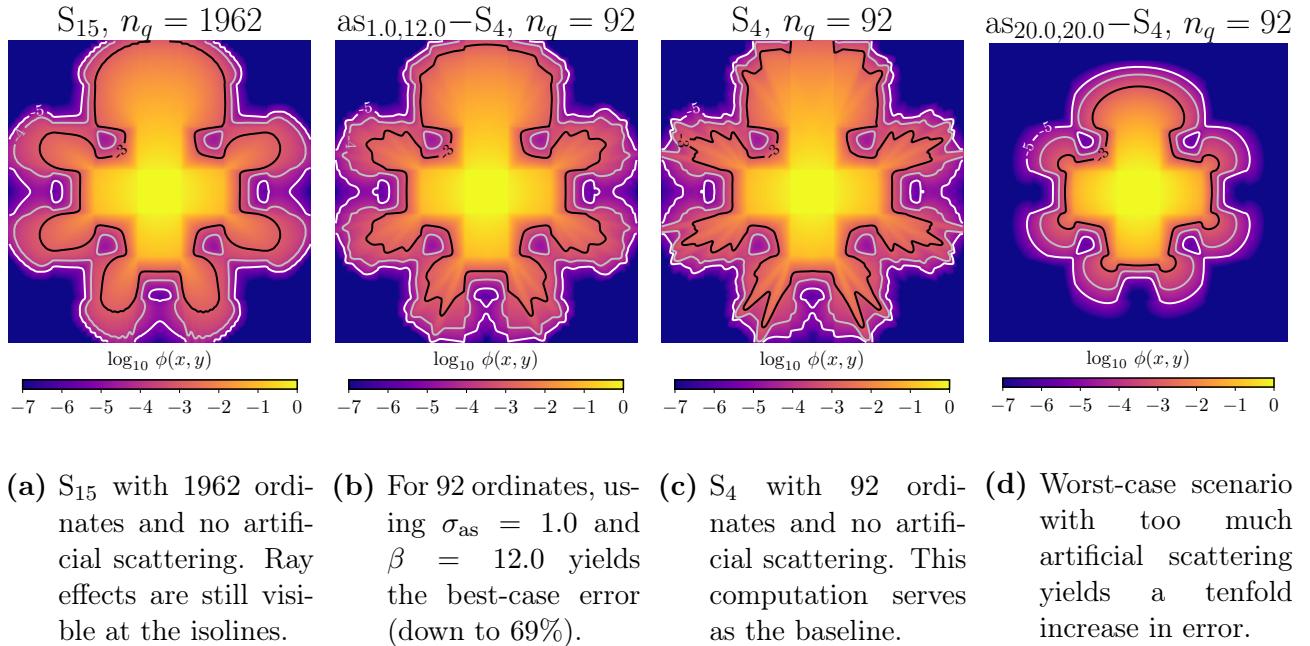


Figure 3.18.: Results for the lattice test case with different amounts of artificial scattering. From left to right: A high-order reference solution; the best-case scenario; the baseline; and the worst-case scenario.

To derive errors, we computed a reference solution that uses $n_q = 1962$ ordinates on a spatial grid of $n_x \times n_y = 280 \times 280$ cells, visualized in Figure 3.18a. The parameter study that was performed for the line-source test case is replicated for the lattice test case ($n_q = 92$), with the results summarized in Figure 3.19. The optimal parameter configuration ($\beta = 12$, $\sigma_{as} = 1$) reduces the error down to 69% of the original error, shown with the baseline S_N result in Figures 3.18b and 3.18c, respectively.

Interestingly, using a close-to-optimal parameter pair from the coarse line-source parameter study for the lattice test case reduces the error down to 72%, indicating the possibility to use different discretizations and test cases as surrogates.

While the worst-case error for the line-source test case was reasonably small with a baseline normalized error of 2.23 (coarse) and 1.65 (fine), this is not the case for the lattice test case where we observe a baseline normalized error of up to 10.8 in Figure 3.18d. The error is due to the enormous amount of artificial diffusion that is added to the solution and smears out the scalar flux unnecessarily.

Parameter study for the as- S_N method and the lattice test case ($n_x = n_y = 280$ and $n_q = 92$)

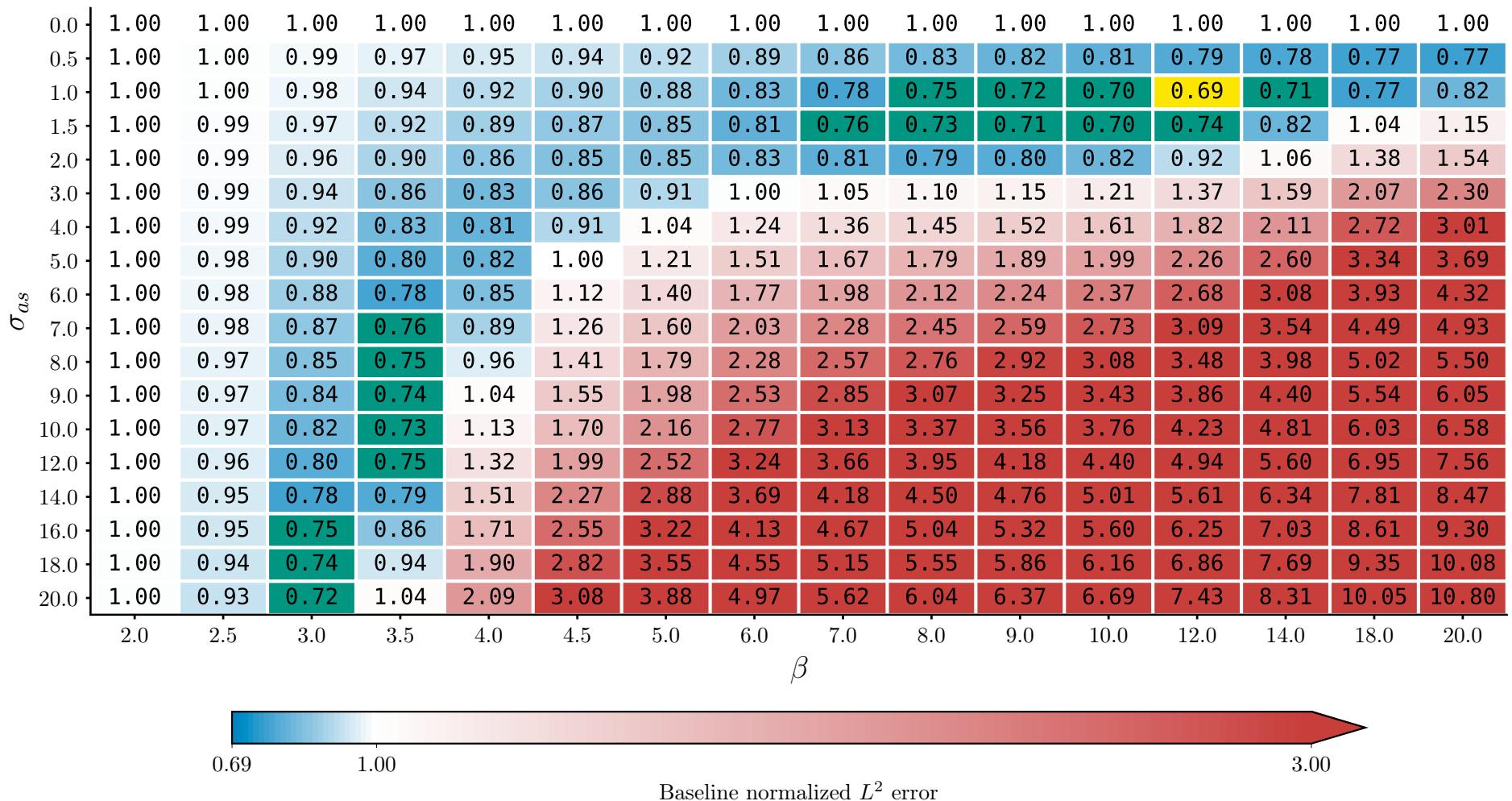


Figure 3.19.: Parameter study for σ_{as} and β on the grid $n_q \times n_x \times n_y = 42 \times 280 \times 280$ in an explicit lattice calculation. For every set of parameters, we compute the L^2 error of the scalar flux ϕ with respect to a reference solution ($n_q = 1962$) on the same spatial grid. The number in each field of the heatmap is then the baseline normalized error, i.e., the L^2 error obtained for that specific parameter configuration divided by the error obtained without artificial scattering. The optimal parameter configuration, highlighted in yellow, reduces the error down to 69%. Close-to-optimal parameter pairs are highlighted in green. The colorbar ranges to a maximum value of 3.00 to avoid color distortion.

Part II.

Non-classical transport

4. Non-classical transport equations

Not all particles are created equal, for some engage in billiard games that violate Section 1’s fundamental rule: Background obstacles ought to be uncorrelated. If they are not—and we will go through various examples where this rule is violated—*classical* transport equations, i.e., those that we have encountered so far, model the underlying transport phenomena incorrectly. Instead, so-called *non-classical* transport equations emerge, augmenting the phase space of classical transport equations by an additional variable. This extra variable can be interpreted both as the distance to collision or the distance since collision, depending on the precise formulation of the non-classical transport equation.

Different from classical transport in Part I, non-classical transport, to which Part II of this thesis is devoted to, is much more novel in its application to radiative transfer. This section motivates the necessity of non-classical transport by three applications before delving into further modeling steps that ultimately lead to various ways to generalize classical transport, one of which is the *generalized linear Boltzmann equation* (GLBE). Throughout Part II, we will discuss and contribute to so-far unsolved questions regarding the modeling of non-classical transport in the presence of heterogeneities.

Non-classical transport dates back to the late 2000s and early 2010s where it was independently formalized by Caglioti and Golse [18, 41] and Larsen and Vasques [56]. Caglioti and Golse discuss the Boltzmann-Grad limit for the periodic Lorentz gas where scatterers are arranged on a lattice rather than at random positions. Larsen and Vasques provide an initial study of non-classical transport, together with the derivation of the respective asymptotic diffusion limit. Numerical test cases show the superiority of non-classical transport equations when modeling transport in a pebble bed reactor. Davis and Xu [32] discussed linear transport in spatially correlated stochastic media in 2014.

Noteworthy, too, are the recent advances in the computer graphics community. Non-classical transport has been used to enhance the quality and realism of rendered images [14, 33, 48].

We postpone an extensive discussion of the relevant equations of non-classical transport for now and start off with the first of three examples that demonstrate the violation of classical transport. These examples are deliberately chosen to cover different applications and fields of research to showcase the universality of non-classical transport.

Example 4.0.1 (Radiation in clouds). According to NASA¹, “about 67 percent of Earth’s surface is typically covered by clouds,” illustrated in Figure 4.1. Consequently, clouds influence the climate in a variety of ways, one of which is through the interaction with solar radiation. In the mid to late 1990s, however, observations of both under- and overestimation of the solar radiation were reported in the literature [27, 83, 80, 101, 6], ranging from -10W/m^2 to $+30\text{W/m}^2$, respectively. With the help of experimental data, Pfeilsticker [79] discovered that the geometric

¹The quote is taken from <https://www.nasa.gov/image-feature/cloudy-earth>, which itself paraphrases the work published by King et al. [53].

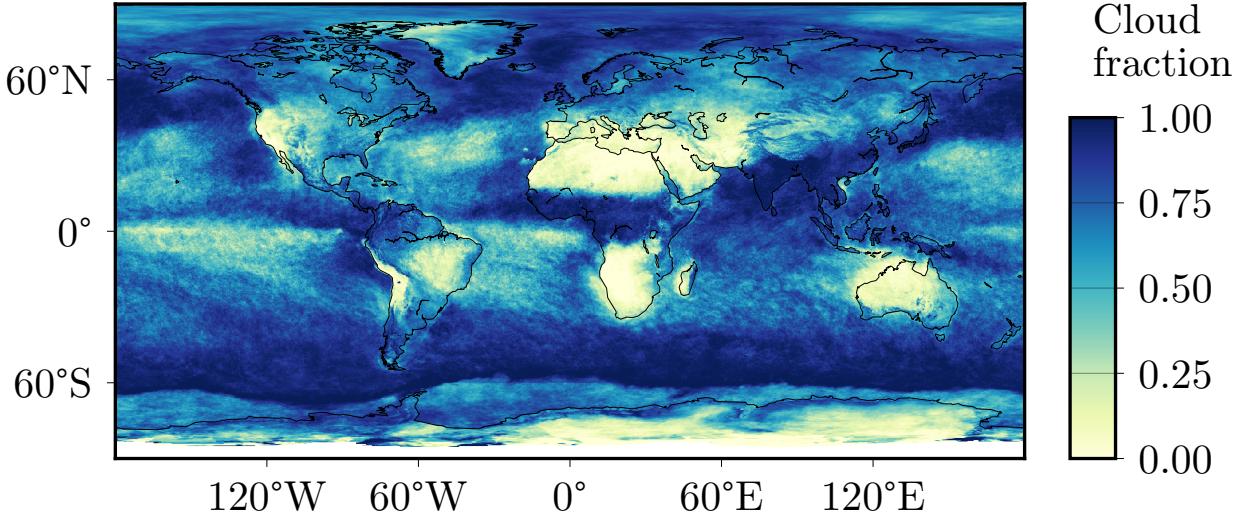


Figure 4.1.: For the month of July 2020, the image shows the averaged cloud fraction with a resolution of 0.25 degrees. The data was recorded with the Aqua/MODIS satellite by NASA Earth Observations. This visualization reconstructs the image by Reto Stockil, hosted on the NASA website [34] (together with the corresponding data).

path length distribution of photons is Lévy distributed ($\sim s^{-\gamma-1}$ for large s) with Lévy index $1 \leq \gamma \leq 2$. This is in stark contrast to Chapter 1, where the path length distribution for the billiard game was exponentially decaying, not algebraically.

When model and data do not match, it is often a good idea to revise the model (rather than, say, the data). Indeed, the modeling assumption that was made does not hold true in reality: Water droplets, a photon's scatterer, are not distributed randomly. Within clouds, droplets cluster together in some regions while being completely absent in others. Consequently, there is an increased (decreased) probability that a particle will scatter if it has (has not) scattered recently, rendering the dynamics non-Markovian. As such, the process can not be described with classical transport equations.

Example 4.0.2 (Pebble-bed reactors). “Nuclear power generated around 10% of the world’s electricity in 2019,” reads a report by the International Atomic Energy Agency. Even though nuclear energy is debated controversially, entirely undisputed, however, is the demand that its mathematical modeling and related predictions be accurate.

Pebble-bed reactors (PBR) are a particular kind of graphite-moderated, gas-cooled, very-high-temperature reactors. They get their name from the spherical pebbles that fuel the reactor. Stacked randomly on top of another, thousands of these pebbles are placed inside a cylindrical reactor core. With an average packing fraction of 81.7%, radiation is modeled by the diffusion approximation which uses the atomic mix model to compute the diffusion coefficient as a weighted average of a pebble’s cross section (weighted with 0.817) and the cross section of vacuum (weighted with 0.183) [97, 56, 98].

When compared with Monte Carlo simulations, the diffusion approximation’s error is small but non-negligible. While it might be argued that the discrepancy between both simulations is due to the inapplicability of the diffusion approximation in that particular regime of transport, the asymptotic diffusion limit of a *non-classical* transport equation reduces the error by a



Figure 4.2.: A cloud rendered with a non-exponentially decaying path length distribution. Light penetrates deeper than in the case of exponentially decaying path lengths. This image was provided by Bitterli and a side-by-side comparison of classical and non-classical renders can be found in Bitterli et al. [14]. Heterogeneity is introduced into the simulation by assuming a statistically heterogeneous material with identical correlation, similar to Camminady et al. [20].

factor two to five for the same setup [98]. Moreover, when allowing for angular-dependent and non-exponential path length distributions, even asymmetries due to gravity (which, in three dimensions, acts only in the z -direction) can be incorporated sufficiently well into non-classical equations. Ultimately, the reason why non-classical equations predict radiation in PBRs more accurately is that a particle's path lengths are not exponentially distributed due to the way the pebbles are stacked.

Example 4.0.3 (Non-classical transport in the field of computer graphics). Produced by Pixar Animation Studios and released by Walt Disney Pictures in 1995, *Toy Story* was the first-ever entirely computer-animated feature film. Its success—*Toy Story* was praised both by the audience and critics and achieved four Academy Awards in 1996—arguably propelled the animation industry as a whole. The high degree of realism, albeit multifaceted, is largely contributed to the (for its time) unparalleled sophistication of Pixar's rendering software RenderMan. For objects to look real, not only had their movements to be modeled correctly but also every frame's illumination. Since light transport is described by kinetic theory's transport equations, there exists an inherent link between the work discussed thus far and that in the field of computer graphics.

While renders of hard, smooth surfaces—*Cars*'s cars or *Toy Story*'s toys—achieve photorealism, some notoriously difficult settings were only recently mastered. This includes, e.g., fur and hair, rough and translucent objects, skin, and—relevant for non-classical transport—clouds. In 2018,

Bitterli et al. [14] explored the possibility to model light transport through clouds with heavy-tailed transmittance curves which allowed for “both deeper light penetration as well as a softer appearance near the surface.” A resulting render is shown in Figure 4.2.

Moreover, non-classical transport increases artistic flexibility, useful to fine-tune renders to look more realistic or appealing by freely choosing the path length distribution. Since scenes in animation movies rarely consist of homogeneous, luminiferous aether, non-classical transport in presence of heterogeneities needs to be understood both from a theoretical and application-oriented viewpoint.

4.1. Particle billiard redux

The particles’ interactions with the background obstacles don’t differ in their dynamics—particles still undergo elastic, instantaneous collisions—but in the way the scatterers are arranged.

With randomly distributed scatterers representing one end of the spectrum of correlation, scatterers that build a perfect lattice fall on the other end; such a situation is depicted with a single particle’s trajectory in Figure 4.3. To quantify the distribution of path lengths, a distinction needs to be made that was not necessary for the random billiard table: Tallying a single particle’s trajectory for n steps is different from tallying n particles’ trajectories for a single step. For a single particle, only one path length (the very first) can be smaller than the minimal distance between two obstacles’ boundaries. Afterwards, a particle that reflects from an obstacle will not encounter another obstacle before having traveled this minimum distance. But the different statistics are not only limited to the infinitesimally small region. For asymptotically large path lengths, the two resulting distributions decay with different orders of magnitude. These findings are summarized in Figure 4.4. This difference in behavior requires a more precise language.

Definition 4.1.1 (Distance to collision). *A particle at $(\mathbf{x}, \Omega) \in \mathbb{R}^d \times \mathbb{S}^{d-1}$ is said to have distance to collision s , if it undergoes its next collision (where it scatters or gets absorbed) at $\mathbf{x} + s \cdot \Omega$.*

Definition 4.1.2 (Distance since collision). *A particle at $(\mathbf{x}, \Omega) \in \mathbb{R}^d \times \mathbb{S}^{d-1}$ is said to have distance since collision s , if it underwent its last collision (where it scattered) at $\mathbf{x} - s \cdot \Omega$.*

Definition 4.1.3 (Distance since birth). *A particle at $(\mathbf{x}, \Omega) \in \mathbb{R}^d \times \mathbb{S}^{d-1}$ is said to have distance since birth s , if, at $\mathbf{x} - s \cdot \Omega$, it entered the domain through the boundary or was emitted by a source.*

Definition 4.1.4 (Distance between collisions). *A particle is said to have distance between collisions s , if it travels a distance s between a scattering event and the consecutive collision (where it scatters or gets absorbed).*

Consequentially, Figures 4.4a and 4.4b show the distances to collisions and the distances between collision (except for the very first of the 10^6 path lengths), respectively.

Similar to the random case, we can also perform a Boltzmann-Grad limit for the periodic Lorentz gas. The details of this procedure and the mathematical analysis, however, are beyond the scope

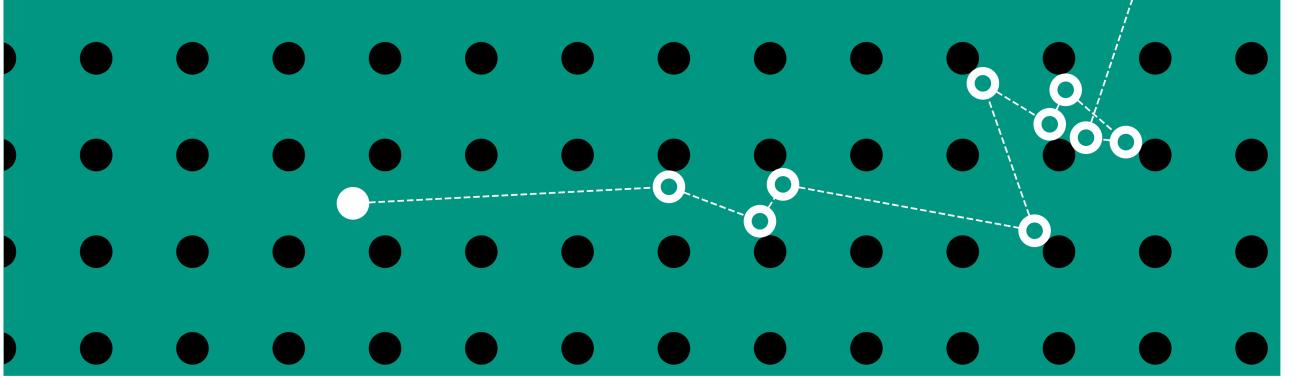


Figure 4.3.: A non-classical billiard game where scatterers' centers are arranged according to a lattice structure. This setup is also known under the name *periodic Lorentz gas*. The (solid) white ball's trajectory is shown.

of this work and are explained thoroughly in the work by Golse et al. [41, 42]. Moreover, non-classical transport is not restricted to the periodic Lorentz gas but a more general phenomena. Nevertheless, two essential properties are exemplified. Both of these can be understood without executing the Boltzmann-Grad limit in a formal manner and instead following Terrence Tao's idea that “[t]here's more to mathematics than rigor and proofs.”

First—though somewhat counterintuitive—it is worth noting that in the limit of $r \rightarrow 0$ with $n_{\text{obs}} \cdot r = \text{const.}$, transport in the periodic Lorentz gas is isotropic since the effect of the lattice orientation vanishes. While the lattice imposes a preference for particles to travel along horizontal and vertical *channels*, shrinking the obstacles' radii effectively opens up more and more channels along all possible directions. This is illustrated in Figure 4.5, where we see trajectories for particles in a periodic Lorentz gas with radii $r = 0.02$, $r = 0.002$, and $r \approx 0.00025$, respectively. The higher the number of obstacles, the smaller the preference for horizontal and vertical directions.

Second—and while this property is not surprising, its consequences are—the path length distributions (all four that were defined earlier) are non-exponential. Consider (without loss of generality) the distance since a particle's previous interaction. If the respective probability were to be exponential,

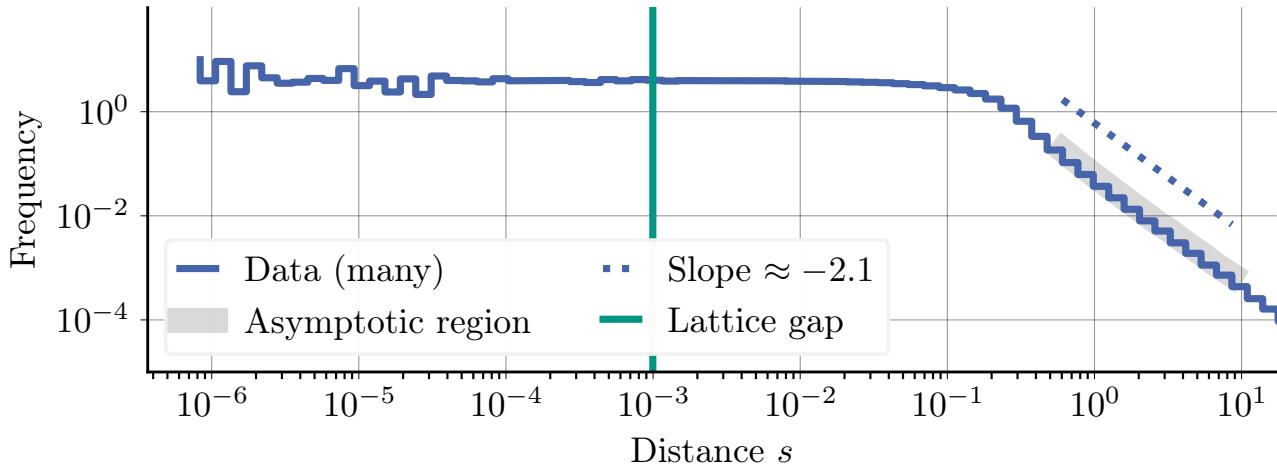
$$\Pr(S > s + \Delta s | S > s) = \Pr(S > \Delta s). \quad (4.1)$$

However, since this is not the case,

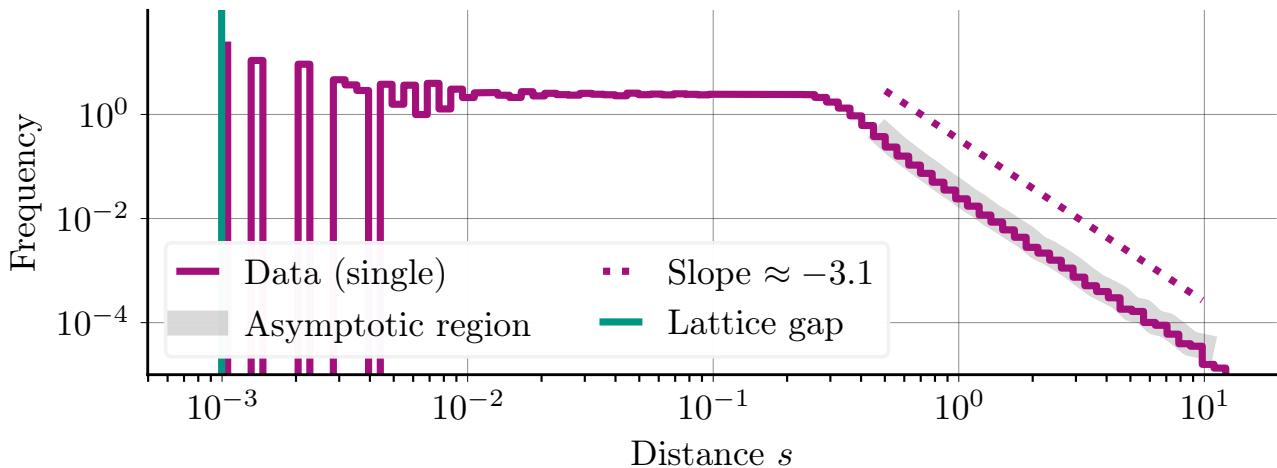
$$\Pr(S > s + \Delta s | S > s) \neq \Pr(S > \Delta s), \quad (4.2)$$

and as a consequence particles must have a *memory*. This memory manifests itself in the fact that the distance traveled by a particle influences its probability to undergo collisions in the future. Thinking back to the example of the periodic Lorentz gas, a particle that has already traveled a long distance might move through one of the domain's channels and is therefore likely to continue unhinderedly. Conversely, a particle that just underwent a collision might be reflected towards a nearby obstacle, scattering again almost immediately. Instead of expressing this memory in terms of the path length distribution, an equivalent formulation makes the total cross section dependent on the distance since collision s , since

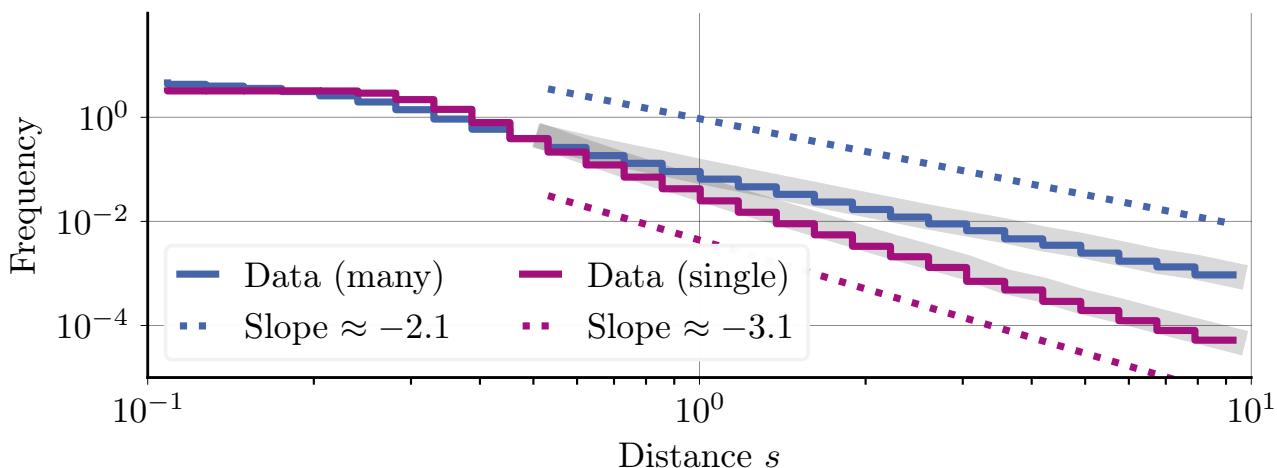
$$\sigma_t(s) = \frac{p(s)}{1 - \int_0^s p(s') ds'}. \quad (4.3)$$



- (a) Path lengths of 10^6 particles from their birth (at a random position inside the lattice, not touching any obstacle) to their first collision. These path lengths are the *distances to collisions*. Distances smaller than the lattice gap occur frequently.



- (b) Path lengths of a single particle's trajectory over the course of 10^6 consecutive collisions. At most one (the first) path length is smaller than the lattice gap. Except for the first path length, these are the *distances between collisions*. The result, especially the asymptotic region, looks very similar when the experiment is repeated with other random initial positions and velocities.



- (c) Both images combined with zoom. The histogram is recomputed with matching binning.

Figure 4.4.: Path length distributions in the periodic Lorentz gas with $\sigma = 1.0$ and 10^6 obstacles, which results in a radius of $2 \cdot 10^{-6}$ for each obstacle. The lattice gap—the minimal distance between obstacles' boundaries—is 0.000996. The first two plots use 100 log-spaced bins, the last plot uses 30.

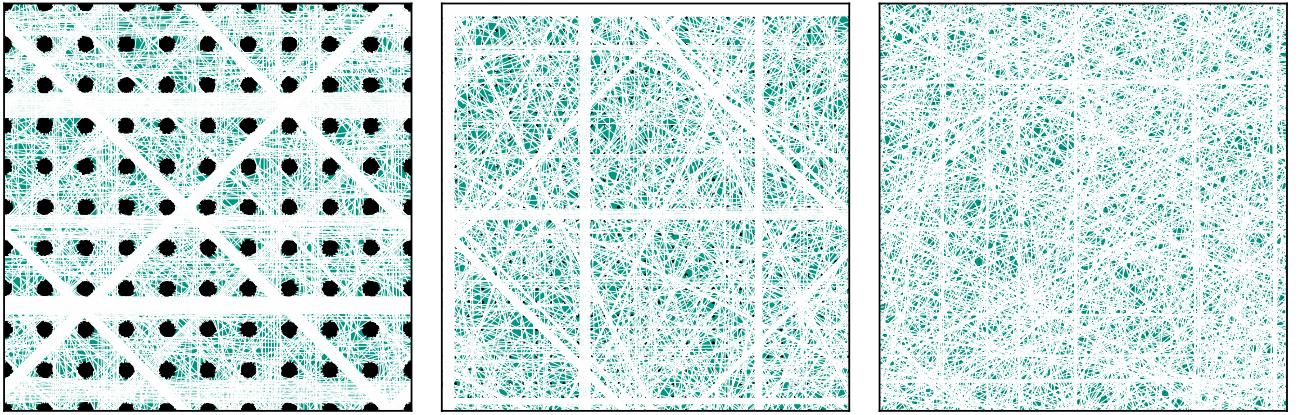


Figure 4.5.: Trajectory of 2000 particles from their birth (at a random position, not touching an obstacle) to their first collision. For all three images, $n_{\text{obs}} \cdot r$ is constant. The orientation of the lattice fades as we increase the number of obstacles, visible in the trajectories. The domains are periodic.

It is easy to verify that this expression is constant if and only if $p(s)$ is an exponential distribution. If, for example,

$$p(s) = se^{-s}, \quad (4.4)$$

then

$$\sigma_t(s) = \frac{s}{1+s} \quad (4.5)$$

and the likelihood to scatter increases as s increases. Alternatively, we can make an algebraic ansatz and assume that p is a Pareto distribution, i.e.,

$$p_{\alpha, x_m}(s) = \begin{cases} 0 & \text{if } s < x_m, \\ \frac{\alpha x_m^\alpha}{s^{\alpha+1}} & \text{if } s \geq x_m. \end{cases} \quad (4.6)$$

For $\alpha = 2$ and $x_m = 1$,

$$\sigma_t(s) = \begin{cases} 0 & \text{if } s < 1, \\ \frac{2}{s} & \text{if } s \geq 1, \end{cases} \quad (4.7)$$

and particles are less likely to scatter the further they travel. Whether the flexibility to choose a path length distribution is reflected in a corresponding arrangement of obstacles is unclear and it seems more likely that certain distributions are not the result of a Boltzmann-Grad-like limit of an arrangement of obstacles.

Finally, to tease Chapter 5, we briefly mention non-classical transport in the presence of material heterogeneities. Here, the following question arises: How do we correctly encode the memory of particles as they transition between the different materials? Given a lack of experimental evidence, arguments for different models can be made—some of which will be presented here. In this chapter, however, we restrict ourselves to the homogeneous case and are now prepared to analyze the governing equations of homogeneous, non-classical transport.

4.2. Generalized linear Boltzmann equations

We start out simple and add more complexity as we go along. For now, we omit the distinction between distance since collision and distance since birth and write the generalized liner Boltzmann equation (GLBE) in terms of the distance since the last interaction. Similar to Larsen and Vasques [56], we make the following definitions:

- V = “the spatial domain with boundary ∂V .”
- \mathbf{x} = “the position $(x, y, z) \in V$.”
- $\mathbf{n}(\mathbf{x}) = \mathbf{n}$ = “the outward pointing normal vector at $\mathbf{x} \in \partial V$.”
- Ω = “the direction of flight $(\Omega_x, \Omega_y, \Omega_z)$ with unit speed.”
- s = “the distance that a particle has traveled since the previous event, be it birth or scattering.”
- v = “the particle speed.”
- $N(s, \mathbf{x}, \Omega) ds d\mathbf{x} d\Omega$ = “the number of particles in $ds d\mathbf{x} d\Omega$ about (s, \mathbf{x}, Ω) .”
- $\psi(s, \mathbf{x}, \Omega) = v N(s, \mathbf{x}, \Omega)$ = “the non-classical angular flux.”
- $\sigma_t(s) ds$ = “the probability that a particle which has traveled a distance s since a previous event experiences a collision while traveling a further distance ds .”
- c = “the probability that a particle that experiences a collision will scatter.”
- $s(\Omega' \cdot \Omega) d\Omega$ = “the probability that a particle with pre-collision direction Ω' scatters into a post-collision direction that lies in $d\Omega$ about Ω .”
- $\delta(s)$ = “the Dirac delta function evaluated at s .”
- $Q(\mathbf{x}) d\mathbf{x}$ = “the rate at which particles are isotropically emitted by an internal source in $d\mathbf{x}$ about \mathbf{x} .”
- $l(\mathbf{x}, \Omega)$ = “the distance to the domain’s boundary from \mathbf{x} in direction $-\Omega$.”

Just like the time-dependent classical transport equation, the GLBE follows from balancing the gain terms (in-scattering and source contributions) with the loss terms (leakage and collisions). Since particles with any value of s can leak or scatter, but we only gain particles with $s = 0$, the gain terms need to be multiplied with the Dirac delta function. The result is then

$$\underbrace{\partial_s \psi(s, \mathbf{x}, \Omega)}_{\text{rate of change}} = - \underbrace{\Omega \cdot \nabla_{\mathbf{x}} \psi(s, \mathbf{x}, \Omega) - \sigma_t(s) \psi(s, \mathbf{x}, \Omega)}_{\text{leakage and collision}} + \underbrace{\delta(s) \int_0^{l(\mathbf{x}, \Omega)} \int_{\mathbb{S}^2} c \sigma_t(s') s(\Omega' \cdot \Omega) \psi(s', \mathbf{x}, \Omega') d\Omega' ds' + \delta(s) Q(\mathbf{x})}_{\text{in-scattering and source}}. \quad (4.8)$$

Written more recognizable, we define the generalized linear Boltzmann equation, formulated in terms of the distance since an interaction, as

$$\begin{aligned} \partial_s \psi(s, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(s, \mathbf{x}, \Omega) + \sigma_t(s) \psi(s, \mathbf{x}, \Omega) \\ = \delta(s) \int_0^{l(\mathbf{x}, \Omega)} \int_{\mathbb{S}^2} c \sigma_t(s') s(\Omega' \cdot \Omega) \psi(s', \mathbf{x}, \Omega') d\Omega' ds' + \delta(s) Q(\mathbf{x}). \end{aligned} \quad (4.9)$$

Some remarks are noteworthy.

Remark. This is the *backward* formulation of the GLBE. Backward in the sense that, given any tuple (s, \mathbf{x}, Ω) , we do know where a particle originated (at $\mathbf{x} - s \cdot \Omega$), but not where it will collide.

Remark. Equation (4.9) looks similar to the time-dependent classical transport equation, but with a time-dependent cross section and additional delta functions on the right-hand side. Indeed, if it weren't for the delta functions, the backward GLBE could be solved with a standard time-dependent transport code that allows to incorporate a dependency on s in the cross section. But because particles have their memory reset after a collision or when they emerge from a source, a numerical algorithm like S_N that evolves transport forward in time does not immediately work here. The Monte Carlo method, however, can be adapted easily.

Remark. When generalizing an equation, it should always reduce to its simpler form if the assumptions are the same. This is the case in (4.9) with $\sigma_t(s) = \sigma_t$ independent of s , followed by operating with $\int_{-\varepsilon}^{\infty} \cdot ds$ and letting $\varepsilon \rightarrow 0$, i.e.,

$$\begin{aligned} \lim_{\varepsilon \rightarrow 0} \int_{-\varepsilon}^{\infty} \partial_s \psi(s, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi(s, \mathbf{x}, \Omega) + \sigma_t \psi(s, \mathbf{x}, \Omega) ds \\ = \lim_{\varepsilon \rightarrow 0} \int_{-\varepsilon}^{\infty} \delta(s) \int_0^{l(\mathbf{x}, \Omega)} \int_{\mathbb{S}^2} c \sigma_t s (\Omega' \cdot \Omega) \psi(s', \mathbf{x}, \Omega') d\Omega' ds' + \delta(s) Q(\mathbf{x}) ds. \end{aligned} \quad (4.10)$$

Since the classical angular (and scalar) flux is agnostic to s , we have $\psi(\mathbf{x}, \Omega) = \int_0^{\infty} \psi(s, \mathbf{x}, \Omega) ds$ ². Equation 4.10 is then equivalent to

$$\begin{aligned} \lim_{\varepsilon \rightarrow 0} \psi(s, \mathbf{x}, \Omega) \Big|_{s=-\varepsilon}^{s=\infty} + \Omega \cdot \nabla_{\mathbf{x}} \psi(\mathbf{x}, \Omega) + \sigma_t \psi(\mathbf{x}, \Omega) \\ = c \sigma_t \int_{\mathbb{S}^2} s (\Omega' \cdot \Omega) \psi(\mathbf{x}, \Omega') d\Omega' + Q(\mathbf{x}), \end{aligned} \quad (4.11)$$

which—because the first term vanishes—is exactly the classical, time-independent transport equation.

Expressing transport in terms of the distance to the next collisions, we can alternatively derive a *forward* formulation of the GLBE. Just like for the *backward* formulation, the phase space of the forward formulation is augmented by one additional variable when compared to the classical transport equation. To distinguish both formulations, we use $f(s, \mathbf{x}, \Omega)$ for the forward formulation, with s denoting the distance to the next interaction. Consequently, all definitions that were made at the beginning of this section still hold true for the forward formulation when *distance since an interaction* is replaced by *distance to an interaction*.

However, now that the memory is encoded differently, the transport equation, too, changes. Representing the distance to the next interaction, s has to decrease as particles move through space. Moreover, the distance to the next interaction has to be known *at the beginning* of a particle's trajectory, sampled from a probability density function. As a result, we know that all particles represented by the infinitesimal phase space element $f(s, \mathbf{x}, \Omega) ds d\mathbf{x} d\Omega$ are (with

²The non-classical angular flux outside the spatial domain is zero and we therefore write $\int_0^{\infty} \cdot ds$ instead of $\int_0^{l(\mathbf{x}, \Omega)} \cdot ds$.

absolute certainty) undergoing an interaction inside $[\mathbf{x} + s\Omega, (\mathbf{x} + d\mathbf{x}) + (s + ds)(\Omega + d\Omega)]$. Despite the precise knowledge of future interactions' locations, the forward formulation is of the same random nature as the backward formulation. Even though the location of an upcoming interaction is deterministic, the distance to the next interaction is sampled from a probability density function.

	the <i>backward</i> formulation	the <i>forward</i> formulation
additional variable	s , distance since interaction	s , distance to interaction
variable behavior	increases, initialized as $s = 0$	decreases, sampled from $p(s)$
deterministic component	value of s after collision known	next collision's location known
random component	collisions possible for any value of s	s is sampled randomly
(s, \mathbf{x}, Ω) implies boundary conditions	particles originate at $\mathbf{x} - s\Omega$ $\psi(s, \mathbf{x}, \Omega) = \delta(s) \cdot g(\mathbf{x}, \Omega)$	particles interact at $\mathbf{x} + s\Omega$ $f(s, \mathbf{x}, \Omega) = p(s) \cdot g(\mathbf{x}, \Omega)$

Table 4.1.: Juxtaposition of the forward and backward formulation.

The forward formulation of the generalized linear Boltzmann equation (omitting boundary contributions) is then given by

$$-\partial_s f(s, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} f(s, \mathbf{x}, \Omega) = p(s)c \int_{\mathbb{S}^2} s(\Omega' \cdot \Omega) f(0, \mathbf{x}, \Omega') d\Omega' + p(s)Q(\mathbf{x}). \quad (4.12)$$

To summarize the differences and similarities, Table 4.1 juxtaposes both formulations in a condensed form. We also include the respective boundary conditions, prescribed for all incoming angles at the domain's boundary, i.e., for all $\mathbf{x} \in \partial V$ and $\Omega \in \mathbb{S}^2$ where $\Omega \cdot \mathbf{n}(\mathbf{x}) < 0$.

Equivalence of both formulations has been proven in Larsen et al. [57], using the transformation

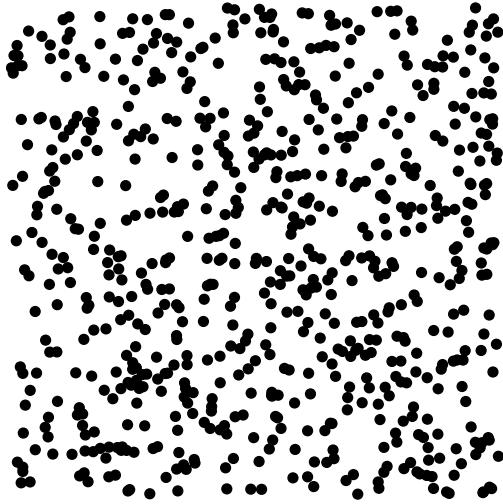
$$f(s, \mathbf{x}, \Omega) \equiv \int_0^{l(\mathbf{x}, \Omega)} \sigma_t(s + s') \psi(s + s', \mathbf{x} + s\Omega, \Omega) ds' \quad (4.13)$$

and the relation between cross section and path length distribution

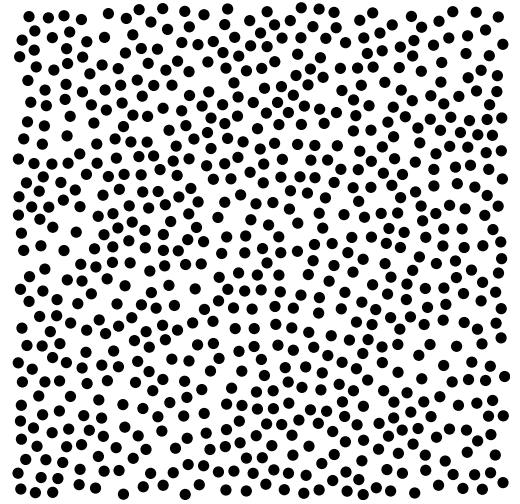
$$\sigma_t(s) = \frac{p(s)}{\int_s^\infty p(s') ds'} \quad \text{or} \quad p(s) = \sigma_t(s) \exp\left(-\int_0^s \sigma_t(s') ds'\right). \quad (4.14)$$

The forward formulation inserts randomness into the dynamics by sampling a particle's distance to the next interaction from $p(s)$; from that point on, the particle's trajectory is deterministic. Conversely, the backward formulation always resets the value of s after a collision and inserts randomness by allowing particles to scatter for any value of s , based on $\sigma_t(s)$. It is therefore naturally, to link $p(s)$ to the distance to the next interaction (i.e., the forward formulation), while $\sigma_t(s)$ refers to the distance since the last interaction (i.e., the backward formulation). Unless explicitly stated otherwise, we will from now on implicitly assume that s is to be interpreted in the forward or backward formulation, based on the choice of $p(s)$ or $\sigma_t(s)$, respectively.

We are not going to recapitulate the thorough analysis of Larsen et al. [57] that proofs equivalence here, but rather try to provide a physical interpretation of the involved terms, demonstrating that both sides of (4.13) indeed express the same physical quantity. The left-hand side is easy to understand: $f(s, \mathbf{x}, \Omega)$ quantifies the amount of particles that pass through (\mathbf{x}, Ω) to undergo a collision at $\mathbf{x} + s\Omega$. Next, $\psi(s + s', \mathbf{x} + s\Omega, \Omega)$ represents the particles at $(\mathbf{x} + s\Omega, \Omega)$ that have traveled a distance $s + s'$ from $\mathbf{x} - s\Omega$ to reach this point, passing through (\mathbf{x}, Ω) along that journey. Multiplied by $\sigma_t(s + s')$, we obtain the fraction of those particles that do undergo a collision at $\mathbf{x} + s\Omega$. Integrating over all distance s' from \mathbf{x} back to the domain boundary at $\mathbf{x} - l(\mathbf{x}, \Omega)\Omega$, we obtain *all* particles that travel through (\mathbf{x}, Ω) and collide at $\mathbf{x} + s\Omega$, i.e., the same quantity as the left-hand side.



(a) Scatterers generated in a uniformly random manner. Scatterers are allowed to overlap.



(b) Scatterers generated via Poisson disc sampling where the minimal distance between two boundaries is equal to the radius r .

Figure 4.6.: Scatterers of radius $r = 0.01$ generated in the domain $[0, 1] \times [0, 1]$. Both pictures show 696 scatterers.

4.3. Correlated and uncorrelated particles

A thought (and numerical) experiment made by d’Eon [33] that was independently considered in Jarabo et al. [48] made the following observation: For non-classical particle transport it is necessary to distinguish particles that are *correlated* with the medium (related quantities are denoted with a subscript c) from those particles that are *uncorrelated* with the medium (denoted with a subscript u). This distinction is explained by considering particle transport in an obstacle field generated via a Poisson disc sampling procedure. Rather than placing scatterers randomly—as it is the assumption in classical transport theory—we place scatterers in a way such that they obey a certain minimal distance from any other scatterer, visualized in Figures 4.6a and 4.6b, respectively. The distributions for the *distance between consecutive collisions* has to be zero for distances smaller than the minimal distance between scatterers. However, for a particle that is placed independently of the scatters—e.g., by a source—the distribution for the *distance to the next collision* does not demand this property.

Consequently, we distinguish correlated from uncorrelated particles. These two species then behave as follows:

- 1) We gain correlated (or uncorrelated) particles by a correlated (or uncorrelated) source.
- 2) If correlated (or uncorrelated) particles get absorbed we lose correlated (or uncorrelated) particles.
- 3) If correlated particles scatter they stay correlated.
- 4) However, if uncorrelated particles scatter, they are no longer independent of the medium and become correlated particles.

Ignoring heterogeneity, this life cycle is depicted in Figure 4.7. We are aware that the above distinction is similar to the one of *collided* and *uncollided* particles. In fact, for the homogeneous case, uncorrelated particles are uncollided ones and correlated particles are collided ones. For

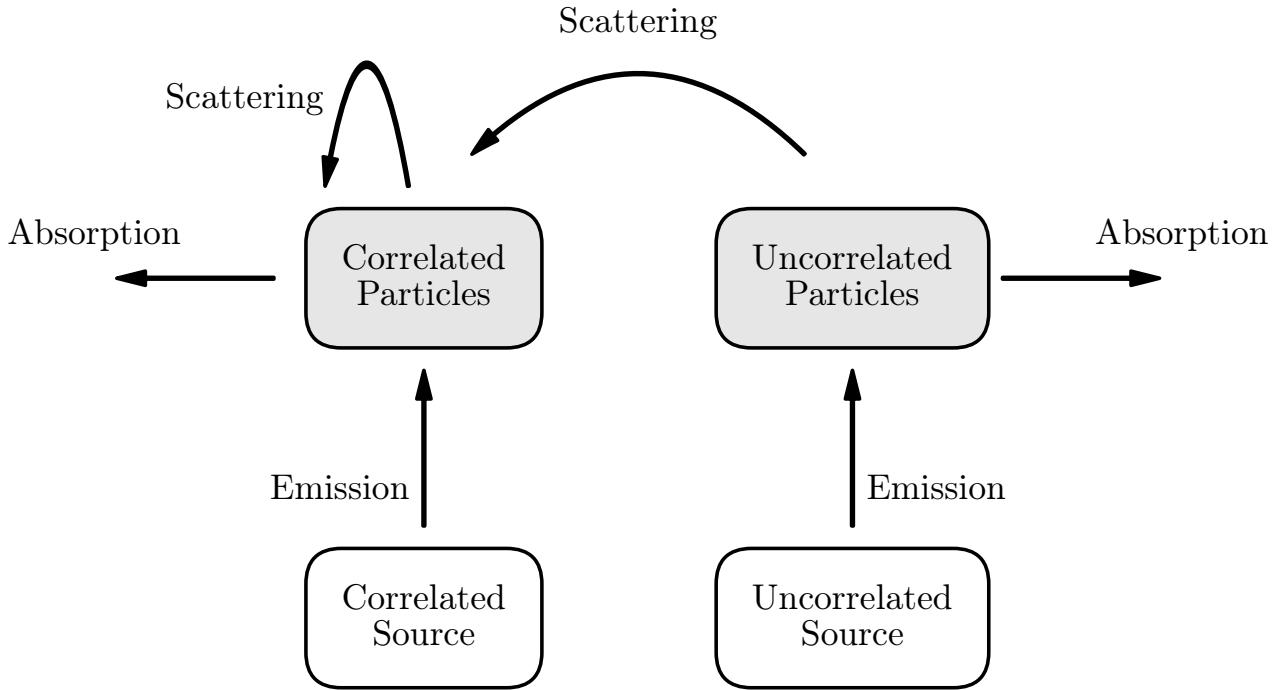


Figure 4.7.: Life cycle of correlated and uncorrelated particles.

the heterogeneous case, however, correlated particles can become uncorrelated, whereas collided particles always stay collided. We make the following definitions:

$N_u(s, \mathbf{x}, \Omega) ds d\mathbf{x} d\Omega$ = “the number of uncorrelated particles in $ds d\mathbf{x} d\Omega$ about (s, \mathbf{x}, Ω) .”

$\psi_u(s, \mathbf{x}, \Omega) = v N_u(s, \mathbf{x}, \Omega)$ = “the non-classical, uncorrelated angular flux.”

$\sigma_{u,t}(s) ds$ = “the probability that an uncorrelated particle which has traveled a distance s since a previous event experiences a collision while traveling a further distance ds .”

$Q_u(\mathbf{x}) d\mathbf{x}$ = “the rate at which uncorrelated particles are isotropically emitted by an internal source in $d\mathbf{x}$ about \mathbf{x} .”

Similarly, we obtain the quantities for correlated particles.

$N_c(s, \mathbf{x}, \Omega) ds d\mathbf{x} d\Omega$ = “the number of correlated particles in $ds d\mathbf{x} d\Omega$ about (s, \mathbf{x}, Ω) .”

$\psi_c(s, \mathbf{x}, \Omega) = v N_c(s, \mathbf{x}, \Omega)$ = “the non-classical, correlated angular flux.”

$\sigma_{c,t}(s) ds$ = “the probability that a correlated particle which has traveled a distance s since a previous event experiences a collision while traveling a further distance ds .”

$Q_c(\mathbf{x}) d\mathbf{x}$ = “the rate at which correlated particles are isotropically emitted by an internal source in $d\mathbf{x}$ about \mathbf{x} .”

Supplementing the definitions above, we remark that (i) these equations describe homogeneous transport, (ii) we only choose isotropic sources to shorten notation, and (iii) we subsequently assume the absence of sources that are correlated with the medium on a microscopic level.

Following Larsen and Vasques [56], we obtain versions of the generalized linear Boltzmann equation, provided first in a similar form by d'Eon [33]. They are given by

$$\partial_s \psi_u(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_u(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_{t,u}(s) \psi_u(s, \mathbf{x}, \boldsymbol{\Omega}) = \delta(s) Q_u(\mathbf{x}) \quad (4.15)$$

for the uncorrelated angular flux and

$$\begin{aligned} \partial_s \psi_c(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_c(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_{t,c}(s) \psi_c(s, \mathbf{x}, \boldsymbol{\Omega}) \\ = c \delta(s) \int_0^{l(\mathbf{x}, \boldsymbol{\Omega})} \int_{\mathbb{S}^2} s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) (\sigma_{t,c}(s') \psi_c(s', \mathbf{x}, \boldsymbol{\Omega}') + \sigma_{t,u}(s') \psi_u(s', \mathbf{x}, \boldsymbol{\Omega}')) d\boldsymbol{\Omega}' ds' \end{aligned} \quad (4.16)$$

for the correlated angular flux, respectively. With $l(\mathbf{x}, \boldsymbol{\Omega})$ we denote the distance from \mathbf{x} to ∂V moving backward with direction $-\boldsymbol{\Omega}$. Boundary conditions are given by

$$\psi_c(0, \mathbf{x}, \boldsymbol{\Omega}) = 0 \quad \text{for } \mathbf{x} \in \partial V, \mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\Omega} < 0, \quad (4.17a)$$

$$\psi_u(s, \mathbf{x}, \boldsymbol{\Omega}) = \delta(s) \psi_u^{bc}(\mathbf{x}, \boldsymbol{\Omega}) \quad \text{for } \mathbf{x} \in \partial V, \mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\Omega} < 0, \quad (4.17b)$$

with a prescribed function $\psi_u^{bc}(\mathbf{x}, \boldsymbol{\Omega})$. This means that particles entering the domain cannot be correlated since, by definition, they have not yet interacted with the obstacles. There is only a flux of uncorrelated particles entering the domain with $s = 0$.

Since (4.15) can be solved independently of (4.16), it is possible to solve (4.15) for $\psi_u(s, \mathbf{x}, \boldsymbol{\Omega})$ first and rewrite (4.16) as

$$\begin{aligned} \partial_s \psi_c(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_c(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_{t,c}(s) \psi_c(s, \mathbf{x}, \boldsymbol{\Omega}) \\ = c \delta(s) \int_0^{l(\mathbf{x}, \boldsymbol{\Omega})} \int_{\mathbb{S}^2} s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \sigma_{t,c}(s') \psi_c(s', \mathbf{x}, \boldsymbol{\Omega}') d\boldsymbol{\Omega}' ds' + Q_c(s, \mathbf{x}, \boldsymbol{\Omega}), \end{aligned} \quad (4.18)$$

with

$$Q_c(s, \mathbf{x}, \boldsymbol{\Omega}) = c \delta(s) \int_0^{l(\mathbf{x}, \boldsymbol{\Omega})} \int_{\mathbb{S}^2} s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \sigma_{t,u}(s') \psi_u(s', \mathbf{x}, \boldsymbol{\Omega}') d\boldsymbol{\Omega}' ds'. \quad (4.19)$$

In the case of classical transport, i.e., $\sigma_{t,c}(s) = \sigma_{t,u}(s) = \sigma_t$ independent of s , the equations reduce to the classical linear transport equation.

Additionally, the necessity for splitting particles into correlated and uncorrelated ones vanishes if $\sigma_{t,c}(s) = \sigma_{t,u}(s) = \sigma_t(s)$, resulting in the generalized linear Boltzmann equation derived by Larsen and Vasques [56] with the non-classical angular flux satisfying $\psi(s, \mathbf{x}, \boldsymbol{\Omega}) = \psi_u(s, \mathbf{x}, \boldsymbol{\Omega}) + \psi_c(s, \mathbf{x}, \boldsymbol{\Omega})$.

Jarabo et al. [48] and d'Eon [33] omitted the consideration of heterogeneous non-classical transport. To extend their analysis to the heterogeneous case, we first need to introduce the concept of heterogeneity with all its subtleties. Correlated and uncorrelated transport in heterogeneous materials is then reconsidered in Section 5.6.

5. Non-classical transport in heterogeneous materials

What happens to a particle's memory when it crosses a material interface?

This is the main question that Chapter 5 attempts to answer. Easy to pose, we will see that the respective answer fundamentally depends upon the underlying assumptions that are made to model heterogeneous materials. We will discuss why this is a relevant, non-trivial question in the context of non-classical particle transport and encounter a fundamental problem of mathematical modeling: Lacking real-world measurements or experimental data, we are limited to educated guesses and bona fide speculations. And while these are *falsifiable* guesses and speculations, it is beyond the scope of this work to actually execute the necessary experiments or measurements to either falsify or verify the subsequent models.

Note, again, that the problem of modeling transport in the presence of heterogeneities is exclusively limited to non-classical particle transport where the governing equations have to encode a particle's memory. As already mentioned in Chapter 4, classical transport is memoryless in the sense that

$$\Pr(S > s + \Delta s | S > s) = \Pr(S > \Delta s).$$

Consequently, the path length distribution for a particle that faces an interface a distance $s^* \geq 0$ away is simply

$$p(s) = \begin{cases} \sigma_1 e^{-\sigma_1 s} & \text{if } s \leq s^*, \\ \sigma_2 e^{-\sigma_1 s^* - \sigma_2(s-s^*)} & \text{if } s > s^*, \end{cases} \quad (5.1)$$

where σ_1 and σ_2 denote the cross sections in front of and behind the interface, respectively.

With non-classical transport dating back to the early 2000s (approximately), heterogeneous non-classical transport has only recently become a topic of greater interest [20, 48, 14] and an equivalent formulation of (5.1) is not proven.

5.1. The crux of particles crossing material interfaces

Using the backward formulation of the generalized linear Boltzmann equation, the total cross section depends upon the distance that a particle has already traveled. For simplicity, we consider the situation depicted in Figure 5.1, where there is only one interface and a particle, moving from left to right across that interface. Considered individually, the cross sections on the left and right side of the interface are $\sigma_1(s)$ and $\sigma_2(s)$, respectively. The key point here, is the word *individually*, because it implies that we are able to model the heterogeneity by two cross sections that are the result of an individual, homogeneous consideration.

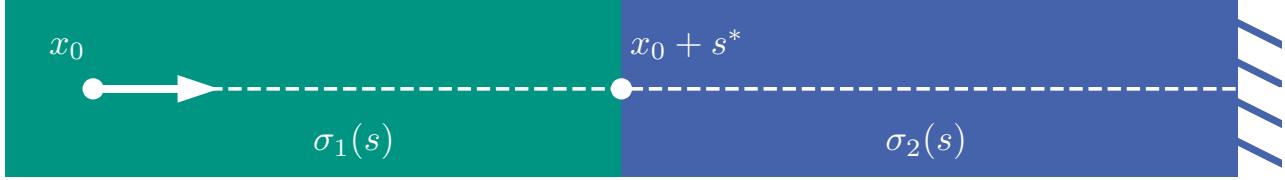


Figure 5.1.: Two materials with cross sections $\sigma_1(s)$ and $\sigma_2(s)$, and a particle traversing from one side of the interface to the other. The material extends to the right indefinitely.

If we for now assume that this is a valid model, several requirements seem reasonable [20]:

1. The resulting path length distribution should be a probability density function in the sense that it integrates to unity.
2. If both cross sections are independent of s , we want to recover (5.1).
3. Up to the interface, the path length distribution should only depend on $\sigma_1(s)$.

The first point is merely a technicality. The second point follows from the fact that the non-classical transport equation reduces to the classical transport equation if the cross sections are independent of the distance that a particle has traveled. It can also be used as a way to check any proposed candidate for the path length distribution for the non-classical, heterogeneous case. Lastly, since we consider the *backward* formulation, the path length distribution should be agnostic to future materials and obey the third point.

Now, if the cross sections of non-classical materials depend on s , and a particle crosses an interface after having traveled a distance s^* to reach a material with $\sigma_2(s)$, should it experience $\sigma_2(s^* + \Delta s)$ or $\sigma_2(\Delta s)$ when traveling another distance Δs in the new material? Per se, there is no compelling reason to evaluate the cross section of the new material at $s^* + \Delta s$ rather than at Δs . Arguing for the former implies that it is possible to relate the distance traveled in *one* material to the likelihood of scattering in *another* material; even when we allow to choose the materials completely independent from each other.

Things become more complicated when we consider one further requirement:

4. If $\sigma_1(s) \equiv \sigma_2(s)$, we want to recover the homogeneous case.

This seems like a reasonable prerequisite since, in the case of $\sigma_1(s) \equiv \sigma_2(s)$, the interface is only artificial and the material is therefore homogeneous. As a result, the cross section should be evaluated at $s^* + \Delta s$ rather than at Δs since the latter implies a memory reset for which there is no reason since the material is homogeneous.

Summarizing, we end up with the following dilemma: For two materials with arbitrarily chosen cross sections $\sigma_1(s)$ and $\sigma_2(s)$, the memory should be reset when crossing the interface. If, however, the two cross sections are identical, the memory should persist.

We proposed an expression for the path length distribution that solves the dilemma in [20]. However, as we will see in Section 5.2, there are arguments against the fourth prerequisite, ultimately rendering the memory loss inevitable. The corresponding path length distribution is analyzed in Section 5.3.

5.2. Defining heterogeneity

To discuss the definition of heterogeneity, we will repeatedly refer to the two situations that are depicted in Figure 5.2. In Figure 5.2a, we see (in the upper half) a homogeneous non-classical material from a microscopic point of view. The arrangement of obstacles is that of the periodic Lorentz gas.¹ Performing the Boltzmann-Grad limit (i.e., $n \rightarrow \infty$ such that $n \cdot r = \text{constant}$ and ensemble averaging over all possible arrangements), we end up with a material that is (on the mesoscopic level) defined by its cross section and path length distribution.

In the heterogeneous case, illustrated in Figure 5.2b, the two materials are identical in the sense that obstacles are arranged in a lattice structure. But—and this is shown for the green and blue scatterers in Figure 5.2b—it is entirely possible that these two lattices do not align perfectly. In general, we observe two arrangements of obstacles which, when considered individually, are valid arrangements of scatterers, but do not resemble the homogeneous case. If we now perform the Boltzmann-Grad limit individually, both of the materials are defined (on the mesoscopic level) by the same cross section and path length distribution. But since we ensemble average both sides independently, a particle's memory should be reset when crossing the interface. Ultimately, we are left with a situation that looks homogeneous on the mesoscopic level, but does not equal the homogeneous material on the microscopic level.

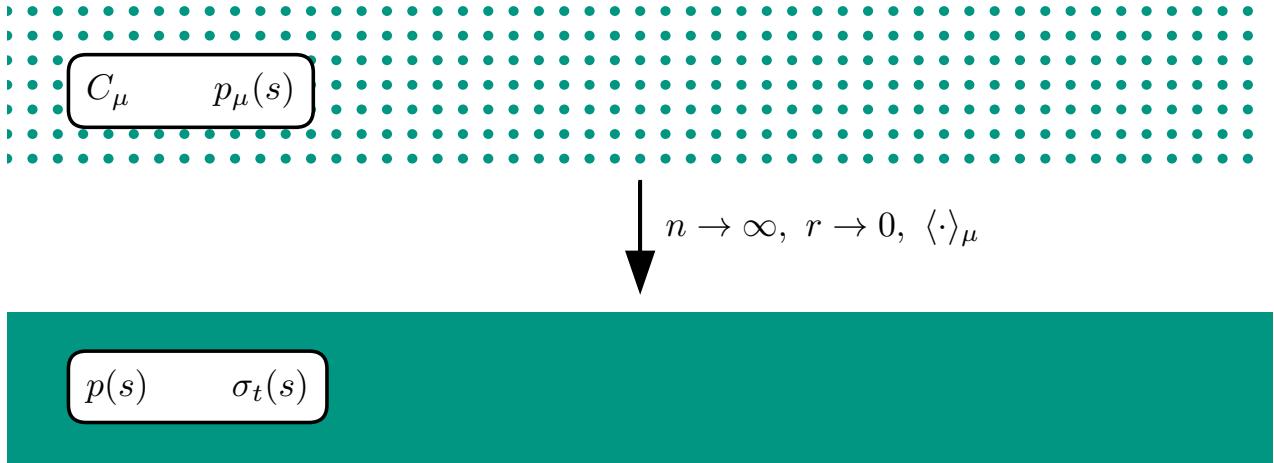
A situation that exemplifies these theoretical considerations is that of two cubes being pushed together. With their crystalline, Lorentz-like structure, both cubes have the same, correlated ordering of atoms. At the interface, however, this two-cube-setup will look different from a single quadrilateral of the same dimensions. If we now place a light source such that photons move through the setup from one cube to the next, we have reproduced the conditions that we described in Figure 5.2b.

Presented first in Camminady et al. [20], a simplified, less general scenario was considered that nevertheless contributes to the discussion of heterogeneous materials. Depicted in Figure 5.3, the two scenarios distinguish heterogeneities that are due to density fluctuations (Figure 5.3a) and those that formed by combining structurally different obstacle arrangements (Figure 5.3b). The ansatz for the first scenario that was presented in Camminady et al. [20] will proceed the more general, fully heterogeneous case in our discussion later on.

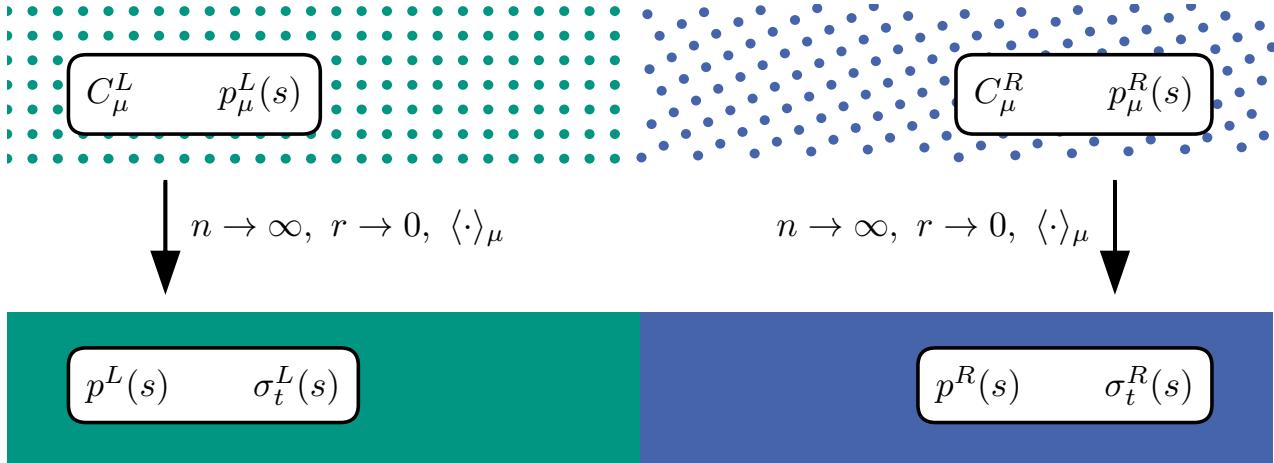
By now it is clear that the definition of heterogeneity depends on our perspective. Whether a heterogeneous domain with cross sections $\sigma_1(s)$ and $\sigma_2(s)$ turns homogeneous if $\sigma_1(s) \equiv \sigma_2(s)$ depends on the circumstances. If we define a material solely by its cross sections, it should be homogeneous. If we instead consider the underlying microscopic situation, an interface remains and a path length distribution has to account for it.

Given these circumstances, we will provide two different types of approaches to express the distribution of path lengths in presence of (potentially) different cross sections. One formulation preserves the particle's memory when crossing an untrue ($\sigma_1(s) \equiv \sigma_2(s)$) interface and another formulation where this is not the case. Both formulations, however, will satisfy the first three requirements that we imposed.

¹We only chose the periodic Lorentz gas as one instance of all possible non-classical materials because it is immediately clear that the depicted material is non-classical. The analysis, however, is not limited to the periodic Lorentz gas.

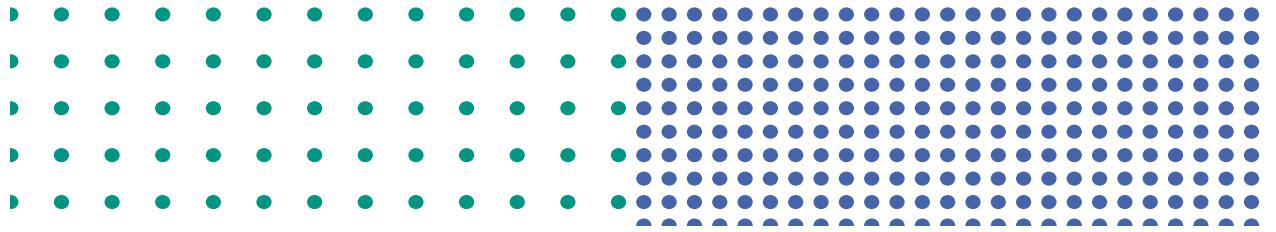


(a) Ensemble averaging the whole domain yields a homogeneous material, defined by $p(s)$ and $\sigma_t(s)$.

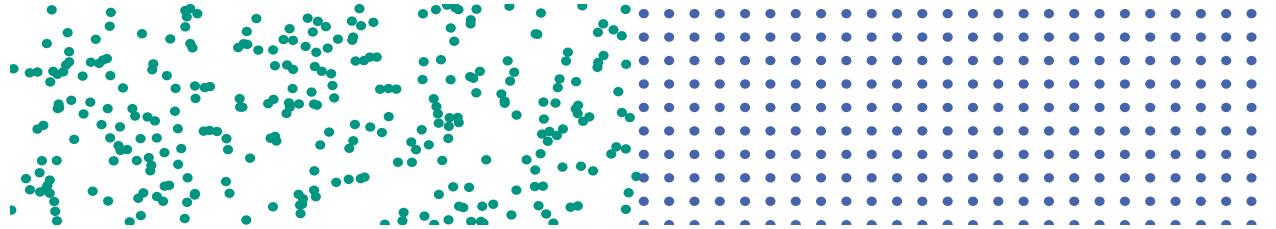


(b) If both halves are ensemble averaged independently, they should still yield the same distribution and cross section when considered individually. When looking at the whole domain, however, it is not clear that this way of executing the Boltzmann-Grad limit has to yield the same result for the whole domain as found in Figure 5.2a.

Figure 5.2.: The path length distribution and cross section are the outcome of the Boltzmann-Grad limit. This involves an ensemble average over all possible configurations of obstacles C_μ , even for the periodic Lorentz gas. For example, obstacle positions could be translated or rotated. This holds true for other arrangements as well, e.g., Poisson disk sampling, where obstacles are at least a prescribed minimum distance apart from another. The subscript μ denotes a singular instance out of all possible arrangements and $\langle \cdot \rangle_\mu$ is the respective ensemble average.



(a) A heterogeneous material that differs only in density as the underlying structure of obstacles is identical.



(b) A heterogeneous material that comprises two fundamentally different types of obstacle arrangements.

Figure 5.3.: Material heterogeneities can be the result of density fluctuations (Figure 5.3a), or structural differences (Figure 5.3b).

5.3. Memory-resetting ansatz

Arguably the simpler—and maybe less exciting—case, we will start by considering the scenario in which the memory of a particle gets reset at an interface.

Consider a particle positioned at x_0 inside Material 1 which is defined by $\sigma_1(s)$ and $p_1(s)$. For a fixed direction of flight, the particle crosses the interface between Material 1 and Material 2 at $x_0 + s_1$. More generally, after traveling a distance s_i , the particle is crossing the interface between Material $i - 1$ and Material i , where the s_i are prescribed by the arrangement of materials; additionally, we define $s_0 = 0$. This situation is visualized for the direction $\Omega = (1, 0)^T$ in Figure 5.4. The individual cross sections and path length distributions for a material describe a particle’s dynamics if it were to be placed inside an infinite, homogeneous domain of exactly that material.

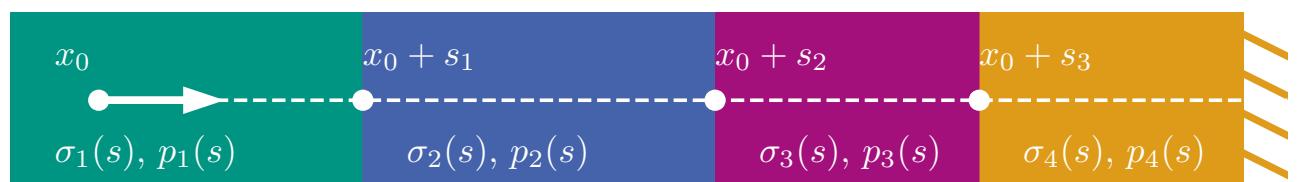


Figure 5.4.: A particle traversing multiple materials with different cross sections $\sigma_i(s)$ and related density functions $p_i(s)$ with interfaces located distances s_i away from its origin. The cross sections and density functions are to be interpreted in the homogeneous sense, i.e., if the particle would be in a homogeneous, infinite domain of Material i , it would experience cross section $\sigma_i(s)$ and density function $p_i(s)$.

A particle's memory is reset by either resampling the distance to the next interaction after the crossing of an interface, or by resetting the value of s that is used to evaluate the probability to scatter via $\sigma_i(s)$.

Thus, the probability for a particle that has traveled a distance s from x_0 to reach $x_0 + s$ to scatter while traveling another distance ds is

$$\sigma(s)ds = \sigma_{\chi(x_0+s)}(s - s_{\chi(x_0+s)-1})ds, \quad (5.2)$$

where the indicator function $\chi(x)$ returns the index of the material at position x . Likewise, we can express the probability that a particle at x_0 will undergo its next collision after traveling a distance s to reach $x_0 + s$ via

$$p(s) = \begin{cases} p_1(s) & \text{if } s \in [0, s_1), \\ (1 - \int_0^{s_1} p(s') ds') p_2(s - s_1) & \text{if } s \in [s_1, s_2), \\ (1 - \int_0^{s_2} p(s') ds') p_3(s - s_2) & \text{if } s \in [s_2, s_3), \\ (1 - \int_0^{s_3} p(s') ds') p_4(s - s_3) & \text{if } s \in [s_3, s_4), \\ \dots & \end{cases} \quad (5.3)$$

The factors involving an integral in every line but the first renormalize the probabilities to take into account that particles might have scattered in an earlier interval. Domain-wise, the relation (4.14) holds. That this is also true for (5.2) and (5.3) demonstrates the following analysis. First, we compute the respective cumulative distribution function for (5.3). Assuming $s \in [s_{i-1}, s_i)$, this becomes

$$P(s) := \int_0^s p(s') ds' \quad (5.4a)$$

$$= P(s_{i-1}) + \int_{s_{i-1}}^s p(s') ds' \quad (5.4b)$$

$$= P(s_{i-1}) + \int_{s_{i-1}}^s (1 - P(s_{i-1})) p_i(s' - s_{i-1}) ds' \quad (5.4c)$$

$$= P(s_{i-1}) + (1 - P(s_{i-1})) P_i(s - s_{i-1}). \quad (5.4d)$$

We know from (4.14) that $\sigma(s) = p(s) / \int_s^\infty p(s') ds'$. In our case, for $s \in [s_{i-1}, s_i)$, this becomes

$$\sigma(s) = \frac{(1 - P(s_{i-1})) p_i(s - s_{i-1})}{1 - [P(s_{i-1}) + (1 - P(s_{i-1})) P_i(s - s_{i-1})]} \quad (5.5a)$$

$$= \frac{p_i(s - s_{i-1})}{1 - P_i(s - s_{i-1})} \quad (5.5b)$$

$$= \sigma_i(s - s_{i-1}), \quad (5.5c)$$

which is exactly the relation from (5.3). Consequently, (5.2) and (5.3) are equivalent ways to describe the same behavior.

By design, this formulation does not reproduce the non-classical, homogeneous formulation if the cross sections or distribution functions are identical. As an example, Figure 5.5 shows different materials which are explained in Table 5.1. Here, the Pareto distribution has both a finite mean and a finite variance for $\alpha \geq 2$ which is why we choose $\alpha = 3$ in this case. In Figure 5.6, we have two domains, both consistent of the same material; path lengths are gamma

i	$p_i(s)$	$\sigma_i(s)$	Name
1	e^{-s}	1	Exponential
2	$2e^{-2s}$	2	Exponential
3	se^{-s}	$\frac{s}{s+1}$	Gamma
4	$\begin{cases} 0 & \text{if } s \leq s_0, \\ \alpha \frac{s_0^\alpha}{s^{\alpha+1}} & \text{if } s > s_0 \end{cases}$	$\begin{cases} 0 & \text{if } s \leq s_0, \\ \frac{\alpha}{s} & \text{if } s > s_0 \end{cases}$	Pareto ($\alpha = 3, s_0 = 1$)

Table 5.1.: Cross sections and distribution functions for the materials in Figure 5.5.

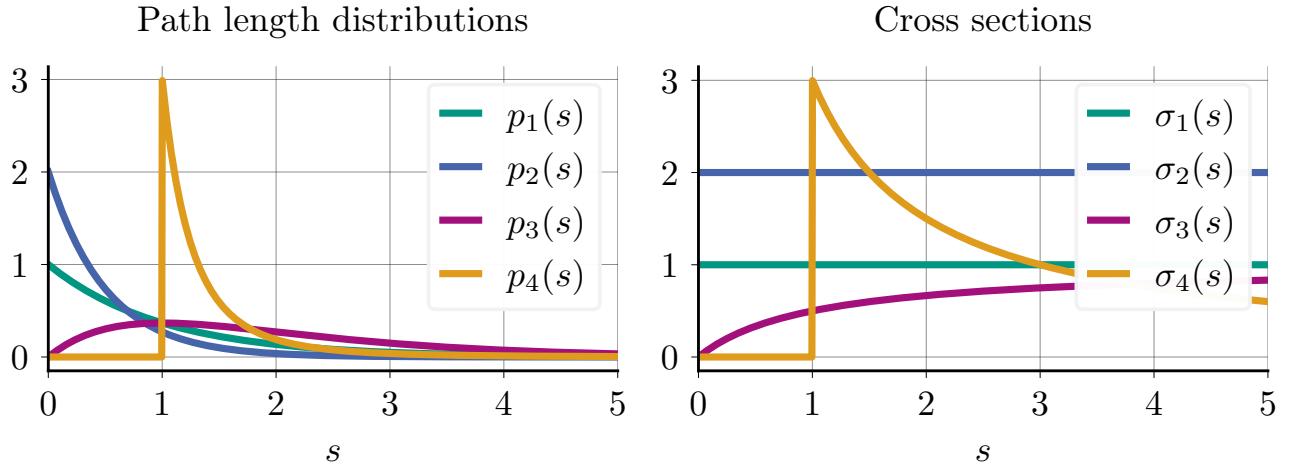


Figure 5.5.: Different materials defined by their density functions and cross sections. Here, p_1 and p_2 are exponential distributions, p_3 is a gamma distribution, and p_4 is a Pareto distribution. Cross sections are computed with (4.14).

distributed. This is also indicated by the color coding; purple corresponds to $p_3(s)$ in Figure 5.5. The combined path length distribution is then made up of two parts: For $s \leq 2$, particles only experience the first material and the combined path length distribution $p(s)$ is identical to the gamma distribution. However, when particles cross the interface at $s = 2$, their memory is reset. The probability to further travel a certain distance, given that the particle has already traveled a distance $s = 2$ is re-sampled from the path length distribution of the new domain. Hence, the combined (unconditional) distribution has a discontinuity at $s = 2$. Nevertheless, $p(s)$ still integrates to unity (as required for any probability density function).

If we had chosen both materials as classical materials, the combined distribution would be equivalent to the distribution for the homogeneous case. For non-classical materials, however, this is not the case.

Lastly, we consider a more elaborate scenario in Figure 5.7 with two interfaces and three distinctively different materials. The domains' colors again correspond to the materials presented in Figure 5.5. As before, the path length distribution results from the application of (5.3) and is visualized in Figure 5.7a. Similarly, we can use (5.2) to compute the resulting s -dependent cross section, shown in Figure 5.7b. At every interface, the particle's distance to the next collision is resampled from the respective distribution and the cross section ignores the distance traveled up to the interface, i.e., the memory is reset.

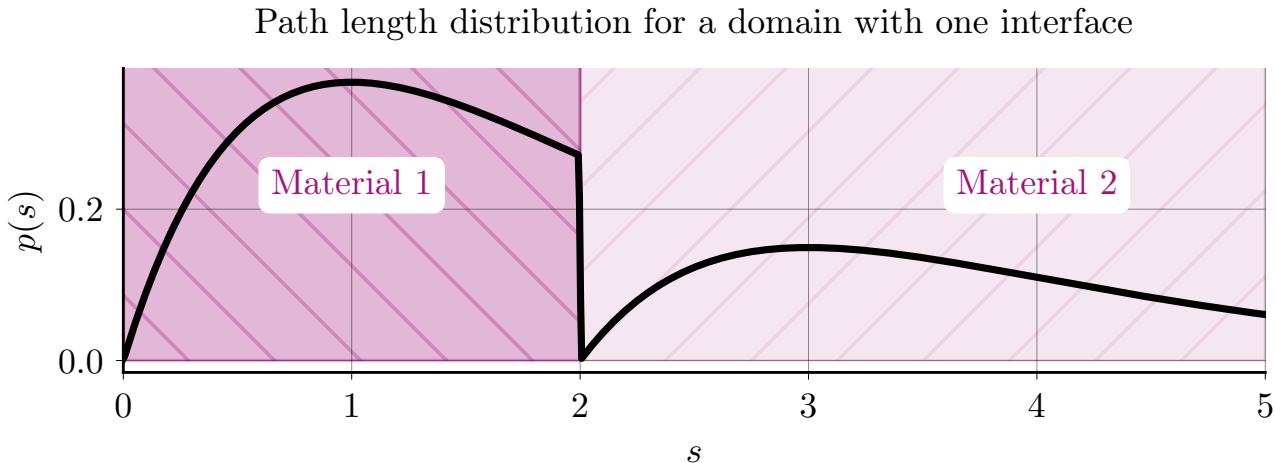
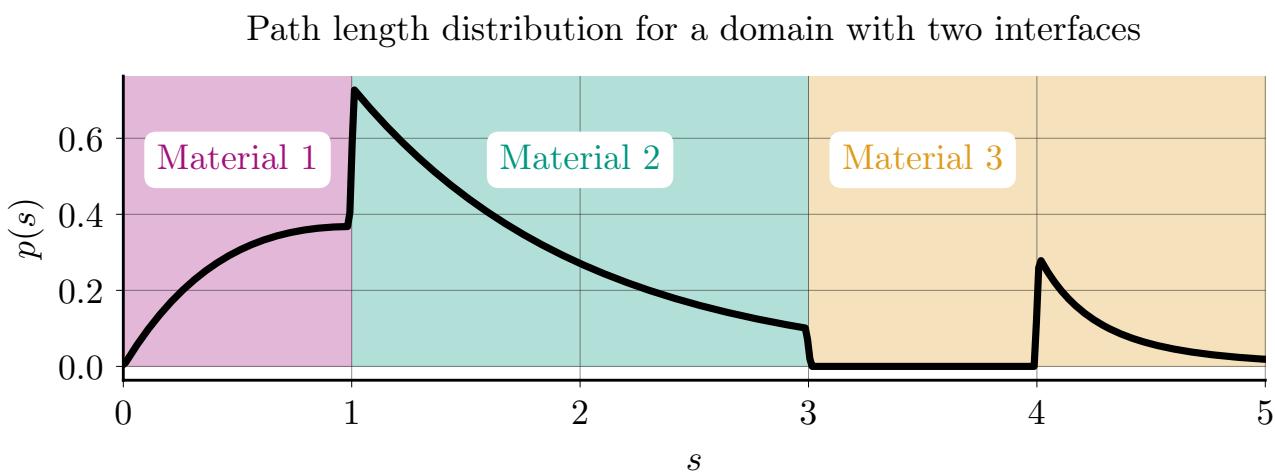
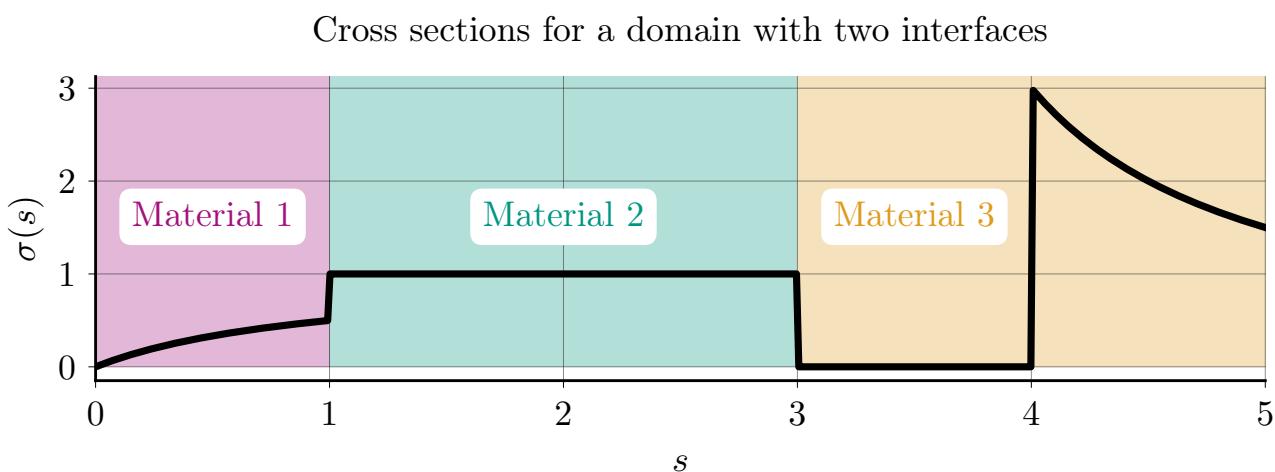


Figure 5.6.: Combined path length distribution for a domain with two materials that are identical; both have gamma distributed path lengths.



(a) Combined path length distribution.



(b) Combined cross section.

Figure 5.7.: Distribution and cross section for a domain composed of three materials: Material 1 is defined by a gamma distribution, Material 2 by an exponential distribution, and Material 3 by a Pareto distribution.

Name	General definition $p_\rho(s)$	Base density $p_1(s)$	Conversion $\rho \cdot (p_1(\rho \cdot s))$
Exponential	$\rho e^{-\rho s}$	e^{-s}	$\rho \cdot (e^{-\rho s})$
Gamma	$\rho^2 s e^{-\rho s}$	$s e^{-s}$	$\rho \cdot (\rho s e^{-\rho s})$
Pareto	$\begin{cases} 0 & \text{if } s \leq 1/\rho, \\ \alpha \frac{(1/\rho)^\alpha}{s^{\alpha+1}} & \text{if } s > 1/\rho \end{cases}$	$\begin{cases} 0 & \text{if } s \leq 1, \\ \alpha \frac{1}{s^{\alpha+1}} & \text{if } s > 1 \end{cases}$	$\rho \cdot \begin{cases} 0 & \text{if } \rho s \leq 1, \\ \alpha \frac{1}{s^{\alpha+1} \rho^{\alpha+1}} & \text{if } \rho s > 1 \end{cases}$

Table 5.2.: Requirement (ii) is satisfied for all probability density functions, showcased for those functions that we have already encountered. For each one, we observe that the general definition equals the conversion. The \mathbf{x} -dependency of ρ was only omitted to shorten notation.

5.4. Memory-preserving ansatz

Next, we are discussing memory-preserving ansätze. We start with the simpler scenario that was initially considered in Camminady et al. [20] and in which heterogeneity is the result of density fluctuations of a single type of material.

5.4.1. Heterogeneity via density fluctuations

In this scenario, the fundamental idea is similar to the conversion between the geometric and optical path length in optics. Light that travels a distance s through a medium has a geometric path length of s . The optical path length, however, takes the refractive index n of that material into account and is computed as the product of s and n . Then, to sample the optical path length, we only need to sample values from the geometric path length and divide them by n .

We make the following two assumptions for the conversion to work in our application. (i) We are equipped with a *base* density function for the distance to the next collision, called $p_1(s)$. (ii) If the material found at \mathbf{x} were to cover the full, infinite, homogeneous domain, its density function could be written as $p_{\rho(\mathbf{x})}(s) := \rho(\mathbf{x}) p_1(\rho(\mathbf{x}) \cdot s)$ for some function $\rho(\mathbf{x})$ that encodes the domain's density fluctuations. Note that (ii) is a reasonable assumption for all probability density functions, shown in Table 5.2 for the distributions we have encountered so far. Moreover, with $P_{\rho(\mathbf{x})}(s)$ and $P_1(s)$ as the cumulative distribution functions, $P_{\rho(\mathbf{x})}(s) = P_1(\rho(\mathbf{x}) \cdot s)$ holds. It is important to emphasize again that this is not the distribution for the full heterogeneous case, but only describes the geometric-to-optical path length translation if the whole domain were to be filled with the material found at \mathbf{x} . However, the structure of this expression allows us to make an ad hoc ansatz for the distribution of a heterogeneous domain—where the heterogeneity is still the result of density fluctuations.

Conjecture 5.4.1 (Path length distribution for the heterogeneous case with density fluctuations). *Assume that a particle underwent its last collision at \mathbf{x} , now facing direction $\mathbf{\Omega}$. Considered individually in a homogeneous setting, the material at \mathbf{x} has a distribution for the distance to the next collision $p_{\rho(\mathbf{x})}(s)$ that satisfies $p_{\rho(\mathbf{x})}(s) = \rho(\mathbf{x}) p_1(\rho(\mathbf{x}) \cdot s)$ for a known $p_1(s)$ and $\rho(\mathbf{x})$ that describe a base distribution and density fluctuations, respectively. Then, the probability for that particle to travel a distance smaller than or equal to s until its next collision is given by*

$$P(s, \mathbf{x}, \mathbf{\Omega}) = P_1 \left(\int_0^s \rho(\mathbf{x} + s' \cdot \mathbf{\Omega}) ds' \right). \quad (5.6)$$

Consequently, the respective density function satisfies

$$p(s, \mathbf{x}, \boldsymbol{\Omega}) = \rho(\mathbf{x} + s \cdot \boldsymbol{\Omega}) p_1 \left(\int_0^s \rho(\mathbf{x} + s' \cdot \boldsymbol{\Omega}) ds' \right) \quad (5.7)$$

and it follows that the cross section satisfies

$$\sigma(s, \mathbf{x}, \boldsymbol{\Omega}) = \frac{\rho(\mathbf{x} + s \cdot \boldsymbol{\Omega}) p_1 \left(\int_0^s \rho(\mathbf{x} + s' \cdot \boldsymbol{\Omega}) ds' \right)}{1 - P_1 \left(\int_0^s \rho(\mathbf{x} + s' \cdot \boldsymbol{\Omega}) ds' \right)}. \quad (5.8)$$

It is eminent that (5.6) and (5.7) reduce to the known results in the case of (i) a classical, but heterogeneous material, i.e., $p_1(s) = e^{-s}$, and (ii) a non-classical, but homogeneous material, i.e., $\rho(\mathbf{x} + s \cdot \boldsymbol{\Omega}) = \sigma$.

In Figure 5.8, we see the expression (5.7) for different densities $\rho_1(s) = (1 \text{ if } s \leq 2 \text{ else } 2)$, $\rho_2(s) = (2 \text{ if } s \leq 2 \text{ else } 1)$, and $\rho_3(s) \equiv 1.5$. The base distribution is the gamma distribution (Figure 5.8a) or the Pareto distribution (Figure 5.8b). For $\rho_3(s)$, the respective path length distribution equals the homogeneous one since we preserve the particles' memory.

5.4.2. True heterogeneity

The obvious drawback of the previous approach is that its application is limited to materials whose heterogeneity has to be explainable by density fluctuations only. Domains composed of *structurally* different materials are not governed by (5.7).

To overcome this limitation, we need to be able to stitch together arbitrary path length distributions. This kind of *surgery* needs to be done carefully to both preserve a particle's memory and ensure that the resulting path length distribution still integrates to unity. The approach presented here ensures this by stitching together the materials' cumulative distribution functions in a way that ensures continuity of the overall, fully heterogeneous cumulative distribution. This is done in the following way: A particle that has already traveled a certain distance s' is bound to enter a new material at $\mathbf{x} + s' \cdot \boldsymbol{\Omega}$. If we know the value of the heterogeneous cumulative distribution function $P(s, \mathbf{x}, \boldsymbol{\Omega})$ up to the point $s = s'$, we can try to find t' such that $P(s', \mathbf{x}, \boldsymbol{\Omega}) = P_{\mathbf{x} + s' \cdot \boldsymbol{\Omega}}(t')$. Here, $P_{\mathbf{x} + s' \cdot \boldsymbol{\Omega}}(t')$ is the cumulative distribution function of the material that the particle is about to enter. Next, we stitch the part of $P_{\mathbf{x} + s' \cdot \boldsymbol{\Omega}}(t)$ where $t \geq t'$ to the right of $P(s', \mathbf{x}, \boldsymbol{\Omega})$. Once the particle enters another material, the process repeats itself.

Finding the value of t' such that $P(s', \mathbf{x}, \boldsymbol{\Omega}) = P_{\mathbf{x} + s' \cdot \boldsymbol{\Omega}}(t')$ is equivalent to asking the question: How far could a particle have traveled inside the material at $\mathbf{x} + s' \cdot \boldsymbol{\Omega}$ to reach the same value of the cumulative distribution function $P(s, \mathbf{x}, \boldsymbol{\Omega})$ that we currently observe? With this intuitive understanding of the process, we are now ready to reformulate it in a more precise, mathematical language.

As before, we assume that $p_{\mathbf{x}}(s)$ defines the probability density function for the distance to the next collisions in an infinite, homogeneous domain that is filled with the material found at \mathbf{x} . Similarly, we interpret $P_{\mathbf{x}}(s)$. A cumulative distribution function, $P_{\mathbf{x}}(s)$ is invertible on $[0, 1]$ with $P_{\mathbf{x}}^{-1} : [0, 1] \rightarrow \mathbb{R}^{\geq 0}$. Abbreviate $\mathbf{x}(s) = \mathbf{x} + s \cdot \boldsymbol{\Omega}$; even though we can assume $\mathbf{x} = \mathbf{0}$ and

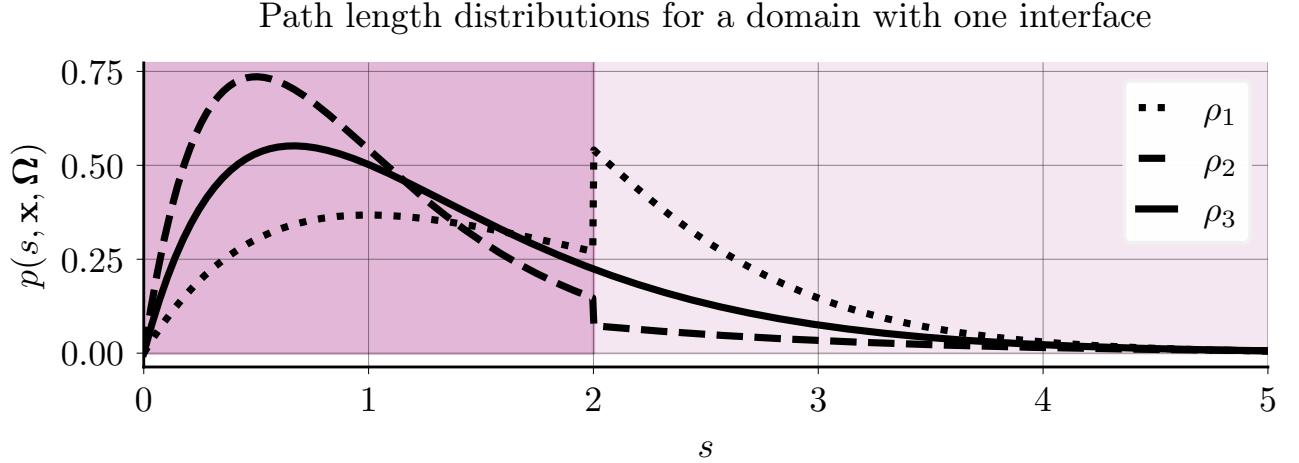
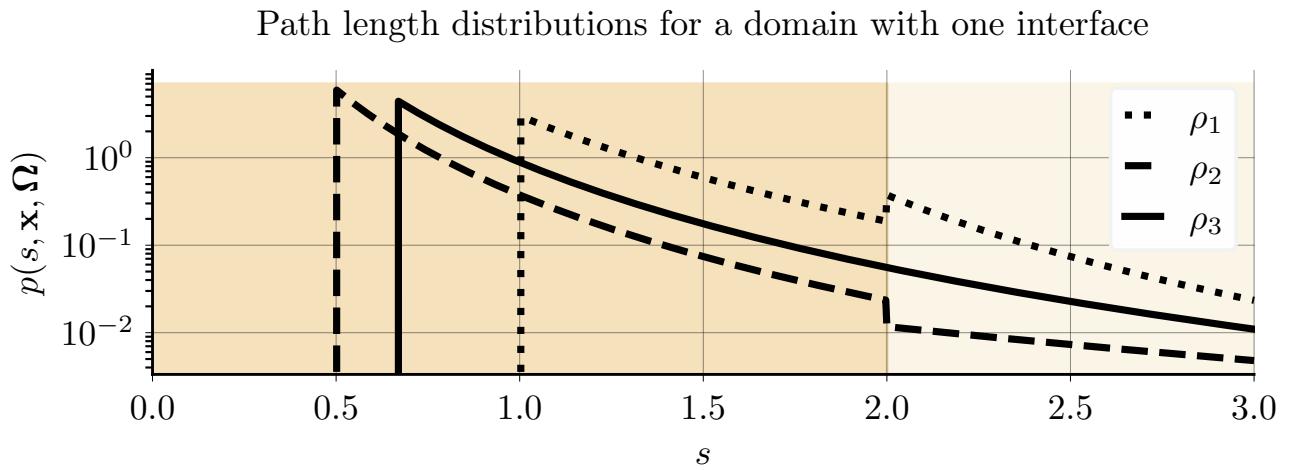
(a) Gamma distribution as $p_1(s)$.(b) Pareto distribution ($\alpha = 3$) as $p_1(s)$. The y -axis is log-scaled to better capture the discontinuity at $s = 2$.

Figure 5.8.: Path length distributions for a heterogeneous domain with a single interface and varying base distribution. The different lines show the distributions for different density fluctuations.

$\Omega = \mathbf{e}_1$ without loss of generality. The expression we are interested in is $p(s, \mathbf{x}, \Omega)$, interpreted as

$p(s, \mathbf{x}, \Omega)$ = “the probability that a particle which underwent a collision at \mathbf{x} and faces direction Ω will travel a distance s through a truly heterogeneous material to scatter at $\mathbf{x} + s \cdot \Omega$.”

For a given value $P(s, \mathbf{x}, \Omega)$, we find t such that $P_{\mathbf{x}(s)}(t) = P(s, \mathbf{x}, \Omega)$ via

$$t = P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \Omega)). \quad (5.9)$$

Stitching the right half of $P_{\mathbf{x}(s)}(t)$ to $P(s, \mathbf{x}, \Omega)$ implies that, for an infinitesimal distance Δs , we update

$$P(s + \Delta s, \mathbf{x}, \Omega) = P(s, \mathbf{x}, \Omega) + \Delta s \cdot p_{\mathbf{x}(s)}(t) \quad (5.10a)$$

$$= P(s, \mathbf{x}, \Omega) + \Delta s \cdot p_{\mathbf{x}(s)}(P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \Omega))). \quad (5.10b)$$

Rearranging terms, letting $\Delta s \rightarrow 0$, and noting that $P(0, \mathbf{x}, \Omega) = 0$, we obtain an ordinary differential equation (ODE) for $P(s, \mathbf{x}, \Omega)$ in the form of

$$\begin{cases} P(0, \mathbf{x}, \Omega) &= 0, \\ \frac{d}{ds} P(s, \mathbf{x}, \Omega) &= p_{\mathbf{x}(s)}(P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \Omega))). \end{cases} \quad (5.11)$$

Solving the above system for $P(s, \mathbf{x}, \Omega)$ and differentiating with respect to s we obtain

$$p(s, \mathbf{x}, \Omega) = \frac{d}{ds} P(s, \mathbf{x}, \Omega) = p_{\mathbf{x}(s)}(t) = p_{\mathbf{x}(s)}\left(P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \Omega))\right), \quad (5.12)$$

i.e., a distribution for traveling distance t in a material defined by the properties at $\mathbf{x}(s)$. Since $p_{\mathbf{x}(s)}(t)$ was chosen exactly such that it represents the probability of traveling distance s in the truly heterogeneous material, we know that we have to sample the distance to collision for a particle at \mathbf{x} facing direction Ω from

$$p_{\mathbf{x}(s)}\left(P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \Omega))\right) \quad (5.13)$$

with $P(s, \mathbf{x}, \Omega)$ as the solution of (5.11). In practice this means that for every particle in a heterogeneous domain, we have to solve (5.11) to generate a sample from the probability density function for the distance to collision. However, there are two cases where we can avoid the high computational costs for sampling: First in the case of generating multiple samples from one single distribution. This would be the case when sampling the distance to collision for particles emitted by a source, all facing the same direction. The second case deals with piecewise homogeneous materials. If we loosen the assumption of continuously varying material and assume the material to be piecewise homogeneous—which is a reasonable modeling assumption—we can avoid solving (5.11). Instead we march along the trajectory of the particle and only need to evaluate the cumulative distribution function and its inverse for the distance to collision of each material which we assumed to be known in the first place.

We can again validate that the generalized expression presented here reduces to the known results for simpler cases.

Is $p(s, \mathbf{x}, \Omega)$ a probability density function?

At every point \mathbf{x} , we know that $p_{\mathbf{x}}(s)$ is a probability density function. The mapping from s to t is one-to-one (t to $P(s, \mathbf{x}, \Omega)$ is one-to-one and $P(s, \mathbf{x}, \Omega)$ to s is one-to-one). Thus, with $P(s, \mathbf{x}, \Omega) = p_{\mathbf{x}(s)}(t)$, also $P(s, \mathbf{x}, \Omega)$ is a probability density function.

Is the case of a classical material treated correctly?

In the classical, possibly heterogeneous case, we have $p_{\mathbf{x}(t)}(s) = \rho(\mathbf{x}(t))e^{-\rho(\mathbf{x}(t))s}$, $P_{\mathbf{x}(t)}(s) = 1 - e^{-\rho(\mathbf{x}(t))s}$, and consequently $P_{\mathbf{x}(t)}^{-1}(y) = -\log(1 - y)/\rho(\mathbf{x}(t))$. We compute

$$p_{\mathbf{x}(s)}\left(P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \Omega))\right) = \rho(\mathbf{x}(s))(1 - P(s, \mathbf{x}, \Omega)) \quad (5.14)$$

via the inverse function theorem (IFT). The resulting ODE system is then

$$\begin{cases} P(0, \mathbf{x}, \Omega) &= 0, \\ \frac{d}{ds} P(s, \mathbf{x}, \Omega) &= \rho(\mathbf{x}(s))(1 - P(s, \mathbf{x}, \Omega)), \end{cases} \quad (5.15)$$

with solution

$$P(s, \mathbf{x}, \boldsymbol{\Omega}) = 1 - e^{-\int_0^s \rho(\mathbf{x}(s')) ds'}.$$

The resulting probability density function $p(s, \mathbf{x}, \boldsymbol{\Omega})$ is thus given by

$$p(s, \mathbf{x}, \boldsymbol{\Omega}) = \rho(\mathbf{x}(s)) e^{-\int_0^s \rho(\mathbf{x}(s')) ds'},$$

which matches the known results for the classical heterogeneous case.

How is the non-classical case treated?

Next, let us consider a material that is heterogeneous because of its density fluctuations and therefore not truly heterogeneous. For this scenario, the distribution

$$p(s, \mathbf{x}, \boldsymbol{\Omega}) = \rho(\mathbf{x} + s \cdot \boldsymbol{\Omega}) p_1 \left(\int_0^s \rho(\mathbf{x} + s' \cdot \boldsymbol{\Omega}) ds' \right)$$

alongside the scaling

$$p_{\rho(\mathbf{x})}(s) = \rho(\mathbf{x}) p_1(\rho(\mathbf{x}) \cdot s) \quad (5.16)$$

was introduced previously. Integrating the last expression yields $P_{\rho(\mathbf{x})}(s) = P_1(\rho(\mathbf{x}) \cdot s)$ with an inverse

$$P_{\rho(\mathbf{x})}^{-1}(y) = P_1^{-1}(y)/\rho(\mathbf{x}). \quad (5.17)$$

Abusing notation, we use $p_{\mathbf{x}}(s)$ and $p_{\rho(\mathbf{x})}(s)$ interchangeably. The ODE then satisfies

$$\begin{aligned} \frac{d}{ds} P(s, \mathbf{x}, \boldsymbol{\Omega}) &= p_{\mathbf{x}(s)} \left(P_{\mathbf{x}(s)}^{-1} (P(s, \mathbf{x}, \boldsymbol{\Omega})) \right) \\ &= \rho(\mathbf{x}(s)) p_1 \left(\rho(\mathbf{x}(s)) P_{\mathbf{x}(s)}^{-1} (P(s, \mathbf{x}, \boldsymbol{\Omega})) \right) \quad \text{by (5.16)} \\ &= \rho(\mathbf{x}(s)) p_1 \left(P^{-1} (P(s, \mathbf{x}, \boldsymbol{\Omega})) \right) \quad \text{by (5.17)} \\ &= \rho(\mathbf{x}(s)) \frac{1}{(P^{-1})'(P(s, \mathbf{x}, \boldsymbol{\Omega}))} \quad \text{by IFT.} \end{aligned}$$

Rearranging terms and integrating yields

$$\begin{aligned} (P^{-1})'(P(s, \mathbf{x}, \boldsymbol{\Omega})) \cdot \frac{d}{ds} P(s, \mathbf{x}, \boldsymbol{\Omega}) &= \rho(\mathbf{x}(s)) \\ \Leftrightarrow \quad \frac{d}{ds} P^{-1}(P(s, \mathbf{x}, \boldsymbol{\Omega})) &= \rho(\mathbf{x}(s)) \\ \Leftrightarrow \quad P^{-1}(P(s, \mathbf{x}, \boldsymbol{\Omega})) &= \int_0^s \rho(\mathbf{x}(s')) ds' \\ \Leftrightarrow \quad P(s, \mathbf{x}, \boldsymbol{\Omega}) &= P \left(\int_0^s \rho(\mathbf{x}(s')) ds' \right). \end{aligned}$$

Hence, for heterogeneity that results from density fluctuations, we recover the expression that we derived previously—translating the geometric distance to the optical distance by means of a base distribution.

Is void treated correctly?

Void regions imply a probability of zero to undergo collisions. Consequently, we have to modify the ODE system since, for the probability density function $p_{\mathbf{x}(t)}(s) \equiv 0$, there is no corresponding cumulative distribution function or an inverse of it. However, because the right-hand side of the ODE system evaluates the probability density function which maps all inputs to zero, a natural modification of the system is given by setting

$$P_{\mathbf{x}(s)} \left(P_{\mathbf{x}(s)}^{-1}(y) \right) := 0 \quad (5.18)$$

for any void region. In this case, the value of $P(s, \mathbf{x}, \Omega)$ will not increase within a void region, corresponding to a zero probability of scattering events.

Another subtlety arises in the case of a particle being sourced inside a material that satisfies $p_{\mathbf{x}(0)}(0) = 0$, i.e., a zero probability of undergoing a collision immediately at the start of the particle's trajectory. With the previous ODE system

$$\begin{cases} P(0, \mathbf{x}, \Omega) &= 0 \\ \frac{d}{ds} P(s, \mathbf{x}, \Omega) &= p_{\mathbf{x}(s)} \left(P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \Omega)) \right) \end{cases} \quad (5.19)$$

and with the assumption $p_{\mathbf{x}(0)}(0) = 0$ and the initial condition $P(0, \mathbf{x}, \Omega) = 0$ we obtain $P(s, \mathbf{x}, \Omega) \equiv 0$. However, $p_{\mathbf{x}(0)}(0) = 0$ has to be non-zero. Since $s = 0$, the last collision must have taken place at $\mathbf{x}(0) = \mathbf{x}$. Thus, with Ω as the direction of travel *away* from the last collision, $-\Omega$ points *towards* the obstacle that caused the collision at \mathbf{x} , causing the particle to collide instantaneously. Consequently, the probability to collide at $s = 0$ has to be nonzero.

Is the case of bounded domains treated correctly?

We restricted ourselves to unbounded domains so far. However, bounded domains can be handled similarly. When a particle reaches the boundary ∂V of the domain V , two scenarios are possible. Either the particle leaves the computational domain or it interacts with the boundary ∂V . In both cases, $P(s, \mathbf{x}, \Omega)$ is only valid for $s < s^*$ where s^* is chosen such that $\mathbf{x} + s^* \cdot \Omega \in \partial V$. Thus, before sampling the distance to the next collision, with probability $1 - P(s^*, \mathbf{x}, \Omega)$, a particle is placed at the boundary (where it either interacts with the boundary or leaves the domain) and with probability $P(s^*, \mathbf{x}, \Omega)$ a sample s from $P(s, \mathbf{x}, \Omega)$ for $s < s^*$ has to be generated as the distance to the next collision. However, this procedure is not specifically related to non-classical or heterogeneous transport.

5.5. Sampling path lengths

When generating samples from the distribution for the distance to the next collision, two scenarios have to be distinguished: (i) Generating multiple samples from a single distribution, and (ii) generating one sample per distribution for several distributions. The first scenario corresponds to a situation where many particles are emitted at the same position, facing the same direction (e.g., due to a source). The second, computationally more expensive scenario is relevant when many particles face different directions. For example, these particles may have been emitted by an isotropic source, thus the trajectories of the particles pass through different material compositions for different directions. In both scenarios, samples will be generated via inverse transform sampling. Here, we can again distinguish two cases. In the general case, the material composition might be different for every point $\mathbf{x}(t)$ along the trajectory. The second case, which is more reasonable in applications, assumes that a particle faces a trajectory along which the material is piecewise homogeneous, illustrated in Figure 5.4.

5.5.1. The general case

In both cases we will assume that $p_{\mathbf{x}(t)}(s)$, $P_{\mathbf{x}(t)}(s)$, and $P_{\mathbf{x}(t)}^{-1}(s)$ are available for all materials within the domain. To generate a single sample for the trajectory of one particle, the evolution equation for $P(s, \mathbf{x}, \Omega)$ has to be solved until a value u is reached which was sampled from a uniform distribution on $[0, 1]$ in advance. This procedure is put into pseudocode in Algorithm 5.1. Undeniably too expensive, we provide this algorithm only for completeness. More relevant for practical applications are Algorithms 5.2 and 5.3.

Algorithm 5.1 Generating a single sample in the general case.

```

1: function SingleSample( $\mathbf{x}(t), p_{\mathbf{x}(t)}(s), P_{\mathbf{x}(t)}(s), P_{\mathbf{x}(t)}^{-1}(s)$  )
2:    $u \sim \mathcal{U}[0, 1]$ 
3:   Solve (5.11) for  $s^*$  such that  $P(s^*, \mathbf{x}, \Omega) = u$ 
4:   return  $s^*$ 

```

If we instead decide to generate multiple samples from a single distribution, we can solve the ODE once and evaluate the inverse of $P(s, \mathbf{x}, \Omega)$ (inverse with respect to the first argument) arbitrarily often afterwards. Since we evaluate $P^{-1}(u, \mathbf{x}, \Omega)$ for $u \in [0, 1]$, we have to solve the ODE up until a final distance $P(s^*, \mathbf{x}, \Omega) = 1 - \varepsilon$ for some $0 < \varepsilon \ll 1$. Note that we can not solve the ODE up until $P(s^*, \mathbf{x}_0, \Omega) = 1$, since this would be the case for $s^* = \infty$. From the solution of the ODE, we can create a lookup table that lets us evaluate $P^{-1}(u, \mathbf{x}, \Omega)$. Following this precomputation, the requested number of samples can be generated by evaluating $P^{-1}(u_i, \mathbf{x}, \Omega)$ for u_i sampled from $\mathcal{U}[0, 1 - \varepsilon]$ and i ranging from one to the number of requested samples, illustrated in Algorithm 5.2.

Algorithm 5.2 Generating multiple samples in the general case.

```

1: function MultipleSample( $\varepsilon, n, \mathbf{x}(t), p_{\mathbf{x}(t)}(s), P_{\mathbf{x}(t)}(s), P_{\mathbf{x}(t)}^{-1}(s)$  )
2:   Solve (5.11) up to  $s^*$  such that  $P(s^*, \mathbf{x}, \Omega) \leq 1 - \varepsilon$ 
3:    $T = \text{lookup table}(P^{-1}(u, \mathbf{x}, \Omega))$  for  $u$  from 0 to  $1 - \varepsilon$ 
4:    $y = \text{zeros}(n)$ 
5:   for  $i = 1, \dots, n$  do
6:      $u_i \sim \mathcal{U}[0, 1 - \varepsilon]$ 
7:      $y[i] = T(u_i)$ 
8:   return  $y$ 

```

5.5.2. Sampling in piecewise homogeneous materials

For piecewise homogeneous, non-classical materials, the computational costs for generating samples reduces dramatically. In the following, we will derive a formulation for the sampling procedure that allows to generate samples for the distance to the next collision for particles by iteratively tracking their trajectories through the spatial domain without having to solve the previous ODE system.

As illustrated in Figure 5.4, we assume a setting where a particle traverses different materials, each defined by $p_i(s)$ and $P_i(s)$. The distance from \mathbf{x} to the i -th interface is given by s_i , with

Algorithm 5.3 Generating a single sample in the piecewise homogeneous case (without void).

```

1: function SingleSamplePiecewise( $s_i$ ,  $p_i(s)$ ,  $P_i(s)$ ,  $P_i^{-1}(s)$  for all materials  $i$ 
                                 along the particle's trajectory)
2:    $i = 0$ ,  $y = 0.0$ ,  $t = 0.0$ 
3:    $u \sim \mathcal{U}[0, 1]$ 
4:   while True do
5:     Solve  $u = P_{i+1}(s - t + P_{i+1}^{-1}(y))$  for  $s$ 
6:     if  $s \in [s_i, s_{i+1})$  then
7:       return  $s$ 
8:     else
9:        $y = P_{i+1}(s_{i+1} - t + P_{i+1}^{-1}(y))$ 
10:       $t = s_{i+1}$ 
11:       $i = i + 1$ 

```

$s_0 = 0$. For $s \in [0, s_1)$, we write

$$\begin{cases} P(0, \mathbf{x}, \Omega) = 0, \\ \frac{d}{ds} P(s, \mathbf{x}, \Omega) = p_{\mathbf{x}(s)}(P_{\mathbf{x}(s)}^{-1} P(s, \mathbf{x}, \Omega)) \\ \qquad\qquad\qquad = p_1(p_1^{-1}(P(s, \mathbf{x}, \Omega))) = 1/(P_1^{-1})'(P(s, \mathbf{x}, \Omega)). \end{cases} \quad (5.20)$$

The last line is equivalent to

$$(P_1^{-1})'(P(s, \mathbf{x}, \Omega)) \cdot \frac{d}{ds} P(s, \mathbf{x}, \Omega) = \frac{d}{ds} P_1^{-1}(P(s, \mathbf{x}, \Omega)). \quad (5.21)$$

Thus,

$$P_1^{-1}(P(s, \mathbf{x}, \Omega)) = s + c_1 \quad (5.22a)$$

$$\Leftrightarrow P(s, x, \Omega) = P_1(s + c_1). \quad (5.22b)$$

Since $P(0, \mathbf{x}, \Omega) = 0$, we have $c_1 = 0$, i.e., $P(s, \mathbf{x}, \Omega) = P_1(s)$ for $s \in [0, s_1]$. Next, assume a particle to be at the interface between material i and $i + 1$, having already traveled a distance s_i . The value $P(\mathbf{x}, \Omega, s_i) = y_i$ is known. For $s \in [s_i, s_{i+1})$ we write

$$\begin{cases} P(s_i, \mathbf{x}, \Omega) &= y_i, \\ \frac{d}{ds} P(s, \mathbf{x}, \Omega) &= 1/(P_{i+1}^{-1})'(P(s, \mathbf{x}, \Omega)), \end{cases} \quad (5.23)$$

or with $\tilde{P}(\tilde{s}, \mathbf{x}, \Omega) = P(s_i + \tilde{s}, \mathbf{x}, \Omega) = P(s, \mathbf{x}, \Omega)$

$$\begin{cases} \tilde{P}(0, \mathbf{x}, \Omega) &= y_i, \\ \frac{d}{d\tilde{s}}\tilde{P}(\tilde{s}, \mathbf{x}, \Omega) &= 1/(P_{i+1}^{-1})'(\tilde{P}(\tilde{s}, \mathbf{x}, \Omega)). \end{cases} \quad (5.24)$$

We solve for $\tilde{P}(\tilde{s}, \mathbf{x}, \Omega)$ to get

$$\tilde{P}(\tilde{s}, \mathbf{x}, \Omega) = P_{i+1}(\tilde{s} + c_{i+1}) \quad (5.25)$$

Then, $c_{i+1} = P_{i+1}^{-1}(y_i)$ and consequently

$$P(s, \mathbf{x}, \Omega) = \tilde{P}(\tilde{s}, \mathbf{x}, \Omega) \quad (5.26a)$$

$$= P_{i+1}(\tilde{s} + P_{i+1}^{-1}(y_i)) \quad (5.26b)$$

$$= P_{i+1}(s - s_i + P_{i+1}^{-1}(P(s_i, \mathbf{x}, \Omega))), \quad (5.26c)$$

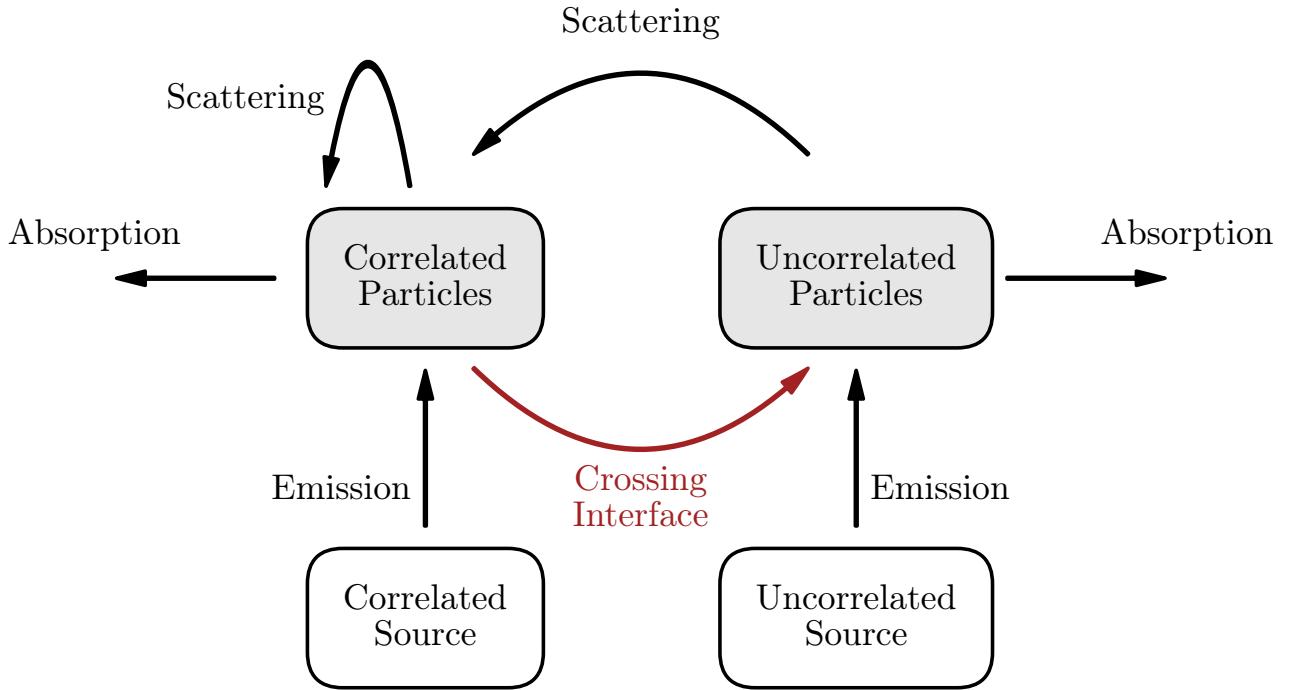


Figure 5.9.: Life cycle of correlated and uncorrelated particles in the heterogeneous case.

for $s \in [s_i, s_{i+1}]$. With $P(s) = P_{i+1}(s - s_i + P_{i+1}^{-1}(P(s_i, \mathbf{x}, \Omega)))$ inverse transform sampling is easy: Finding s^* such that $P(s^*, \mathbf{x}, \Omega) = u$ can be solved by iteratively evaluating the expression in (5.26c) and checking whether the argument of P_{i+1} lies within $[s_i, s_{i+1}]$. The procedure is written out as pseudocode in Algorithm 5.3.

5.6. Correlated and uncorrelated particles with heterogeneities

Having discussed heterogeneities, we are now equipped with the necessary tools to reconsider the notion of correlated and uncorrelated particles—this time for heterogeneous materials. A correlated particle follows potentially different dynamics than an uncorrelated particle. Recall, for example, the obstacle field that resulted from a Poisson disk sampling procedure, where scatterers obey a certain minimal distance. A particle that is placed at a random position inside the obstacle field might collide after traveling a distance smaller than the minimal distance between scatterers; something that is not possible for a particle that just underwent a collision.

If that particle now moves from one homogeneous subdomain into another, it again behaves like a particle that was randomly seeded at the interface between two materials, agnostic to a minimal distance between scatterers. It became, ipso facto, an uncorrelated particle. The notion of correlated and uncorrelated particles is therefore intertwined with the memory-resetting ansatz. Crossing a domain interface, a particle becomes indistinguishable from a particle that is sourced at the same position (with the same direction).

This naturally defines interface conditions for the heterogeneous case. Consider the situation in Figure 5.10. For a domain that spans all of \mathbb{R}^2 , we place an interface at $x = 0$. Quantities on the left (or right) side of the interface are denoted with an additional superscript L (or R).

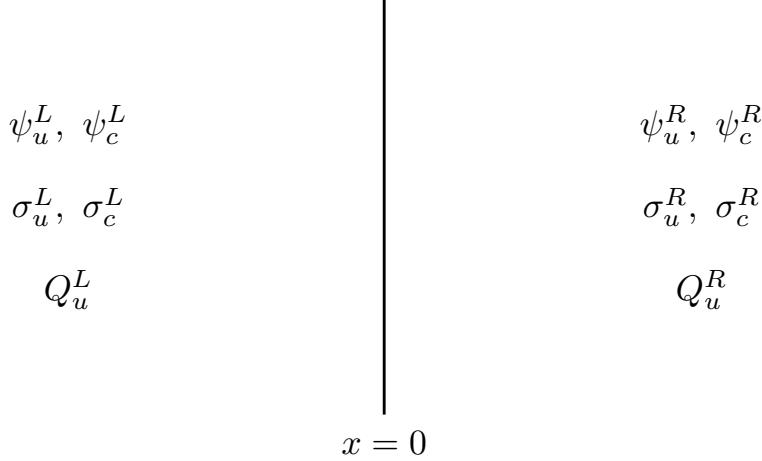


Figure 5.10.: An infinite, heterogeneous domain with an interface at $x = 0$.

The equations for both sides of the domain are then given by

$$\partial_s \psi_u^L(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_u^L(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_{t,u}^L(s) \psi_u^L(s, \mathbf{x}, \boldsymbol{\Omega}) = \delta(s) Q_u^L(\mathbf{x}), \quad (5.27a)$$

$$\begin{aligned} \partial_s \psi_c^L(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_c^L(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_{t,c}^L(s) \psi_c^L(s, \mathbf{x}, \boldsymbol{\Omega}) \\ = c \delta(s) \int_0^\infty \int_{4\pi} s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) [\sigma_{t,c}^L(s') \psi_c^L(s', \mathbf{x}, \boldsymbol{\Omega}') + \sigma_{t,u}^L(s') \psi_u^L(s', \mathbf{x}, \boldsymbol{\Omega}')] d\boldsymbol{\Omega}' ds', \end{aligned} \quad (5.27b)$$

for $x \leq 0$, and

$$\partial_s \psi_u^R(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_u^R(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_{t,u}^R(s) \psi_u^R(s, \mathbf{x}, \boldsymbol{\Omega}) = \delta(s) Q_u^R(\mathbf{x}), \quad (5.28a)$$

$$\begin{aligned} \partial_s \psi_c^R(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi_c^R(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_{t,c}^R(s) \psi_c^R(s, \mathbf{x}, \boldsymbol{\Omega}) \\ = c \delta(s) \int_0^\infty \int_{4\pi} s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) [\sigma_{t,c}^R(s') \psi_c^R(s', \mathbf{x}, \boldsymbol{\Omega}') + \sigma_{t,u}^R(s') \psi_u^R(s', \mathbf{x}, \boldsymbol{\Omega}')] d\boldsymbol{\Omega}' ds', \end{aligned} \quad (5.28b)$$

for $x \geq 0$. Interface conditions describe how particles—correlated and uncorrelated ones—that leave the domain become uncorrelated particles with $s = 0$ for the domain they move into. They are given by

$$\psi_u^L(0, \boldsymbol{\Omega}, s) = \delta(s) \int_0^\infty \psi_u^R(0, \boldsymbol{\Omega}, s') + \psi_c^R(0, \boldsymbol{\Omega}, s') ds' \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_L < 0, \quad (5.29a)$$

$$\psi_c^L(0, \boldsymbol{\Omega}, s) = 0 \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_L < 0, \quad (5.29b)$$

$$\psi_u^R(0, \boldsymbol{\Omega}, s) = \delta(s) \int_0^\infty \psi_u^L(0, \boldsymbol{\Omega}, s') + \psi_c^L(0, \boldsymbol{\Omega}, s') ds' \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_R < 0, \quad (5.29c)$$

$$\psi_c^R(0, \boldsymbol{\Omega}, s) = 0 \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_R < 0. \quad (5.29d)$$

This single-interface example can easily be generalized to arbitrary complex geometries. Consider therefore a heterogeneous domain $\bar{V} \subset \mathbb{R}^d$ with $d \in \{1, 2, 3\}$, and assume $\bar{V} = \cup_{i=1}^N \bar{V}_i$, with all open V_i being homogeneous and $V_i \cap V_j = \emptyset$ if $i \neq j$. Let ∂V denote the boundary of \bar{V} . With $\partial V_i = \partial V_i^I \cup \partial V_i^\Gamma$ we divide the boundary of \bar{V}_i into interface boundaries ∂V_i^I and non-interface boundaries ∂V_i^Γ . The interior indicator function χ_{in} satisfies $\chi_{\text{in}}(\mathbf{x}) = i$ iff

$\mathbf{x} \in V_i$. The outward pointing normal of \bar{V}_i is $\mathbf{n}_i(\mathbf{x})$ for $\mathbf{x} \in \partial V_i$. For $\mathbf{x} \in \partial V_i^I \cap \partial V_j^I$, the exterior indicator function χ_{ex} satisfies $\chi_{\text{ex}}(i, \mathbf{x}) = j$ if $\mathbf{n}_i(\mathbf{x}) = -\mathbf{n}_j(\mathbf{x})$, i.e. \bar{V}_j neighbors \bar{V}_i at \mathbf{x} . All cross sections, sources, and the angular fluxes receive a subscript for the respective domain that they are restricted to. Abusing notation, $l(\mathbf{x}, \Omega)$ denotes now the distance to the next boundary or interface, whichever is closer.

The backward formulation

We start with the backward formulation where s represents the distance since the last event. All non-interface boundaries require boundary conditions for the incoming angular flux, given for $i = 1, \dots, N$ by

$$\psi_c^i(s, \mathbf{x}, \Omega) = 0 \quad \text{for } \Omega \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^\Gamma, \quad (5.30a)$$

$$\psi_u^i(s, \mathbf{x}, \Omega) = \delta(s) \psi_u^{bc}(\mathbf{x}, \Omega) \quad \text{for } \Omega \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^\Gamma. \quad (5.30b)$$

Interface conditions for $i = 1, \dots, N$ are the generalization of (5.29), i.e.,

$$\psi_c^i(s, \mathbf{x}, \Omega) = 0 \quad \text{for } \Omega \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^\Gamma, \quad (5.31a)$$

$$\psi_u^i(s, \mathbf{x}, \Omega) = \delta(s) \int_0^{l(\mathbf{x}, \Omega)} \psi_u^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \Omega) ds' \quad \text{for } \Omega \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^\Gamma, \quad (5.31b)$$

with $\psi^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \Omega) = \psi_u^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \Omega) + \psi_c^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \Omega)$. Transport for the correlated and uncorrelated flux is described cell-wise for $\mathbf{x} \in \bar{V}_i$ via

$$\partial_s \psi_u^i(s, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi_u^i(s, \mathbf{x}, \Omega) + \sigma_{t,u}^i(s) \psi_u^i(s, \mathbf{x}, \Omega) = \delta(s) Q_u^i(\mathbf{x}), \quad (5.32a)$$

$$\begin{aligned} \partial_s \psi_c^i(s, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} \psi_c^i(s, \mathbf{x}, \Omega) + \sigma_{t,c}^i(s) \psi_c^i(s, \mathbf{x}, \Omega) \\ = c \delta(s) \int_0^{l(\mathbf{x}, \Omega)} \int_{\mathbb{S}^2} s(\Omega' \cdot \Omega) [\sigma_{t,c}^i(s') \psi_c^i(s', \mathbf{x}, \Omega') + \sigma_{t,u}^i(s') \psi_u^i(s', \mathbf{x}, \Omega')] d\Omega' ds'. \end{aligned} \quad (5.32b)$$

Summarizing, we can model non-classical linear transport in heterogeneous media via the boundary conditions (5.30), the interface conditions (5.31), and the transport equations (5.32).

The forward formulation

Recall the forward formulation of non-classical transport in its simplest form

$$-\partial_s f(s, \mathbf{x}, \Omega) + \Omega \cdot \nabla_{\mathbf{x}} f(s, \mathbf{x}, \Omega) = c p(s) \int_{\mathbb{S}^2} s(\Omega' \cdot \Omega) f(0, \mathbf{x}, \Omega') d\Omega', \quad (5.33)$$

equipped with the boundary condition

$$f(s, \mathbf{x}, \Omega) = p(s) f^{bc}(\mathbf{x}, \Omega) \quad \text{for } \Omega \cdot \mathbf{n}(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V^\Gamma, \quad (5.34)$$

and s now interpreted as the distance to the next interaction. To model heterogeneous transport, we again need to distinguish correlated from uncorrelated particles and consider interface conditions. As with the backward formulation, we assume that the memory of particles is lost when crossing a domain interface. Consequently, we need to re-sample the distance to collision in the forward formulation. With the same spatial domain as before, we end up with the equations that follow next.

Since particles enter the domain uncorrelated, boundary conditions are given by

$$f_c^i(s, \mathbf{x}, \boldsymbol{\Omega}) = 0 \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^\Gamma, \quad (5.35a)$$

$$f_u^i(s, \mathbf{x}, \boldsymbol{\Omega}) = p_u^i(s) f_u^{bc}(\mathbf{x}, \boldsymbol{\Omega}) \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^\Gamma. \quad (5.35b)$$

The interface conditions for $i = 1, \dots, N$ read

$$f_c^i(s, \mathbf{x}, \boldsymbol{\Omega}) = 0 \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^I, \quad (5.36a)$$

$$f_u^i(s, \mathbf{x}, \boldsymbol{\Omega}) = p_u^i(s) \int_0^{l(\mathbf{x}, \boldsymbol{\Omega})} f^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \boldsymbol{\Omega}) ds' \quad \text{for } \boldsymbol{\Omega} \cdot \mathbf{n}_i(\mathbf{x}) < 0 \text{ and } \mathbf{x} \in \partial V_i^I, \quad (5.36b)$$

with $f^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \boldsymbol{\Omega}) = f_u^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \boldsymbol{\Omega}) + f_c^{\chi_{\text{ex}}(i, \mathbf{x})}(s', \mathbf{x}, \boldsymbol{\Omega})$. Transport for the correlated and uncorrelated flux is then described cell-wise for $\mathbf{x} \in \bar{V}_i$ via

$$-\partial_s f_u^i(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} f_u^i(s, \mathbf{x}, \boldsymbol{\Omega}) = p_u^i(s) Q_u^i(\mathbf{x}), \quad (5.37a)$$

$$-\partial_s f_c^i(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} f_c^i(s, \mathbf{x}, \boldsymbol{\Omega}) = c p_c^i(s) \int_{\mathbb{S}^2} s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) [f_c^i(0, \mathbf{x}, \boldsymbol{\Omega}') + f_u^i(0, \mathbf{x}, \boldsymbol{\Omega}')] d\boldsymbol{\Omega}'. \quad (5.37b)$$

Cross sections and path length distributions are connected as before.

This added complexity obviously comes at a cost. Evolving two distributions in time inevitably doubles the computational complexity. To justify this, experimental work is necessary that quantifies the improvements that a distinction between *correlated* and *uncorrelated* particles can achieve.

Part III.

Concluding remarks

6. Summary

As already hinted in this document's name, we have covered two topics—classical and non-classical transport—with three different perspectives—theory, modeling, and numerics.

The initial microscopic description of transport in Chapter 1 was not only crucial in understanding the origin of kinetic theory, it also highlighted how different modeling assumptions lead to different governing equations. This has manifested itself throughout Part II: Changing the scatterers' positions from completely random to a lattice structure fundamentally changed the dynamics of particle transport, requiring a completely new set of governing equations.

Moreover, the particle billiard game of Chapter 1 directly motivated the Monte Carlo method as one member of the class of numerical algorithms to solve the transport equation that were presented in Chapter 2.

We discussed the crucial role of spherical quadratures for the discrete ordinates method and presented quadratures that are based on the triangulation of Platonic solids, resulting in a highly uniform distribution of quadrature points on the unit sphere.

The underlying connectivity of these quadratures was imperative to the successful realization of the rS_N method. Together with $as-S_N$, these two ray effect mitigation techniques drastically reduced the spurious oscillations that are inherent to the discrete ordinates method.

We saw in Chapter 3 that ray effects are a result of incorrect computation of the spherical integral. The rS_N and $as-S_N$ method both circumvent this problem to an extend where it is possible to (i) either gain a significant boost in accuracy while keeping the runtime constant, or (ii) reduce the runtime while keeping the solution quality virtually unaltered.

The introduction of non-classical particle transport in Chapter 4 allowed us to tackle questions around non-classical transport in heterogeneous materials. Throughout Chapter 5, we provided different ansätze to model the behavior of particles that cross material interfaces; each obeying different theoretical requirements. The existing notion of correlated and uncorrelated particles was also extended to include heterogeneities.

Let us now discuss the main results of the different chapters in more detail. Acknowledging the introductory nature of Chapter 1, we start with Chapter 2.

Numerical solution methods

Three different solution methods for transport equations, as well as their main ingredients were presented and analyzed.

For the Monte Carlo method, we discussed ways to efficiently sample random numbers, e.g., via inverse transform sampling or rejection sampling. Moreover, we discussed the advantages and disadvantages of interpreting Monte Carlo as a numerical integrator that can be used to approximately solve the transport equation.

The main focus was on the discrete ordinates method where we started with a discussion of quadrature sets. The octahedron- and icosahedron-based quadratures (Camminady, Frank, and

Kusch [22]) have three desirable properties: (i) Arbitrary many quadrature points with strictly positive quadrature weights can be generated at low costs; (ii) the ratio between the maximal and minimal quadrature weight can be kept at around 2.0 and below 1.5 for the icosahedron *lerp* and *slerp* versions, respectively; and (iii) the resulting triangulation can be used to easily interpolate function values at points that are not included in the original quadrature set.

For transport sweeps as one of the ways to solve the system of S_N equations, we have proven that it is always possible to find an ordering that allows us to march through all spatial cells while considering the involved dependencies.

Representing the class of moment methods, we have seen that the P_N method can be used to obtain symmetrical solutions, albeit at the cost of oscillations that can render the solution negative.

Ray effects and their mitigation

With the help of a simple experiment, we have seen that ray effects are ultimately not the result of restricting transport to a fixed set of ordinates. Even in cases where we can solve for the angular flux $\psi(t, \mathbf{x}, \Omega)$ exactly (or at least up to machine precision), ray effects in the scalar flux $\phi(t, \mathbf{x})$ are omnipresent. Instead, the occurrence of these numerical artifacts can be attributed to the inexact approximation

$$\int_{\mathbb{S}^2} \psi(t, \mathbf{x}, \Omega') d\Omega' \approx \sum_{q'=1}^{n_q} w_{q'} \psi(t, \mathbf{x}, \Omega_{q'}).$$

Both the presented rS_N method (Camminady, Kusch, Frank, and Küpper [21]) and the $as-S_N$ method (Frank, Kusch, Camminady, and Hauck [37]) mitigated this problem by making the angular flux easier to integrate, once via additional angular diffusion and once with the help of artificial scattering.

By adding a rotation-and-interpolation step to the standard S_N method, the rS_N method significantly reduced ray effects in the line-source problem and the lattice test case. The solution quality of the scalar flux that is computed with the rS_N method is on par with that of the S_N method with up to three times the number of quadrature points. Adding the rotation-and-interpolation step only marginally increases the runtime of the S_N method (by up to approximately 10% in our implementation). A drawback of the rS_N method is the change of ordinates between two consecutive time steps. As a result, the order of transport sweeps needs to be recomputed which might create a bottleneck in highly optimized, many-core implementations on large-scale computers. However, forth-and-back rotations with two interpolation steps have the potential to overcome this problem by keeping the quadrature set fixed.

The $as-S_N$ method altered the transport equation by including an additional scattering term with scattering kernel

$$s_\varepsilon(\mu) = \frac{2}{\sqrt{\pi} \varepsilon \operatorname{Erf}\left(\frac{2}{\varepsilon}\right)} \exp\left(-\frac{(1-\mu)^2}{\varepsilon^2}\right).$$

Asymptotically, this forward-peaked type of scattering effectively behaves like a Fokker-Planck operator. Furthermore, the choice $\varepsilon = \beta/n_q$ ensured that the added effect vanishes in the limit of $n_q \rightarrow \infty$ as we recover the original S_N equations. Together with σ_{as} , the $as-S_N$ method then requires the user to select two parameters. For the line-source test case, we have seen an error reduction in the scalar flux of a factor four. Transferring the same pair of optimal parameters

to the lattice test case on a significantly finer grid (in space, angle, and time) still reduced the error down to 72% of the original error. This indicated that both different test cases as well as different discretizations can be used as a surrogate model with which optimal parameters can be derived at low costs before applying them to the actual test case of interest.

Since the as- S_N method does not change the initial ordinates, it is potentially better suited for optimized codes than the r S_N method.

Non-classical transport equations

This chapter introduced the non-classical transport equation in its forward and backward formulation. Several examples and numerical experiments revealed discrepancies between the behavior predicted by classical transport and the observations that are made. A transformation between both formulations is possible (Larsen, Frank, and Camminady [57]), and we can equivalently encode the non-classical dynamics in s -dependent cross sections or non-exponential path length distributions.

Furthermore, the existing concept of *correlated* and *uncorrelated* particles was presented, indicating the necessity to differentiate between particles that have already collided with scatterers from those that have not.

Non-classical transport in heterogeneous materials

Different ways to model material heterogeneities were introduced and juxtaposed. Depending on the model, we were able to argue for a preservation or a reset of the particles' memory. For both cases, we extended the non-classical cross sections and path length distributions to include material heterogeneities (Camminady, Frank, and Larsen [20]). In the more complicated setting of *true* heterogeneity in which the particles' memory should be preserved, the ODE system

$$\begin{cases} P(0, \mathbf{x}, \boldsymbol{\Omega}) &= 0, \\ \frac{d}{ds} P(s, \mathbf{x}, \boldsymbol{\Omega}) &= p_{\mathbf{x}(s)}(P_{\mathbf{x}(s)}^{-1}(P(s, \mathbf{x}, \boldsymbol{\Omega}))) \end{cases}$$

had to be solved in order to generate samples of the path length distribution. Since this is costly, we provided efficient sampling strategies, applicable to (reasonably) simplified scenarios. Additionally, we discussed a series of sanity checks to investigate the validity of the proposed distribution.

Lastly, the notion of correlated and uncorrelated particles was extended to include material heterogeneities as well. Particles that cross a material interface enter the new domain as uncorrelated particles—ultimately, this was the reason to not adopt the notion of *collided* and *uncollided* particles. As a result, two species of particles are described by two coupled non-classical transport equations.

7. Outlook

The presented ray effect mitigation techniques are implemented in a Julia research code that cannot be used to make predictions about the effectiveness of both methods on large-scale computing systems. An integration of both methods into production-ready software is therefore necessary to investigate both methods on a more quantitative level—rather than the more qualitative studies included in this thesis.

Complexity that arises from adding energy-dependency, unstructured meshes, or high-order time-integration is another hurdle that needs to be tackled as a logical next step. Several improvements, especially for the as- S_N method are apparent. (i) A more granular tuning of the amount of artificial scattering is desirable. The artificial scattering strength σ_{as} can become space-dependent. Moreover, work that researches whether it is sufficient to only add artificial scattering temporarily is underway. (ii) Whether it is possible to transfer optimal parameter sets from low-cost surrogate models to the actual high-cost simulation needs to be tested more broadly. To promote the as- S_N method as a reliable ray effect mitigation technique, it is necessary to provide guidance on how to choose the involved parameters optimally and efficiently. (iii) For the r S_N method, changing ordinates in every time-step might impact the sweeping procedure too severely to prove useful. If this is the case, versions of the r S_N method that do not suffer from this problem need to be investigated more rigorously. For example, instead of rotating once per time step, it is possible to perform two rotations—once forth, once back—in every time step with half the rotation magnitude. This has already been discussed, but not further analyzed.

For the r S_N and as- S_N method, certain test cases might be especially ill-suited. For example, in settings where particles are expected to advect freely through certain parts of the domain due to σ_t being small, the added diffusion and artificial scattering are problematic. For the as- S_N method, this relates to the previous point (i), i.e., the necessity to make σ_{as} space-dependent. For the r S_N method this is more difficult, because the rotation magnitude is identical in every spatial cell—at least in the standard r S_N method. For the r S_N method with forth-and-back rotations, the rotation magnitude can become space-dependent, since the set of ordinates effectively remains unchanged. For time-dependent problems, the parameters of both r S_N and as- S_N can theoretically become time-dependent, too. However, this was not investigated in this work and future research is necessary.

Current research on angular adaptivity for the S_N method has shown promising results [31]. The angular quadrature set is refined or coarsened in different areas of the spatial domain and at different time steps. The filtered spherical harmonics method is used as an error indicator, telling the algorithm where to refine or coarsen the angular mesh. This error indicator can, in theory, be used for the as- S_N or r S_N method as well, indicating regions that require artificial scattering or rotation, respectively.

Compared to classical transport, non-classical transport theory is still in its infancy. Here too, quantitative studies are necessary to understand the effect that s -dependent cross sections or non-exponential path length distributions have on real-world problems. So far, mostly the

Monte Carlo method has been explored as a way to solve the non-classical transport equation. However, non-classical time-independent transport is structurally similar to time-dependent classical transport. Whereas time only increases, the path length s gets reset after every collision (in the backward formulation). Nevertheless, if we rewrite the backward formulation as

$$\begin{cases} \partial_s \psi(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_t(s) \psi(s, \mathbf{x}, \boldsymbol{\Omega}) = 0, \\ \psi(0, \mathbf{x}, \boldsymbol{\Omega}) = \int_0^{l(\mathbf{x}, \boldsymbol{\Omega})} \int_{\mathbb{S}^2} c \sigma_t(s') s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \psi(s', \mathbf{x}, \boldsymbol{\Omega}') d\boldsymbol{\Omega}' ds' + Q(\mathbf{x}), \end{cases}$$

we can add indices l and $l - 1$ to the equation to obtain

$$\begin{cases} \partial_s \psi^l(s, \mathbf{x}, \boldsymbol{\Omega}) + \boldsymbol{\Omega} \cdot \nabla_{\mathbf{x}} \psi^l(s, \mathbf{x}, \boldsymbol{\Omega}) + \sigma_t(s) \psi^l(s, \mathbf{x}, \boldsymbol{\Omega}) = 0, \\ \psi^l(0, \mathbf{x}, \boldsymbol{\Omega}) = \int_0^{l(\mathbf{x}, \boldsymbol{\Omega})} \int_{\mathbb{S}^2} c \sigma_t(s') s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}) \psi^{l-1}(s', \mathbf{x}, \boldsymbol{\Omega}') d\boldsymbol{\Omega}' ds'. \end{cases}$$

With $\psi^0(s, \mathbf{x}, \boldsymbol{\Omega}) \equiv Q(\mathbf{x})$, we can solve this system repeatedly with a standard S_N solver. In this formulation, ψ^{l+1} relates to those particles that have scattered at most l times. Assuming convergence due to sufficient absorption, $\psi^L(s, \mathbf{x}, \boldsymbol{\Omega}) \approx \psi(s, \mathbf{x}, \boldsymbol{\Omega})$ for a sufficiently large L . True when moving from classical to non-classical transport, the added complexity of the distinction between correlated and uncorrelated particles needs to be quantified, too.

Next is the topic of heterogeneities. Ultimately, the proposed distributions need to be validated by means of real-world, physical experiments or computer simulations. Actually simulating trajectories of particles throughout a heterogeneous field of non-classically arranged scatterers and tallying those that do cross a domain interface is computationally expensive, if a sufficient statistics is to be achieved. This procedure gets more complicated when acknowledging the fact that we have limited knowledge about non-classical arrangements of scatterers. For example, algebraic decay of the path length distribution for the periodic Lorentz gas is observed only asymptotically. However, it does seem as if this is problem is conquerable by efficient implementations of the particle billiard games mentioned throughout this thesis.

The computer graphics community has been using non-exponential transport as a tool that enhances the artistic degree of freedom when creating renders [14, 48]. The s -dependent cross sections are chosen freely, without considering whether they correspond to a certain arrangement of obstacle. Understanding which cross sections (and path length distributions) are the result of actual arrangements of scatterers might prove fruitful beyond mathematical curiosity: It enables us to better understand the physical world around us and make simulations and renders more accurate.

Bibliography

- [1] I.K. Abu-Shumays. Angular quadratures for improved transport computations. *Transport Theory and Statistical Physics*, 30(2-3):169–204, 2001. [3](#)
- [2] Marvin L. Adams and Edward W. Larsen. Fast iterative methods for discrete-ordinates particle transport calculations. *Progress in Nuclear Energy*, 40(1):3–159, 2002. ISSN 0149-1970. doi: 10.1016/S0149-1970(01)00023-3. URL <http://www.sciencedirect.com/science/article/pii/S0149197001000233>. [2.2.5](#)
- [3] Graham W. Alldredge, Cory D. Hauck, and Andre L. Tits. High-order entropy-based closures for linear transport in slab geometry ii: A computational study of the optimization problem. *SIAM Journal on Scientific Computing*, 34(4):B361–B391, 2012. [2.3.2](#)
- [4] Graham W. Alldredge, Cory D. Hauck, Dianne P. O Leary, and Andre L. Tits. Adaptive change of basis in entropy-based moment closures for linear kinetic equations. *Journal of Computational Physics*, 258:489–508, 2014. [2.3.2](#)
- [5] P.M. Anselone and A.G. Gibbs. Convergence of the discrete ordinates method for the transport equation. In *Constructive and Computational Methods for Differential and Integral Equations*, pages 1–27. Springer, 1974. [2.2](#)
- [6] Albert Arking. Absorption of solar energy in the atmosphere: Discrepancy between model and observations. *Science*, 273(5276):779–782, 1996. [4.0.1](#)
- [7] Roland Assaraf and Michel Caffarel. Zero-variance principle for monte carlo algorithms. *Physical review letters*, 83(23):4682, 1999. [2.1.1.3](#)
- [8] Kendall Atkinson. Numerical integration on the sphere. *The ANZIAM Journal*, 23(3):332–347, 1982. [2.2.2](#)
- [9] Robert Axtell. Why agents?: On the varied motivations for agent computing in the social sciences. 2000. [2.1](#)
- [10] Hans Babovsky. *Die Boltzmann-Gleichung: Modellbildung-Numerik-Anwendungen*. Springer, 1998. [1.2](#)
- [11] J. Baró, J. Sempau, J.M. Fernández-Varea, and F. Salvat. Penelope: An algorithm for monte carlo simulation of the penetration and energy loss of electrons and positrons in matter. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 100(1):31–46, 1995. ISSN 0168-583X. doi: 10.1016/0168-583X(95)00349-5. URL <http://www.sciencedirect.com/science/article/pii/0168583X95003495>. [2.1.2](#)
- [12] N. Bellomo and L. Arlotti. *Lecture Notes on the Mathematical Theory of the Boltzmann Equation*, volume 33. World Scientific, 1995. [1](#)

- [13] Dobriyan M. Benov. The manhattan project, the first electronic computer and the monte carlo method. *Monte Carlo Methods and Applications*, 22(1):73–79, 2016. [2.1](#)
- [14] Benedikt Bitterli, Srinath Ravichandran, Thomas Müller, Magnus Wrenninge, Jan Novák, Steve Marschner, and Wojciech Jarosz. A radiative transfer framework for non-exponential media. *ACM Transactions on Graphics (TOG)*, 37(6):225–1, 2018. [\(document\)](#), [4](#), [4.2](#), [4.0.3](#), [5](#), [7](#)
- [15] L. L. Briggs, W. F. Miller, Jr., and E. E. Lewis. Ray-effect mitigation in discrete ordinate-like angular finite element approximations in neutron transport. *Nuclear Science and Engineering*, 57(3):205–217, 1975. doi: 10.13182/NSE75-A26752. URL <https://doi.org/10.13182/NSE75-A26752>. [2.2.3.1](#)
- [16] Thomas A. Brunner. Forms of approximate radiation transport. *Sandia report*, 2002. [3.2.2](#)
- [17] Thomas A. Brunner and James Paul Holloway. Two-dimensional time dependent riemann solvers for neutron transport. *Journal of Computational Physics*, 210(1):386–399, 2005. [3.2.2](#)
- [18] Emanuele Caglioti and François Golse. On the boltzmann-grad limit for the two dimensional periodic lorentz gas. *Journal of Statistical Physics*, 141(2):264–317, 2010. [4](#)
- [19] Thomas Camminady and Martin Frank. A short proof that sweeping is always possible for a spatial discretization with regular triangles and no hanging nodes, 2018. [2.2.5](#)
- [20] Thomas Camminady, Martin Frank, and Edward W. Larsen. Nonclassical particle transport in heterogeneous materials. In *International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering*, 2017. [\(document\)](#), [4.2](#), [5](#), [5.1](#), [5.1](#), [5.2](#), [5.4](#), [6](#)
- [21] Thomas Camminady, Martin Frank, Kerstin Küpper, and Jonas Kusch. Ray effect mitigation for the discrete ordinates method through quadrature rotation. *Journal of Computational Physics*, 382:105–123, August 2019. [\(document\)](#), [3.2](#), [3.2](#), [6](#)
- [22] Thomas Camminady, Martin Frank, and Jonas Kusch. Highly uniform quadrature sets for the discrete ordinates method. In *Proc. Int. Conf. Mathematics and Computational Methods Applied to Nuclear Science and Engineering*, pages 25–29, 2019. [\(document\)](#), [2.2.3](#), [2.2.3.1](#), [6](#)
- [23] Bengt G. Carlson and Clarence E. Lee. *Mechanical quadrature and the transport equation*, volume 2573. Los Alamos Scientific Laboratory of the University of California, 1961. [2.2](#), [2.2.1](#)
- [24] Bengt G. Carlson, Kaye D. Lathrop, et al. *Transport theory: the method of discrete ordinates*. Los Alamos Scientific Laboratory of the University of California, 1965. [2.2.1](#)
- [25] Thomas M. Carsey and Jeffrey J. Harden. *Monte Carlo simulation and resampling methods for social science*. Sage Publications, 2013. [2.1](#)
- [26] Carlo Cercignani et al. *Ludwig Boltzmann: the man who trusted atoms*. Oxford University Press, 1998. [1.3](#)

- [27] Robert D. Cess, M.H. Zhang, P. Minnis, L. Corsetti, E.G. Dutton, B.W. Forgan, D.P. Garber, W.L. Gates, J.J. Hack, E.F. Harrison, et al. Absorption of solar radiation by clouds: Observations versus models. *Science*, 267(5197):496–499, 1995. [4.0.1](#)
- [28] Subrahmanyam Chandrasekhar. On the radiative equilibrium of a stellar atmosphere. II. *The Astrophysical Journal*, 100:76, 1944. [2.2](#)
- [29] Subrahmanyam Chandrasekhar. *Radiative transfer*. Courier Corporation, 2013. [1](#), [2.2](#), [2.2.2](#)
- [30] Jean-François Coulombel, François Golse, and Thierry Goudon. Diffusion approximation and entropy-based moment closure for kinetic equations. *Asymptotic Analysis*, 45(1, 2):1–39, 2005. [1.4.1](#)
- [31] S. Dargaville, A.G. Buchan, R.P. Smedley-Stevenson, P.N. Smith, and C.C. Pain. Scalable angular adaptivity for boltzmann transport. *Journal of Computational Physics*, 406:109124, 2020. ISSN 0021-9991. doi: 10.1016/j.jcp.2019.109124. URL <http://www.sciencedirect.com/science/article/pii/S0021999119308290>. [2.2.3](#), [2.2.6](#), [7](#)
- [32] Anthony B. Davis and Feng Xu. A generalized linear transport model for spatially correlated stochastic media. *Journal of Computational and Theoretical Transport*, 43(1-7):474–514, 2014. doi: 10.1080/23324309.2014.978083. URL <https://doi.org/10.1080/23324309.2014.978083>. [4](#)
- [33] Eugene d’Eon. A reciprocal formulation of nonexponential radiative transfer. 1: Sketch and motivation. *Journal of Computational and Theoretical Transport*, 47(1-3):84–115, 2018. [\(document\)](#), [4](#), [4.3](#), [4.3](#), [4.3](#)
- [34] NASA Earth Observations. *Cloud fraction website*, 2020 (Accessed: 2020-08-05). URL https://neo.sci.gsfc.nasa.gov/view.php?datasetId=MODAL2_M_CLD_FR. [\(document\)](#), [4.1](#)
- [35] Roger Eckhardt. The Monte Carlo method. *Los Alamos Science*, (15):131, 1987. [2.1](#)
- [36] Martin Frank. Lecture Notes Introduction to Kinetic Theory, 2018. [1](#), [1.4.1](#), [1.6](#), [1.6](#)
- [37] Martin Frank, Jonas Kusch, Thomas Camminady, and Cory D. Hauck. Ray effect mitigation for the discrete ordinates method using artificial scattering. *Nuclear Science and Engineering*, pages 1–18, 2020. [\(document\)](#), [3.3](#), [3.3](#), [6](#)
- [38] Baray D. Ganapol, Randall S. Baker, Jon A. Dahl, and Raymond E. Alcouffe. Homogeneous infinite media time-dependent analytical benchmarks. Technical report, Los Alamos National Laboratory, 2001. [3.2.1](#)
- [39] Barry D. Ganapol, P.W. McKenty, and K.L. Peddicord. The generation of time-dependent neutron transport solutions in infinite media. *Nuclear Science and Engineering*, 64(2):317–331, 1977. [3.1](#)
- [40] Alan E. Gelfand and Adrian F. M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409, 1990. doi: 10.1080/01621459.1990.10476213. URL <https://amstat.tandfonline.com/doi/abs/10.1080/01621459.1990.10476213>. [2.1.1.3](#)

- [41] François Golse. The periodic lorentz gas in the boltzmann-grad limit. In *Proc. ICM*, pages 183–201, 2006. [4](#), [4.1](#)
- [42] François Golse. On the periodic lorentz gas and the lorentz kinetic equation. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 17, pages 735–749, 2008. [4.1](#)
- [43] Harold Grad. Principles of the kinetic theory of gases. In *Thermodynamik der Gase/Thermodynamics of Gases*, pages 205–294. Springer, 1958. [1](#), [1.3](#)
- [44] Brian L. Hammond, William A. Lester, and Peter James Reynolds. *Monte Carlo methods in ab initio quantum chemistry*, volume 1. World Scientific, 1994. [2.1](#)
- [45] Gregory R. Hancock, Laura M. Stapleton, and Ralph O. Mueller. *The reviewer’s guide to quantitative methods in the social sciences*. Routledge, 2018. [2.1](#)
- [46] Louis G. Henyey and Jesse L. Greenstein. Diffuse radiation in the galaxy. *The Astrophysical Journal*, 93:70–83, 1941. [1.4](#)
- [47] Peter Jäckel. *Monte Carlo methods in finance*, volume 71. J. Wiley, 2002. [2.1](#)
- [48] Adrian Jarabo, Carlos Aliaga, and Diego Gutierrez. A radiative transfer framework for spatially-correlated materials. *ACM Transactions on Graphics (TOG)*, 37(4):83, 2018. [4](#), [4.3](#), [4.3](#), [5](#), [7](#)
- [49] Joshua J. Jarrell and Marvin L. Adams. Discrete-ordinates quadrature sets based on linear discontinuous finite elements. 2011. [2.2.3.1](#)
- [50] Corwin Joy, Phelim P. Boyle, and Ken Seng Tan. Quasi-monte carlo methods in numerical finance. *Management Science*, 42(6):926–938, 1996. [2.1](#)
- [51] Jungchung Jung, Hiroshi Chijiwa, Keisuke Kobayashi, and Hiroshi Nishihara. Discrete ordinate neutron transport equation equivalent to pl approximation. *Nuclear Science and Engineering*, 49(1):1–9, 1972. [3](#)
- [52] A. B. Kahn. Topological sorting of large networks. *Commun. ACM*, 5(11):558–562, November 1962. ISSN 0001-0782. doi: 10.1145/368996.369025. URL <https://doi.org/10.1145/368996.369025>. [2.2.5](#)
- [53] M. D. King, S. Platnick, W. P. Menzel, S. A. Ackerman, and P. A. Hubanks. Spatial and temporal distribution of clouds observed by modis onboard the terra and aqua satellites. *IEEE Transactions on Geoscience and Remote Sensing*, 51(7):3826–3852, 2013. [1](#)
- [54] Kerstin Küpper. *Models, numerical methods, and uncertainty quantification for radiation therapy*. PhD thesis, RWTH Aachen University, 2016. [2.3.1](#)
- [55] Jonas Kusch. *Realizability-preserving discretization strategies for hyperbolic and kinetic equations with uncertainty*. PhD thesis, Karlsruhe Institute of Technology, may 2020. URL <https://doi.org/10.5445/IR/1000121168>. (document), [3.2](#), [3.3](#), [3.3](#)
- [56] Edward W. Larsen and Richard Vasques. A generalized linear boltzmann equation for non-classical particle transport. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 112(4):619–631, 2011. [4](#), [4.0.2](#), [4.2](#), [4.3](#), [4.3](#)

- [57] Edward W. Larsen, Martin Frank, and Thomas Camminady. The equivalence of 'forward' and 'backward' nonclassical particle transport theories. *Proceedings M&C2017, Korea, April*, 2017. [4.2](#), [4.2](#), [6](#)
- [58] Kaye D. Lathrop. Remedies for ray effects. *Nuclear Science and Engineering*, 45(3):255–268, 1971. [3](#), [3.3](#)
- [59] Kaye D. Lathrop. The early days of the SN method. *Trans. Am. Nucl. Soc*, 66:241–242, 1992. [3](#)
- [60] Kaye D. Lathrop and Bengt G. Carlson. Discrete ordinates angular quadrature of the neutron transport equation. Technical report, Los Alamos Scientific Lab., N. Mex., 1964. [2.2.1](#), [2.2.6](#), [3](#)
- [61] Cheuk Y. Lau and Marvin L. Adams. Discrete ordinates quadratures based on linear and quadratic discontinuous finite elements over spherical quadrilaterals. *Nuclear Science and Engineering*, 185(1):36–52, 2017. doi: 10.13182/NSE16-28. URL <https://doi.org/10.13182/NSE16-28>. [2.2.3.1](#)
- [62] Randall J. LeVeque. *Numerical methods for conservation laws*, volume 3. Springer, 1992. [1.7](#)
- [63] C. David Levermore. Moment closure hierarchies for kinetic theories. *Journal of statistical Physics*, 83(5-6):1021–1065, 1996. [2.3.2](#)
- [64] Elmer Eugene Lewis and Warren F. Miller. *Computational methods of neutron transport*. Wiley, 1984. [1](#), [2.2.1](#), [2.2.5](#), [2.2.5](#)
- [65] N.K. Madsen. Pointwise convergence of the three-dimensional discrete ordinate method. *SIAM Journal on Numerical Analysis*, 8(2):266–269, 1971. [2.2](#)
- [66] Bryan F.J. Manly. *Randomization, bootstrap and Monte Carlo methods in biology*, volume 70. CRC press, 2006. [2.1](#)
- [67] George Marsaglia et al. Xorshift rngs. *Journal of Statistical Software*, 8(14):1–6, 2003. [2.1.1](#)
- [68] Makoto Matsumoto and Takuji Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 8(1):3–30, 1998. [2.1.1](#)
- [69] Ryan G. McClarren and Cory D. Hauck. Robust and accurate filtered spherical harmonics expansions for radiative transfer. *Journal of Computational Physics*, 229(16):5597–5614, 2010. [2.3.2](#), [3.1](#)
- [70] N. Metropolis. The beginning of the monte carlo method. *Los Alamos Science*, 15:125–30, 1987. [2.1](#)
- [71] W. F. Miller, Jr. and Wm. H. Reed. Ray-effect mitigation methods for two-dimensional neutron transport theory. *Nuclear Science and Engineering*, 62(3):391–411, 1977. doi: 10.13182/NSE62-391. URL <https://doi.org/10.13182/NSE62-391>. [3](#)
- [72] Charles J. Mode. *Applications of Monte Carlo methods in biology, medicine and other fields of science*. BoD–Books on Demand, 2011. [2.1](#)

- [73] J. E. Morel, T. A. Wareing, R. B. Lowrie, and D. K. Parsons. Analysis of ray-effect mitigation techniques. *Nuclear Science and Engineering*, 144(1):1–22, 2003. doi: 10.13182/NSE01-48. 3
- [74] Claus Müller. *Spherical harmonics*, volume 17. Springer, 2006. 1.4
- [75] Harald Niederreiter. *Random number generation and quasi-Monte Carlo methods*. SIAM, 1992. 2.1.1.3
- [76] M. Peter Nightingale and Cyrus J. Umrigar. *Quantum Monte Carlo methods in physics and chemistry*. Number 525. Springer Science & Business Media, 1998. 2.1
- [77] Harald Paganetti, H. Jiang, S.-Y. Lee, and H.M. Kooy. Accurate monte carlo simulations for nozzle design, commissioning and quality assurance for a proton radiation therapy facility. *Medical physics*, 31(7):2107–2118, 2004. 2.1
- [78] François Panneton, Pierre L’ecuyer, and Makoto Matsumoto. Improved long-period generators based on linear recurrences modulo 2. *ACM Transactions on Mathematical Software (TOMS)*, 32(1):1–16, 2006. 2.1.1
- [79] K. Pfeilsticker, F. Erle, O. Funk, H. Veitel, and U. Platt. First geometrical pathlengths probability density function derivation of the skylight from spectroscopically highly resolving oxygen a-band observations: 1. measurement technique, atmospheric observations and model calculations. *Journal of Geophysical Research: Atmospheres*, 103(D10):11483–11504, 1998. 4.0.1
- [80] Peter Pilewskie and Francisco P.J. Valero. Direct observations of excess solar absorption by clouds. *Science*, 267(5204):1626–1629, 1995. 4.0.1
- [81] G.C. Pomraning. The fokker-planck operator as an asymptotic limit. *Mathematical Models and Methods in Applied Sciences*, 2(01):21–36, 1992. 3.3.2, 3.3.2, 3.3.2
- [82] Anil K. Prinja and Edward W. Larsen. General principles of neutron transport. In *Handbook of nuclear engineering*. 2010. 1, 1.5
- [83] V. Ramanathan, B. Subasilar, G.J. Zhang, W. Conant, R.D. Cess, J.T. Kiehi, H. Grassi, and L. Shi. Warm pool heat budget and shortwave cloud forcing: A missing physics? *Science*, 267(5197):499–503, 1995. 4.0.1
- [84] W. H. Reed. Spherical harmonic solutions of the neutron transport equation from discrete ordinate codes. *Nuclear Science and Engineering*, 49(1):10–19, 1972. 3
- [85] Daniel Rieger, M. Bangert, I. Bischoff-Gauss, J. Förstner, K. Lundgren, D. Reinert, J. Schröter, H. Vogel, G. Zängl, R. Ruhnke, et al. Icon-art 1.0—a new online-coupled model system from the global to regional scale. *Geoscientific Model Development*, 8(6):1659–1676, 2015. 2.2.3
- [86] Richard Sanchez. On SN-PN Equivalence. *Transport Theory and Statistical Physics*, 41(5-6):418–447, 2012. doi: 10.1080/00411450.2012.672360. URL <https://doi.org/10.1080/00411450.2012.672360>. 3.3
- [87] Joao Seco and Frank Verhaegen. *Monte Carlo techniques in radiation therapy*. CRC press, 2013. 2.1

- [88] Hyung Jin Shim, Beom Seok Han, Jong Sung Jung, Ho Jin Park, Chang Hyo Kim, et al. Mccard: Monte carlo code for advanced reactor design and analysis. *Nucl. Eng. Technol.*, 44(2):161, 2012. [2.1](#)
- [89] Ralph C. Smith. *Uncertainty quantification: theory, implementation, and applications*, volume 12. Siam, 2013. [2.1.1.1](#)
- [90] Laurent Soucasse, Steven Dargaville, Andrew G. Buchan, and Christopher C. Pain. A goal-based angular adaptivity method for thermal radiation modelling in non grey media. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 200:215–224, 2017. ISSN 0022-4073. doi: 10.1016/j.jqsrt.2017.06.015. URL <http://www.sciencedirect.com/science/article/pii/S0022407317302352>. [2.2.3](#), [2.2.6](#)
- [91] Herbert Spohn. *Large scale dynamics of interacting particles*. Springer Science & Business Media, 2012. [1.4.2](#)
- [92] John Tencer. Ray effect mitigation through reference frame rotation. *Journal of Heat Transfer*, 138(11), 2016. [3](#)
- [93] CP Thurgood, A Pollard, and HA Becker. The tn quadrature set for the discrete ordinates method. *Journal of heat transfer*, 117(4):1068–1070, 1995. [2.2.3](#), [2.2.3.1](#)
- [94] Lloyd N. Trefethen. Is gauss quadrature better than clenshaw–curtis? *SIAM Review*, 50(1):67–87, 2008. doi: 10.1137/060659831. URL <https://doi.org/10.1137/060659831>. [2.2.2](#)
- [95] Lloyd N. Trefethen. Cubature, approximation, and isotropy in the hypercube. *SIAM Review*, 59(3):469–491, 2017. [2.1.3](#)
- [96] S. Ulam, R.D. Richtmyer, and J. Von Neumann. Statistical methods in neutron diffusion. *LAMS-551, Los Alamos National Laboratory*, pages 1–22, 1947. [2.1](#)
- [97] Richard Vasques and Edward W. Larsen. Anisotropic diffusion in model 2-d pebble-bed reactor cores. In *Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications, American Nuclear Society*, 2009. [4.0.2](#)
- [98] Richard Vasques and Edward W. Larsen. Non-classical particle transport with angular-dependent path-length distributions. i: Theory. *Annals of Nuclear Energy*, 70:292–300, 2014. [4.0.2](#)
- [99] Frank Verhaegen and Jan Seuntjens. Monte carlo modelling of external radiotherapy photon beams. *Physics in medicine & biology*, 48(21):R107, 2003. [2.1](#)
- [100] Gian Carlo Wick. Über ebene diffusionsprobleme. *Zeitschrift für Physik*, 121(11-12):702–718, 1943. [2.2](#)
- [101] Martin Wild, Atsumu Ohmura, Hans Gilgen, and Erich Roeckner. Validation of general circulation model radiative fluxes using surface observations. *Journal of climate*, 8(5):1309–1324, 1995. [4.0.1](#)
- [102] Yican Wu, Jing Song, Huaqing Zheng, Guangyao Sun, Lijuan Hao, Pengcheng Long, Liqin Hu, FDS Team, et al. Cad-based monte carlo program for integrated simulation of nuclear system supermc. *Annals of Nuclear Energy*, 82:161–168, 2015. [2.1](#)