Iterative Methods for Criticality Computations in Neutron Transport Theory

submitted by

Fynn Scheben

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Mathematical Sciences

January 2011

Signature of Author ....

Fynn Scheben

Summary

This thesis studies the so-called "criticality problem", an important generalised eigenvalue problem arising in neutron transport theory. The smallest positive real eigenvalue of the problem contains valuable information about the status of the fission chain reaction in the nuclear reactor (i.e. the criticality of the reactor), and thus plays an important role in the design and safety of nuclear power stations. Because of the practical importance, efficient numerical methods to solve the criticality problem are needed, and these are the focus of this thesis.

In the theory we consider the time-independent neutron transport equation in the monoenergetic homogeneous case with isotropic scattering and vacuum boundary conditions. This is an unsymmetric integro-differential equation in 5 independent variables, modelling transport, scattering, and fission, where the dependent variable is the neutron angular flux. We show that, before discretisation, the non-symmetric eigenproblem for the angular flux is equivalent to a related eigenproblem for the scalar flux, involving a symmetric positive definite weakly singular integral operator (in space only). Furthermore, we prove the existence of a simple smallest positive real eigenvalue with a corresponding eigenfunction that is strictly positive in the interior of the reactor. We discuss approaches to discretise the problem and present discretisations that preserve the underlying symmetry in the finite dimensional form.

The thesis then describes methods for computing the criticality in nuclear reactors, i.e. the smallest positive real eigenvalue, which are applicable for quite general geometries and physics. In engineering practice the criticality problem is often solved iteratively, using some variant of the inverse power method. Because of the high dimension, matrix representations for the operators are often not available and the inner solves needed for the eigenvalue iteration are implemented by matrix-free inner iterations. This leads to inexact iterative methods for criticality computations, for which there appears to be no rigorous convergence theory. The fact that, under appropriate assumptions, the integro-differential eigenvalue problem possesses an underlying symmetry (in a space of reduced dimension) allows us to perform a systematic convergence analysis for inexact inverse iteration and related methods. In particular, this theory provides rather precise criteria on how accurate the inner solves need to be in order for the whole iterative method to converge. The theory is illustrated with numerical examples on several test problems of physical relevance, using GMRES as the inner solver.

We also illustrate the use of Monte Carlo methods for the solution of neutron transport source problems as well as for the criticality problem. Links between the steps in the Monte Carlo process and the underlying mathematics are emphasised and numerical examples are given. Finally, we introduce an iterative scheme (the so-called "method of perturbation") that is based on computing the difference between the solution of the problem of interest and the known solution of a base problem. This situation is very common in the design stages for nuclear reactors when different materials are tested, or the material properties change due to the burn-up of fissile material.

We explore the relation of the method of perturbation to some variants of inverse iteration, which allows us to give convergence results for the method of perturbation. The theory shows that the method is guaranteed to converge if the perturbations are not too large and the inner problems are solved with sufficiently small tolerances. This helps to explain the divergence of the method of perturbation in some situations which we give numerical examples of. We also identify situations, and present examples, in which the method of perturbation achieves the same convergence rate as standard shifted inverse iteration. Throughout the thesis further numerical results are provided to support the theory.

Acknowledgements

It is a pleasure to thank the many people that have helped me, in various ways, to produce this thesis.

First and foremost, and the very biggest thanks, go to my supervisor Ivan Graham at the University of Bath. His guidance, patience, confidence, and incredible knowledge has helped tremendously during my PhD. Despite the huge amount of work on his plate, Ivan always made time for a meeting and managed to check my work. Ivan, I have been incredibly fortunate to have you as a supervisor and I am deeply grateful for all the support you have given me – with this project and beyond.

A huge thank you goes to my industrial supervisor Paul Smith at Serco (Technical and Assurance Services). Paul always made me feel most welcome during the numerous visits at Winfrith, found time to answer my never-ending lists of questions and to come for meetings to Bath. He even took the time to show me the beautiful Dorset coast and to invite me to his house for a proper barbecue. Paul, I will miss the long discussions on your whiteboard and "tracking all these neutrons".

I am also extremely grateful to the other members of the ANSWERS team at Serco in Winfrith. In particular, a big thank you to Tim Newton, Brian Turland, Les Hutton and Glynn Hosking for the many helpful discussions. Also thanks to Adam Bird, Dave Powney, Pete Smith, Malcolm Armishaw and Chris Dean for explaining the basics of neutron transport theory to me and answering my many beginner's questions. A special thank you goes to Caroline Middlemas who always found some money and ways to allow me to take part in the industrial conferences and courses at Serco.

Furthermore, I would like to thank John Toland for fruitful discussions, brushing up my functional analysis knowledge and helping me with operator theory on cones leading to the proof of the simplicity of the smallest eigenvalue in Section 2.3, as well as Tom Manteuffel and Steve McCormick for inviting me to Denver and giving me helpful feedback on my work.

Big thanks must go to the brilliant Numerical Analysis group in Bath for all the seminars and, in particular, to Alastair Spence for the many good suggestions during, and after, seminar and report presentations. Thank you also to the staff in the Department of Mathematical Sciences, in particular Jill Parker, Carole Negre, Mary Baines, Sarah Hardy, Eric Wing and Jon Elmes. In addition I am much obliged to the EPSRC, the University of Bath and Serco for the funding of this project.

I would now like to thank some very special friends that helped me with this thesis; Melina for sharing her knowledge on iterative methods in many useful discussions, Curdin for the private lessons in probability, Richard and Euan for helping me with my English and proofreading big parts of the thesis, and Ray and Tania for answering the many daily questions and keeping the motivation up.

Thank you to David and Tom, PhD students at the University of Nottingham, for joining me on this journey into neutron transport theory and the many discussions during workshops at the OECD in Paris and meetings, seminars, and conferences in the UK. For sharing an office and putting up with me (and my sometimes smelly sports clothes), as well as many enjoyable breaks from work, I would like to thank Giampiero, Natasha, Dave, Jane, Adam and Haojie. Big thanks also to Geoff, Ray, Matt, Tom, Elvijs, James, Phil, Sean, Adam K., and many other fellow PhD students for organising the social life in Bath and for sometimes pulling me away from my desk. Thanks also to the departmental football team for all the hours of exercise and league titles, as well as great trips to the tournaments in Germany and the help to organise (and hopefully establish) the "UK Maths Postgrad Football Championships".

Last, but certainly not least, I would like to thank Janet and my family for the continuous encouragement, unlimited patience and daily support. This thesis is dedicated to you.

Contents

Chapter 1

Introduction

Climate change and ensuring a reliable energy supply are challenging problems of great contemporary interest. It is still open to debate whether or not nuclear power is part of a solution to these problems, but certainly ensuring the safety and optimal performance of new, as well as existing, nuclear reactors is an important task of considerable environmental and economic significance. When operating a nuclear reactor, the engineer seeks to achieve a sustainable chain reaction where the neutrons produced by fission balance the neutrons that are either absorbed or leave the reactor through the outer boundary. The chain reaction depends on the material composition and geometry of the reactor and can be controlled by inserting or removing control rods.

Neutron transport theory is of crucial importance to nuclear engineers since it describes the mechanism by which a stable and continuous fission reaction can be maintained to generate heat in a safe and controlled manner. As the cost of building nuclear reactors is high, the task of accurately modelling neutron transport and solving the resulting problems is essential for designing new nuclear power stations. Also, assurance tests to model potential accidents in existing plants and to ensure the safety of waste repositories rely on efficient numerical methods for solving transport problems.

By requiring a balance between the loss and gain of neutrons in the neutron transport equation an eigenvalue problem can be formulated. The smallest positive real eigenvalue of this problem gives information about the change in the number of neutrons in the reactor, a value that determines the safety of the nuclear reactor. In this thesis we discuss iterative methods to compute this eigenvalue.

This dissertation is the result of a PhD project at the University of Bath in cooperation with Serco Technical and Assurance Services in Winfrith, Dorset, United Kingdom.

Outline of the introduction

In the first section of this introduction we introduce the linear Boltzmann equation that governs the distribution of neutrons and photons. By considering the example of a nuclear reactor we explain the physical meaning of the operators and functions involved before motivating how the solution of an eigenvalue problem provides vital information about the safety of the reactor (Section 1.2).

Sections 1.3 and 1.4 describe common boundary conditions for the neutron transport equation and discuss the adjoint problem which we will use in Chapter 5. In Section 1.5 we present a brief overview of solution methods for source problems before giving a short review of neutron transport literature. In the final section of the first chapter the contributions of this thesis to the research community are summarised and an outline of the remainder of the thesis is given.

1.1 Neutron transport equation

The *neutron transport equation* (also referred to as *linear Boltzmann equation*) describes the distribution of neutrons in terms of their positions in space and time, their energies and their travel directions. It can be derived by considering the change in the particle distribution in time using the *neutron density* $N(\mathbf{r}, E, \mathbf{\Omega}, t)$, which denotes the number of neutrons per unit volume with kinetic energy $E \in \mathbb{R}^+$ at position $\mathbf{r} \in \mathbb{R}^3$ that move at time $t \in \mathbb{R}^+$ in direction $\mathbf{\Omega} \in \mathbb{S}^2$ (where $\mathbb{S}^2$ is the unit sphere in $\mathbb{R}^3$). The motivation of the linear Boltzmann equation appears in many books (e.g. [20, 24, 29, 80]), with a particularly good descriptive derivation in [12, §1.1c].

Usually, the physical quantity that is used to state the linear Boltzmann equation is not the neutron density N, but the *(neutron) angular flux* Ψ. This is related to the neutron density by

$$\Psi(\mathbf{r}, E, \mathbf{\Omega}, t) \;=\; v(E) N(\mathbf{r}, E, \mathbf{\Omega}, t) \;,$$

where the neutron speed v is defined implicitly via $E = \frac{1}{2} m v^2$.

In order to derive the governing equation, several assumptions on the physics are made (see [80, p. 3]), which are therefore also present in this thesis. In particular, we consider all neutrons as point particles and suppose that they travel in straight lines between collisions. Furthermore, due to their small size, neutron-neutron interactions are neglected and we assume that all material properties are known.

The time-dependent form of the neutron transport equation is then (see e.g. [80, (1-16),(1-94)])

$$\begin{aligned}\frac{1}{v}\frac{\partial\Psi}{\partial t}(\mathbf{r},E,\mathbf{\Omega},t) \;=\; & -\,\mathbf{\Omega}\cdot\nabla\Psi(\mathbf{r},E,\mathbf{\Omega},t) \;-\; \sigma(\mathbf{r},E)\Psi(\mathbf{r},E,\mathbf{\Omega},t) \\ & +\; \frac{1}{4\pi}\int_{\mathbb{R}^+}\int_{\mathbb{S}^2}\sigma_s(\mathbf{r},E',E,\mathbf{\Omega}',\mathbf{\Omega})\Psi(\mathbf{r},E',\mathbf{\Omega}',t)\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}E' \\ & +\; \frac{\chi(E)}{4\pi}\int_{\mathbb{R}^+}\nu(\mathbf{r},E')\sigma_f(\mathbf{r},E')\int_{\mathbb{S}^2}\Psi(\mathbf{r},E',\mathbf{\Omega}',t)\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}E' \\ & +\; Q(\mathbf{r},E,\mathbf{\Omega},t)\ ,\end{aligned} \tag{1.1}$$

where

$\mathbb{R}^+$	non-negative real numbers,
$\mathbf{r}=(x,y,z)^T\in V\subset\mathbb{R}^3$	position in volume V (the reactor),
$E\in\mathbb{R}^+$	energy,
$\mathbf{\Omega}\in\mathbb{S}^2=\{\mathbf{\Omega}\in\mathbb{R}^3:\lVert\mathbf{\Omega}\rVert_2=1\}$	direction,
$t\in\mathbb{R}^+$	time,
$\Psi(\mathbf{r},E,\mathbf{\Omega},t)\in\mathbb{R}^+$	neutron flux of energy E at position $\mathbf{r}$ in direction $\mathbf{\Omega}$ at time t,
$\sigma(\mathbf{r},E)\in\mathbb{R}^+$	total cross-section at position $\mathbf{r}$ for energy E,
$\sigma_s(\mathbf{r},E',E,\mathbf{\Omega}',\mathbf{\Omega})\in\mathbb{R}^+$	scatter cross-section at position $\mathbf{r}$ from energy E' to energy E and from direction $\mathbf{\Omega}'$ to direction $\mathbf{\Omega}$,
$\sigma_f(\mathbf{r},E')\in\mathbb{R}^+$	fission cross-section at position $\mathbf{r}$ for energy E',
$\nu(\mathbf{r},E')\in\mathbb{R}^+$	neutron yield from fission events in energy E' at position $\mathbf{r}$,
$\chi(E)\in\mathbb{R}^+$	resulting fission neutron distribution in energy E, and
$Q(\mathbf{r},E,\mathbf{\Omega},t)\in\mathbb{R}^+$	non-fission sources of neutrons with energy E and direction $\mathbf{\Omega}$ at position $\mathbf{r}$ and time t.

As mentioned above, the equation (1.1) describes the flux Ψ at a certain position, energy, direction, and time, and can be derived by considering how the neutron flux changes with time. This change in time is determined by taking the difference between neutrons gained and neutrons lost. The gain of neutrons is represented by the three positive terms on the right-hand side of (1.1), while the loss of neutrons is represented by the two negative terms. We now give a brief description of the terms involved.

The first two terms on the right-hand side of (1.1) describe the neutron loss occurring due to streaming (denoted by $\mathbf{\Omega}\cdot\nabla\Psi$) and neutrons having a collision with the underlying nuclei ($\sigma\Psi$). When a collision occurs, the neutron may be *captured*, *scatter*, or it may cause a *fission*. In any of these cases the collided neutron no longer travels in the same direction $\mathbf{\Omega}$ with the same energy E and is therefore lost from the considered angular flux $\Psi(\mathbf{r}, E, \mathbf{\Omega}, t)$. The probability that neutrons have any kind of collision is described by the *total cross-section* σ.

The third term on the right of (1.1) accounts for the increase in the flux Ψ when neutrons are scattered from a previously different direction $\mathbf{\Omega}'$ and energy level E' into the direction $\mathbf{\Omega}$ and energy E that we consider. The probability for this is determined by the *scatter cross-section* σ_s. In order to account for the in-scatter of neutrons from all possible other energy levels and directions we integrate over these ranges and obtain the scattering contribution

$$\frac{1}{4\pi} \int_{\mathbb{R}+} \int_{\mathbb{S}^2} \sigma_s(\mathbf{r}, E', E, \mathbf{\Omega}', \mathbf{\Omega}) \Psi(\mathbf{r}, E', \mathbf{\Omega}', t) \, \mathrm{d}\mathbf{\Omega}' \, \mathrm{d}E' \ .$$

A common simplification of the problem is the assumption of *isotropic* scattering. In this case, the scatter cross-section is independent of the incoming and outgoing direction, resulting in $\sigma_s(\mathbf{r}, E', E, \mathbf{\Omega}', \mathbf{\Omega}) = \sigma_s(\mathbf{r}, E', E)$.

The fourth term represents the gain of neutrons from fission events. The *fission cross-section* σ_f is independent of the incoming and outgoing directions as it does not matter for a fission from which direction the nucleus is hit and for the emerging neutrons all directions are equally likely. However, the speed of the incident neutron does affect the probability of a fission occurring and also impacts ν, the number of neutrons produced in a fission event. This results in an energy dependence of ν and σ_f.

Finally, the distribution of the emerging fission neutrons in energy is described by χ. Hence the total number of neutrons entering the angular flux in direction $\mathbf{\Omega}$ with energy E at time t due to fission events is given by

$$\frac{\chi(E)}{4\pi} \int_{\mathbb{R}+} \nu(\mathbf{r}, E') \sigma_f(\mathbf{r}, E') \int_{\mathbb{S}^2} \Psi(\mathbf{r}, E', \mathbf{\Omega}', t) \, \mathrm{d}\mathbf{\Omega}' \, \mathrm{d}E' \ .$$

Lastly, the fifth term on the right-hand side of (1.1) contains neutrons that are gained from non-fission sources such as radioactive materials that decay and emit neutrons without a neutron-nucleus collision. These are represented by the external source Q. Combining all of these terms to describe the change in neutron flux with respect to time gives the linear Boltzmann equation (1.1).

In addition, the cross-sections in the equation are related by

$$\sigma(\mathbf{r}, E) \ = \ \sigma_c(\mathbf{r}, E) \ + \ \frac{1}{4\pi} \int_{\mathbb{R}+} \int_{\mathbb{S}^2} \sigma_s(\mathbf{r}, E, E', \mathbf{\Omega}, \mathbf{\Omega}') \, \mathrm{d}\mathbf{\Omega}' \, \mathrm{d}E' \ + \ \sigma_f(\mathbf{r}, E) \ , \qquad (1.2)$$

where $\sigma_c(\mathbf{r}, E)$ denotes the *capture cross-section* for neutrons of energy E at position $\mathbf{r}$. Note that here we integrate the scatter cross-section over the *outgoing* energies and directions.

1.2 Criticality problem

When operating a nuclear reactor, the aim of the engineer is to achieve a controlled sustainable chain reaction where the number of neutrons that are produced is equal to the number of neutrons that are absorbed or leave the system through the outer boundary. The reactor is started by introducing a source Q that releases neutrons due to spontaneous radioactive decay. These neutrons then cause fission events which result in further neutrons being produced.

The chain reaction depends on the material composition and geometry of the nuclear reactor and can be controlled by inserting and removing control rods. The goal of nuclear reactor design is to obtain a neutron distribution that does not change with time after the initial start-up period. A reactor, where the gain and loss of neutrons is exactly balanced, is called *critical.* When the reactor reaches this operating stage, the contribution of neutrons from non-fission sources is negligible.

We will focus on this situation by setting $\partial\Psi/\partial t$ and the non-fission neutron contribution Q to zero and therefore enforcing a balance between the loss and gain of neutrons. In this case the *time-independent neutron transport equation* is, for suitable ranges of $\mathbf{r}$, E and $\boldsymbol{\Omega}$,

$$\begin{aligned} & \boldsymbol{\Omega} \cdot \nabla\Psi(\mathbf{r}, E, \boldsymbol{\Omega}) \; + \; \sigma(\mathbf{r}, E)\Psi(\mathbf{r}, E, \boldsymbol{\Omega}) \\ = \; & \frac{1}{4\pi} \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \sigma_s(\mathbf{r}, E', E, \boldsymbol{\Omega}', \boldsymbol{\Omega})\Psi(\mathbf{r}, E', \boldsymbol{\Omega}') \, \mathrm{d}\boldsymbol{\Omega}' \, \mathrm{d}E' \\ + \; & \frac{\chi(E)}{4\pi} \int_{\mathbb{R}^+} \nu(\mathbf{r}, E')\sigma_f(\mathbf{r}, E') \int_{\mathbb{S}^2} \Psi(\mathbf{r}, E', \boldsymbol{\Omega}') \, \mathrm{d}\boldsymbol{\Omega}' \, \mathrm{d}E' \; . \end{aligned} \tag{1.3}$$

The left-hand side of (1.3) is referred to as the transport part of the equation while the first term on the right contains the scatter contribution and the second term represents neutrons from fission events. We will discuss boundary conditions for the problem (1.3) in the next section.

Introducing the transport, scatter and fission operators

$$\begin{aligned}
\mathcal{T}\Psi &= \mathbf{\Omega}\cdot\nabla\Psi(\mathbf{r},E,\mathbf{\Omega}) \;+\; \sigma(\mathbf{r},E)\Psi(\mathbf{r},E,\mathbf{\Omega}) \;, \\
\mathcal{S}\Psi &= \frac{1}{4\pi}\int_{\mathbb{R}+}\int_{\mathbb{S}^2}\sigma_s(\mathbf{r},E',E,\mathbf{\Omega}',\mathbf{\Omega})\Psi(\mathbf{r},E',\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}E' \;, \quad \text{and} \\
\mathcal{F}\Psi &= \frac{\chi(E)}{4\pi}\int_{\mathbb{R}+}\nu(\mathbf{r},E')\sigma_f(\mathbf{r},E')\int_{\mathbb{S}^2}\Psi(\mathbf{r},E',\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}E' \;,
\end{aligned}$$

the problem (1.3) can be written in operator form as

$$\mathcal{T}\Psi \;=\; \mathcal{S}\Psi \;+\; \mathcal{F}\Psi \;. \tag{1.4}$$

By arriving at the steady-state equation (1.3) we assumed that we obtained a reactor where the loss and gain of neutrons is exactly balanced. This is the objective of reactor design but for an arbitrary configuration this will not be the case and the only solution to the homogeneous problem (1.4) is in general the zero function. In order to evaluate "how far away from criticality" the reactor is, the problem is transformed into an eigenvalue problem.

We introduce a scalar λ in front of the fission term and search for the smallest real eigenvalue $\lambda > 0$ and the corresponding eigenfunction Ψ to satisfy the resulting generalised eigenvalue problem

$$(\mathcal{T}-\mathcal{S})\Psi \;=\; \lambda\mathcal{F}\Psi \;. \tag{1.5}$$

This is the *criticality problem* and its smallest positive real eigenvalue, which we call the principal eigenvalue or fundamental mode, has a physical meaning and gives direct information about the criticality of the system. If $\lambda > 1$, we have to artificially increase the fission contribution to find a non-zero solution to (1.4) which means that the problem is *subcritical*. If $\lambda < 1$, it is *supercritical*, and if $\lambda = 1$, the reactor is in the desired *critical* state. Hence, the solution of the eigenvalue problem can be used by nuclear engineers to test and optimise the design of nuclear reactors.

Note that in this thesis we are not concerned about the optimisation part of the process but our primary objective is to obtain and analyse efficient numerical methods to compute the principal eigenvalue λ for any given reactor. In the terminology for designing a critical reactor, it is the "forward problem" that we are aiming to solve here, rather than the "inverse problem" of designing the reactor to optimise λ.

In the literature, and especially in industry, the reciprocal $k_{\text{eff}} := 1/\lambda$ is frequently used instead of λ. For other eigenvalue problems in neutron transport theory (such as the search for time-eigenvalues) see, for example, [80, §1.5].

In this thesis we consider a class of monoenergetic homogeneous model problems with isotropic scattering for which the operators $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{F}$ obtain a simpler structure than in the energy-dependent heterogeneous case (1.3). We introduce the model problems in Section 2.1 and prove for these in Section 2.3 that the criticality problem is well-defined, i.e. that a smallest positive real eigenvalue with non-negative eigenfunction exists. In Chapters 3, 4 and 5 we then discuss iterative methods to compute the criticality and consider their convergence.

1.3 Boundary conditions

In nuclear engineering several different boundary conditions are used. The two most common ones are "vacuum" and "reflective" boundary conditions.

The *vacuum boundary conditions* are often referred to as zero (incoming) flux boundary conditions since they demand that the angular flux on the boundary is zero for all incoming directions, i.e.

$$\Psi(\mathbf{r}, E, \mathbf{\Omega}) \;=\; 0 \quad \text{when} \quad \mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r}) \;<\; 0\;, \quad \mathbf{r}\in\partial V\;, \tag{1.6}$$

where $\mathbf{n}(\mathbf{r})$ denotes the outward normal at $\mathbf{r} \in \partial V$, the boundary of V (see Figure 1.1). We assume throughout that V is a convex domain and has a C^1-boundary.

The *reflective boundary conditions* state that the flux on the boundary in the incoming direction $\mathbf{\Omega}$ is the same as the flux at that boundary point in direction $\mathbf{\Omega}'$, where $\mathbf{\Omega}' = \mathbf{\Omega} - 2[\mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r})]\mathbf{n}(\mathbf{r})$ is the reflected direction (see Figure 1.1) and satisfies

$$\mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r}) \;=\; -\mathbf{\Omega}'\cdot\mathbf{n}(\mathbf{r}) \quad \text{and} \quad (\mathbf{\Omega}\times\mathbf{\Omega}')\cdot\mathbf{n}(\mathbf{r}) \;=\; 0\;.$$

The reflective boundary conditions can thus be stated as

$$\Psi(\mathbf{r}, E, \mathbf{\Omega}) \;=\; \Psi(\mathbf{r}, E, \mathbf{\Omega}') \quad \text{when} \quad \mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r}) \;<\; 0\;, \quad \mathbf{r}\in\partial V\;. \tag{1.7}$$

Other (less common) boundary conditions are "translational" (also called "periodic") and "rotational" boundary conditions, as well as "white" boundary conditions, where the outgoing flux is reflected isotropically (with an equal distribution in angle) back into the system. These, and their numerical treatment, are discussed, for example, in [102].

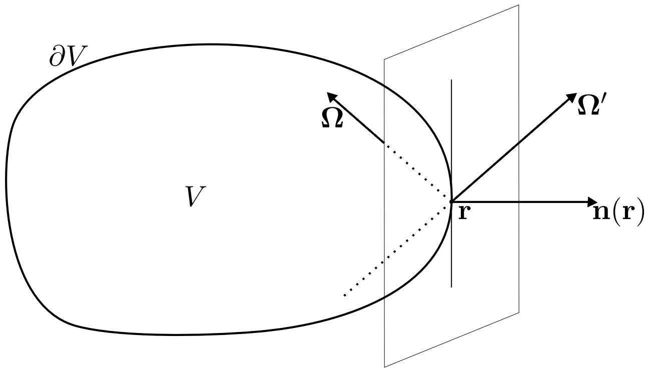


Figure 1.1: *Illustration of the directions $\mathbf{\Omega}$ and $\mathbf{\Omega}'$ used in the reflective boundary conditions (1.7) together with the outgoing normal $\mathbf{n}(\mathbf{r})$. The flux at $\mathbf{r} \in \partial V$ in the incoming direction $\mathbf{\Omega}$ has to equal the flux at $\mathbf{r}$ in the outgoing direction $\mathbf{\Omega}'$.*

In this thesis we will restrict ourselves to vacuum and reflective boundary conditions (1.6) and (1.7).

1.4 Adjoint problem

In [80, §1.6] the *adjoint problem* to the criticality problem (1.5), namely

$$\mathcal{T}^*\Psi^* \;=\; \mathcal{S}^*\Psi^* \;+\; \lambda^*\mathcal{F}^*\Psi^* \;, \tag{1.8}$$

is derived. The adjoint operators are given by

$$\begin{aligned}
\mathcal{T}^*\Psi^* &= -\mathbf{\Omega}\cdot\nabla\Psi^*(\mathbf{r},E,\mathbf{\Omega}) \;+\; \sigma(\mathbf{r},E)\Psi^*(\mathbf{r},E,\mathbf{\Omega}) \;, \\
\mathcal{S}^*\Psi^* &= \frac{1}{4\pi}\int_{\mathbb{R}+}\int_{\mathbb{S}^2}\sigma_s(\mathbf{r},E,E',\mathbf{\Omega},\mathbf{\Omega}')\Psi^*(\mathbf{r},E',\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}E' \;, \quad \text{and} \\
\mathcal{F}^*\Psi^* &= \frac{\nu(\mathbf{r},E)\sigma_f(\mathbf{r},E)}{4\pi}\int_{\mathbb{R}+}\chi(E')\int_{\mathbb{S}^2}\Psi^*(\mathbf{r},E',\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}E' \;.
\end{aligned}$$

The corresponding adjoint boundary conditions for (1.6) demand that the *outgoing* adjoint flux is zero, i.e.

$$\Psi^*(\mathbf{r},E,\mathbf{\Omega}) \;=\; 0 \quad \text{when} \quad \mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r}) \;>\; 0 \;, \quad \mathbf{r}\in\partial V \;. \tag{1.9}$$

The boundary conditions (1.6) and (1.9) ensure that $\mathcal{T}^*$, $\mathcal{S}^*$ and $\mathcal{F}^*$ are the adjoint operators to $\mathcal{T}$, $\mathcal{S}$ and $\mathcal{F}$, and satisfy

$$\begin{aligned}\langle\Psi^*,\mathcal{T}\Psi\rangle &= \langle\mathcal{T}^*\Psi^*,\Psi\rangle\ , \\ \langle\Psi^*,\mathcal{S}\Psi\rangle &= \langle\mathcal{S}^*\Psi^*,\Psi\rangle\ , \quad \text{and} && (1.10) \\ \langle\Psi^*,\mathcal{F}\Psi\rangle &= \langle\mathcal{F}^*\Psi^*,\Psi\rangle\ , && (1.11)\end{aligned}$$

where the inner product $\langle\,\cdot\,,\,\cdot\,\rangle$ on $L^2(V\times\mathbb{R}^+\times\mathbb{S}^2)$ is defined by

$$\langle f,g\rangle := \int_V\int_{\mathbb{R}^+}\int_{\mathbb{S}^2} f(\mathbf{r},E,\mathbf{\Omega})g(\mathbf{r},E,\mathbf{\Omega})\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\,\mathrm{d}\mathbf{r}\ .$$

Considering the transport operator and using integration by parts for the spatial derivative, we have (using shorthand notation)

$$\begin{aligned}\langle\Psi^*,\mathcal{T}\Psi\rangle &= \iiint_V \Psi^*\big(\mathbf{\Omega}\cdot\nabla\Psi+\sigma\Psi\big) \\ &= \iiint_V (\Psi^*\mathbf{\Omega})\cdot\nabla\Psi + \iiint_V \sigma\Psi^*\Psi \\ &= \iiint_{\partial V}\Psi^*\Psi\,(\mathbf{\Omega}\cdot\mathbf{n}) \;-\; \iiint_V \big(\nabla\cdot(\Psi^*\mathbf{\Omega})\big)\Psi \;+\; \iiint_V \sigma\Psi^*\Psi \\ &= \iiint_V \big(-\mathbf{\Omega}\cdot\nabla\Psi^*+\sigma\Psi^*\big)\Psi \;=\; \langle\mathcal{T}^*\Psi^*,\Psi\rangle\ ,\end{aligned}$$

where we made in the penultimate step use of the boundary conditions (1.6) and (1.9) to obtain that the boundary integral vanishes. The proofs that the scatter and fission operators satisfy (1.10) and (1.11) follow in a similar way by rearranging the integrals and are shown, for example, in [80, p. 49].

The physical interpretation of the adjoint flux is that of the *importance* of the forward flux at the same position, energy and direction. An explanation of this interpretation is given in [80, p. 50]. This concept of importance also provides a simple interpretation of the adjoint boundary conditions (1.9) in the case of vacuum boundary conditions for the forward problem: The neutrons that leave the system will not re-enter it and have therefore zero importance.

Adjoint equations have been applied in several areas of neutron transport problems. Their use to estimate detector responses is particularly important for Monte Carlo methods where it is easier to estimate intervals over a large region than over a small detector area as explained in [80, p. 50]. Recently, the use of coarse deterministic adjoint solutions to define input parameters of Monte Carlo codes has lead to *hybrid Monte Carlo-deterministic* methods (also called *automatic variance reduction* methods). For further details on this approach see the review [52] and references therein.

The solution to (1.8) can also be applied in perturbation theory to estimate the change in criticality for small variations of the problem. We will make use of the adjoint equation in Chapter 5, where we derive an iterative scheme to compute changes of arbitrary size (i.e. not only small changes as in many perturbation theories) from a known base solution by using the forward and adjoint solution of the base problem. Further details on the adjoint equation in neutron transport theory and additional application areas can be found in [12, §6].

1.5 Solution methods

In this section we discuss solution methods in neutron transport theory that aim to find a solution Ψ of the *source problem*

$$(\mathcal{T} - \mathcal{S} - \mathcal{F})\Psi \;=\; Q \tag{1.12}$$

subject to suitable boundary conditions. We shall see in Chapter 3 that iterative methods for solving the *eigenvalue* problem (1.5) lead to a sequence of source problems of the form (1.12). Hence, being able to solve these problems is also important for the solution of the criticality problem.

Many different numerical methods for solving neutron transport source problems have been developed over the years and describing them all in detail would go beyond the scope of this thesis. Essentially, the solution approaches can be separated into two classes: deterministic methods and stochastic techniques. Deterministic methods discretise the problem and then solve the subsequent system of algebraic equations, whereas the Monte Carlo method, which models the individual neutrons in a randomised way, is an example of a stochastic technique. A good discussion of the difference between the two approaches is given in [75].

In this section we give a brief overview about some of the more widely used solution methods (discrete ordinates, spherical harmonics, method of characteristics, diffusion theory, and the Monte Carlo method). We will provide further details of the discrete ordinates method and the Monte Carlo approach in Section 2.4 and Chapter 4 respectively.

Energy discretisation

For all deterministic methods the discretisation with respect to energy is performed by considering the so-called *multi-group equations* (see e.g. [80, §2.2]). This uses a partition of the energy range into G intervals (or groups) as shown in Figure 1.2. There E_0 and E_G denote the lower and upper limit for neutron energies such that the neutron flux outside this range can be neglected.

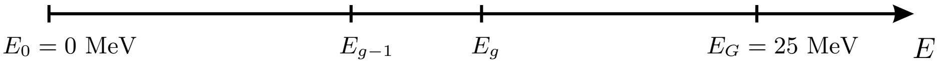


Figure 1.2: *Division of the energy range into G energy groups.*

The goal of this discretisation is to arrive at a system of equations in terms of the *group fluxes*

$$\Psi_g(\mathbf{r}, \mathbf{\Omega}) \;:=\; \int_{E_{g-1}}^{E_g} \Psi(\mathbf{r}, E, \mathbf{\Omega})\,\mathrm{d}E \ , \quad \text{for} \quad g = 1, \dots, G \ .$$

Furthermore, it is assumed that the angular flux Ψ can be expressed in each energy group as the product of a known globally defined function of energy f and the corresponding group flux Ψ_g, i.e. that

$$\Psi(\mathbf{r}, E, \mathbf{\Omega}) \;=\; f(E)\Psi_g(\mathbf{r}, \mathbf{\Omega}) \ , \quad \text{for} \quad E_{g-1} \le E \le E_g \ , \tag{1.13}$$

where

$$\int_{E_{g-1}}^{E_g} f(E)\,\mathrm{d}E \;=\; 1 \ , \quad \text{for} \quad g = 1, \dots, G \ .$$

Analogous to (1.13) we assume that the source Q can be expressed as

$$Q(\mathbf{r}, E, \mathbf{\Omega}) \;=\; f(E)Q_g(\mathbf{r}, \mathbf{\Omega}) \ , \quad \text{for} \quad E_{g-1} \le E \le E_g \ , \tag{1.14}$$

using the group sources

$$Q_g(\mathbf{r}, \mathbf{\Omega}) \;:=\; \int_{E_{g-1}}^{E_g} Q(\mathbf{r}, E, \mathbf{\Omega})\,\mathrm{d}E \ , \quad \text{for} \quad g = 1, \dots, G \ .$$

Dividing the energy intervals in (1.12) into sums of intervals over the different energy ranges (E_{g-1}, E_g), then using (1.13) and (1.14), and integrating over $E \in (E_{g-1}, E_g)$

gives

$$\begin{aligned} &\boldsymbol{\Omega}\cdot\nabla\Psi_g(\mathbf{r},\boldsymbol{\Omega}) \;+\; \sigma_g(\mathbf{r})\Psi_g(\mathbf{r},\boldsymbol{\Omega}) \\ =\;& \frac{1}{4\pi}\sum_{g'=1}^{G}\int_{\mathbb{S}^2}\sigma_{sgg'}(\mathbf{r},\boldsymbol{\Omega}',\boldsymbol{\Omega})\Psi_{g'}(\mathbf{r},\boldsymbol{\Omega}')\,\mathrm{d}\boldsymbol{\Omega}' \\ &+\;\frac{\chi_g}{4\pi}\sum_{g'=1}^{G}\nu_{g'}(\mathbf{r})\sigma_{fg'}(\mathbf{r})\int_{\mathbb{S}^2}\Psi_{g'}(\mathbf{r},\boldsymbol{\Omega}')\,\mathrm{d}\boldsymbol{\Omega}' \;+\; Q_g(\mathbf{r},\boldsymbol{\Omega}) \end{aligned} \tag{1.15}$$

for $g = 1,\dots,G$, where

$$\begin{aligned} \sigma_g(\mathbf{r}) &= \int_{E_{g-1}}^{E_g}\sigma(\mathbf{r},E)f(E)\,\mathrm{d}E\ , \\ \sigma_{sgg'}(\mathbf{r},\boldsymbol{\Omega}',\boldsymbol{\Omega}) &= \int_{E_{g-1}}^{E_g}\int_{E_{g'-1}}^{E'_g}\sigma_s(\mathbf{r},E',E,\boldsymbol{\Omega}',\boldsymbol{\Omega})f(E')\,\mathrm{d}E'\,\mathrm{d}E\ , \\ \nu_{g'}(\mathbf{r})\sigma_{fg'}(\mathbf{r}) &= \int_{E_{g'-1}}^{E'_g}\nu(\mathbf{r},E')\sigma_f(\mathbf{r},E')f(E')\,\mathrm{d}E'\ , \quad \text{and} \\ \chi_g &= \int_{E_{g-1}}^{E_g}\chi(E)\,\mathrm{d}E\ . \end{aligned}$$

The boundary conditions for (1.15) are obtained by replacing Ψ in (1.6) and (1.7) by the group fluxes Ψ_g.

Although we will give numerical solutions for some two energy group problems, for the theoretical analysis of this thesis we shall consider only monoenergetic problems with $G = 1$. In this case all fission neutrons have the same energy and hence $\chi_1 = 1$. As only one energy group is present, we can remove the group indices to reduce the notational burden for the monoenergetic case and write (1.15) as

$$\begin{aligned} &\boldsymbol{\Omega}\cdot\nabla\Psi(\mathbf{r},\boldsymbol{\Omega}) \;+\; \sigma(\mathbf{r})\Psi(\mathbf{r},\boldsymbol{\Omega}) \\ =\;& \frac{1}{4\pi}\int_{\mathbb{S}^2}\sigma_s(\mathbf{r},\boldsymbol{\Omega}',\boldsymbol{\Omega})\Psi(\mathbf{r},\boldsymbol{\Omega}')\,\mathrm{d}\boldsymbol{\Omega}' \\ &+\;\frac{1}{4\pi}\nu(\mathbf{r})\sigma_f(\mathbf{r})\int_{\mathbb{S}^2}\Psi(\mathbf{r},\boldsymbol{\Omega}')\,\mathrm{d}\boldsymbol{\Omega}' \;+\; Q(\mathbf{r},\boldsymbol{\Omega})\ , \end{aligned} \tag{1.16}$$

where we implicitly assume that a suitable function $f(E)$ exists and that the cross-section terms have been adjusted accordingly.

Discrete ordinates (S_N)

The first deterministic approach for solving a source problem of the form (1.12) that we discuss here is the so-called *discrete ordinates* or S_N-method. This method consists of approximating the angular integrals in the scatter and fission operators with a quadrature rule. The resulting equation is then evaluated at the angular quadrature points and a semidiscrete problem is obtained which is discrete in the angular variables but continuous in the spatial variable. For the numerical approximation of the spatial derivative there is a wide choice of finite difference and finite element schemes. After they are applied one obtains a fully-discrete problem that is equivalent to a generalised matrix eigenvalue problem.

We discuss the discrete ordinates approach in more detail in Section 2.4, where we also provide discretisation error estimates. Furthermore, we use this technique for our numerical results and give additional details about its application to a particular model problem in Section 3.3.3.

Spherical harmonics (P_N)

In the *spherical harmonics* method the angular component of the flux Ψ is expanded in spherical harmonics. A discretisation is obtained by truncating the expansion after a small number of terms. The spatial approximation again uses finite difference or finite element methods. This approach is usually denoted as the P_N-method and further details can be found, for example, in [12, §3].

Method of characteristics

The *method of characteristics* is typically used to reduce first-order partial differential equations to ordinary differential equations which are easier to solve. This is done by finding the characteristic curves, or *characteristics*, of the problem along which the partial differential equation can be expressed as an ordinary differential equation. Using the given initial data from the boundary conditions the solution can be found by integration of the ordinary differential equation.

Conveniently, the characteristics of the linear Boltzmann equation are straight lines in the flux directions $\mathbf{\Omega}$, along which the differential operator of (1.3) reduces to a total derivative. For a source problem of the form (1.12), an iterative process (the *source iteration*, which is discussed further in Section 4.3.2) is formed, where the scatter and fission contributions are fixed in each iteration, leading to problems of the form

$$\mathcal{T}\Psi^{(i+1)} \;=\; \widetilde{Q}^{(i)}\ , \quad \text{where} \quad \widetilde{Q}^{(i)} \;:=\; (\mathcal{S}+\mathcal{F})\Psi^{(i)} + Q\ . \tag{1.17}$$

The method of characteristics is now applied to find, for given $\widetilde{Q}^{(i)}$, the solution $\Psi^{(i+1)}$ of (1.17). This new flux estimate is then used to compute the next source $\widetilde{Q}^{(i+1)}$. The resulting iteration can be shown to converge under certain conditions (see, for example, Corollaries 4.6 and 4.9).

One of the first papers describing the method of characteristics for neutron transport problems in realistic geometries is [8]. Today, this method is frequently applied (see, for example, [62, 115] and references therein) and is subject to further theoretical investigations (see e.g. [10] for a new error estimate for two-dimensional problems).

Diffusion theory

Although not treated in this thesis, it is worth mentioning *diffusion theory* since it is widely used. Indeed, the following citation from 2008 [112, p. 43] claims that "Diffusion theory [...] is, in fact, the workhorse computational method of nuclear reactor physics."

As stated in [112], the diffusion approximation assumes that (i) the probability of scatter events is much higher than the probability of neutron absorption; (ii) the neutron distribution varies only linearly in space (only the first two terms of a spatial Taylor expansion of the flux are used to derive the diffusion equations); and (iii) the scatter is isotropic, i.e. the scatter cross-sections are independent of the incoming and outgoing direction. The first assumption is true for most moderating and structural materials in a nuclear reactor but does not hold for the fuel and control rod materials. The second assumption fails close to material boundaries and the final assumption only holds for heavy nuclei. Fortunately, transport theory can be used to adjust the boundary conditions and cross-sections used in diffusion theory, such that the results of numerical methods based on diffusion theory become more accurate ([112, p. 44]).

The multi-group diffusion equations for (1.15) are (see e.g. [80, (2-21)])

$$\begin{aligned} & -\nabla D_g(\mathbf{r})\cdot\nabla\phi_g(\mathbf{r}) \;+\; \sigma_g(\mathbf{r})\phi_g(\mathbf{r}) \\ =\; & \sum_{g'=1}^{G}\sigma_{sgg'}(\mathbf{r})\phi_{g'}(\mathbf{r}) \;+\; \chi_g\sum_{g'=1}^{G}\nu_{g'}(\mathbf{r})\sigma_{fg'}(\mathbf{r})\phi_{g'}(\mathbf{r}) \;+\; \widetilde{Q}_g(\mathbf{r}) \end{aligned}$$

for $g = 1,\dots,G$, where D_g are the so-called diffusion coefficients, the isotropic scattering gives $\sigma_{sgg'}(\mathbf{r}) = \sigma_{sgg'}(\mathbf{r}, \boldsymbol{\Omega}', \boldsymbol{\Omega})$ and

$$\begin{aligned} \phi_g(\mathbf{r}) &= \frac{1}{4\pi} \int_{\mathbb{S}^2} \Psi_g(\mathbf{r}, \boldsymbol{\Omega}) \, \mathrm{d}\boldsymbol{\Omega} \ , \\ \widetilde{Q}_g(\mathbf{r}) &= \frac{1}{4\pi} \int_{\mathbb{S}^2} Q_g(\mathbf{r}, \boldsymbol{\Omega}) \, \mathrm{d}\boldsymbol{\Omega} \ . \end{aligned}$$

For the time-dependent problem (1.1) diffusion theory leads to a system of parabolic partial differential equations which can be solved more efficiently than the general integro-differential form (1.1) by using numerical methods for diffusion problems (see e.g. [103, 127]).

However, it is important to note that diffusion theory is a simplification and does not incorporate all of the detail in the transport equation. Therefore, efficient methods to solve the linear Boltzmann equation and the criticality problem (1.5) are needed, the latter of which is the concern of this thesis.

Monte Carlo

The *Monte Carlo method* differs substantially from the deterministic methods described above. While the deterministic approaches aim to find a discretisation of the neutron transport equation (1.12) and subsequently solve the resulting system of algebraic equations, the Monte Carlo method appears to ignore the mathematical description (1.12) and solve the problem by simulating the behaviour of individual neutrons. The idea is that a sufficiently large number of simulations recovers the average behaviour of the system, which is described by (1.12).

We discuss this method in more detail in Chapter 4, where we also establish links between the apparently random processes in Monte Carlo and the underlying mathematics of the neutron transport equation.

1.6 Literature survey

In this thesis we deal with three different research areas (neutron transport theory, iterative eigenvalue solvers, and Monte Carlo methods) each of which has been studied extensively and has given rise to large numbers of publications. At this point we only provide a brief literature review about neutron transport theory and, in particular, criticality computations, but we give more detailed reviews of iterative methods and

Monte Carlo techniques in Section 3.1 and Chapter 4 respectively. Throughout the thesis additional references are cited where relevant.

There exists a large amount of background literature on neutron transport theory and nuclear engineering. Standard references for transport theory include the books [12, 20, 24, 29] and [80]. The recent text [112] focuses on the physical and engineering aspects in nuclear reactors, updating and extending results from [28] and [43].

The paper [2] by Adams describes progress in neutron transport theory between 1973 and 2004 from a nuclear engineer's perspective. The article reviews the work of Ed Larsen containing several important contributions, ranging from analytic solutions for simple model problems to the diffusion synthetic acceleration scheme (which plays a major role in real world computations today, see [3]) and the Monte Carlo method.

There has also been widespread interest in transport theory from numerical analysts (e.g. [6, 63, 82, 83, 93]), but this activity is mainly related to the solution of source problems of the form (1.12) and not the criticality problem (1.5). However, some discretisation error estimates for computed eigenvalues are presented in, for example, [6] and [93], and the well-known integral form of the neutron transport equation that is used in the latter paper was the basis of the analysis performed in Section 2.2.

Numerical methods for neutron transport problems are the focus of the book [83] which contains many numerical techniques and research results from the former USSR. The monograph mainly considers iterative methods for source problems but it also contains a short section (X.31) on iterative schemes for the computation of the criticality and corresponding flux shape, presenting five different methods including inverse iteration.

The study of the numerical solution of the *criticality* problem (1.5) has been, compared with the vast amount of literature for *source* problem solutions, very limited. Some standard references contain short sections on the solution of the criticality problem, such as [12, §4.4] where eigenvalue problems in multi-group theory have been considered, and the example of an iterative scheme (a variation of the power method) for the multi-group diffusion equation has been given.

In 1984 [80, p. 92] the following remark was made about iterative methods in the context of the solution of the criticality problem: "Multiplication eigenvalue problems are invariably solved by the method of power iteration." Section 2.5 in this book then discusses the power method and acceleration schemes based on extrapolation and the use of a diffusion solution. While in other areas alternative eigenvalue techniques have become the preferred method of choice, in the thesis [115, p. 140] from 2008 it is stated that the power method is (still) the most commonly used approach to solve neutron transport eigenvalue problems.

However, recently other methods such as implicitly restarted Arnoldi for the search of time-eigenvalues (see [78]) and the computation of the criticality value $1/k_{\text{eff}}$ (see [120]) have gained interest. The latter paper tests the implicitly restarted Arnoldi method from the ARPACK software package [79] and compares the numerical results to the power method showing that for several test problems, the Krylov subspace based Arnoldi method requires substantially fewer iterations. This is particularly the case for problems with a dominance ratio close to one, i.e. where the difference between the principal eigenvalue and the next eigenvalue is small.

The recent PhD thesis [23] considers the explicitly restarted Arnoldi method for Monte Carlo criticality computations. Several numerical results are provided that compare the explicitly restarted Arnoldi approach with the power method. Tests show that using similar computational effort both methods produce estimates for the fundamental eigenvalue within statistical uncertainties while the former method additionally provides estimates for the second and third largest eigenvalue. Furthermore, the superiority of the Arnoldi method over the power method is again shown numerically for problems with a small distance of the fundamental eigenvalue to the next.

A possible drawback of an Arnoldi method in the context of Monte Carlo is the need for a discrete mesh on which to orthogonalise the iterates. This introduces a discretisation error which is not present when the power method is used to compute only the criticality as we will see in Chapter 4. Overall, the thesis [23] presents the first application of an Arnoldi method to the criticality problem using a Monte Carlo implementation showing that a combination of the two methods is possible. Several further extensions and challenges are mentioned and need to be answered to determine if this technique is feasible for real world problems.

Another approach based on using Krylov subspaces is discussed in [51] and [87] and uses the Orthomin method (see [48, §2.2] for details about Orthomin for the solution of linear systems). Even the simple bisection method has been used in [89] to find the criticality estimate by altering $1/k_{\text{eff}}$ until the determinant of a matrix representation of $(\mathcal{T} - \mathcal{S} - 1/k_{\text{eff}}\mathcal{F})$ becomes zero. Finally, a brief discussion of the inverse power method with a fixed shift for solving the criticality problem is given in [4], where a discretisation of the problem is assumed.

Despite the development and application of new algorithms, we note that few of the recent advances in the *theory* of iterative methods for eigenvalue problems have been applied to the solution of the reactor criticality problem. In particular, it appears that so far there has been no analysis of *inexact* methods.

To the author's knowledge the analysis of inexact inverse iteration for the criticality problem that is presented in Chapter 3 of this thesis has not been done before. It is founded on a rigorous mathematical basis which we establish in Chapter 2 and considers the criticality problem in operator form. Therefore, our results are independent of the discretisation and (source problem) solution method employed. We consider the general situation of shifted inverse iteration that includes the power method as a special case, but also extends to faster converging shift-invert methods such as Rayleigh Quotient iteration.

For details of numerical methods for general matrix eigenvalue problems (i.e. not necessarily discretisations of the criticality problem), we refer to standard textbooks such as [45, 91, 99, 122]. A concise review of different eigenvalue computation techniques (including direct and iterative solvers) and the research developments in the last century is given in [44]. We focus in this thesis on *inexact iterative methods* for the eigenvalue problem, which we discuss in Chapter 3 where we also look in more detail at the literature for these methods.

1.7 Contributions of this thesis and outline

This thesis provides the following original contributions to the knowledge of the research community. It

- explores an underlying symmetry of the neutron transport equation (known to nuclear engineers but not exploited by numerical analysts) and provides a rigorous mathematical basis for a class of model problems (Sections 2.1 and 2.2);
- provides some fundamental theory for integral operators occurring in neutron transport theory (for example Theorem 2.9);
- shows the existence of a simple smallest positive real eigenvalue with corresponding eigenfunction that is strictly positive in the interior of the reactor using only simple analytical tools and avoiding semigroup theory (Section 2.3);
- establishes a systematic convergence analysis for inexact shifted inverse iteration as an eigenvalue solver applied to the criticality problem on the space of linear operators (Section 3.2);
- provides simple links between the mathematics of neutron transport theory and the use of Monte Carlo methods as a solution technique for source and eigenvalue problems (Chapter 4);

- derives an iterative method for criticality computations which computes the change in the eigenvalue when the problem is perturbed from one with a known solution (this situation often occurs when designing new nuclear reactors) (Section 5.1), and it
- gives a convergence analysis of the perturbation scheme that shows that for small perturbations and inexact inner solves the method can compete with standard approaches, but that also highlights when the convergence is no longer guaranteed (Sections 5.2 to 5.4).

Throughout the work numerical results to support the theory are given.

The thesis is organised as follows. In the second chapter we introduce a class of model problems for which we establish a relationship between the non-symmetric criticality problem (1.5) and a symmetric problem in a space of reduced dimension. Several properties of the operators involved are shown and a rigorous mathematical basis for the following parts of this thesis is provided. Furthermore, we prove the existence of a simple smallest positive real eigenvalue. We finish Chapter 2 by considering a popular discretisation approach and giving examples of how to preserve the underlying symmetry of the problem in the finite dimensional form.

The following three chapters focus on numerical methods for solving the criticality problem. Chapter 3 considers iterative methods for the criticality problem and provides a convergence analysis for inexact inverse iteration which is supported by numerical results. In Chapter 4 the application of the Monte Carlo method to neutron transport problems is described, and Chapter 5 discusses a new iterative scheme that is based on computing eigenvalue and eigenvector corrections to a known solution. Again a convergence analysis and numerical examples are given. Chapter 6 contains the conclusions and suggests possible directions for further research.

Chapter 2

Integral equation analysis and discretisation

This chapter provides the mathematical foundations for the remainder of the thesis. We discussed in Section 1.2 how the criticality of a nuclear reactor is determined by the solution of an eigenvalue problem. In particular, we asked to find the smallest real eigenvalue $\lambda > 0$ of (1.5) without proving that such an eigenvalue exists. In Section 2.3 we answer this question for a class of model problems which are described in the following Section 2.1.

Section 2.2 shows that for these model problems the search for the smallest positive real eigenvalue of the non-symmetric eigenvalue problem (1.5) is equivalent to finding the smallest positive real eigenvalue of a *symmetric* eigenvalue problem in a lower dimensional space. Several mathematical properties of this symmetric problem and the operators involved therein are established, including a new proof for a not well-known norm estimate that is crucial for the following analysis.

In Section 2.3 we then show for the model problems from Section 2.1 that a smallest positive real eigenvalue exists. Furthermore, we show that it is simple and has a corresponding eigenfunction that is strictly positive within the reactor.

The final section of this chapter discusses a frequently used discretisation technique and the challenge to preserve the underlying symmetry of the problem in the finite dimensional form. We also give discretisation error estimates for a scheme that we employ in our numerical tests in the following chapters.

2.1 Model problems

The majority of this thesis focuses on homogeneous model problems with isotropic scattering in the monoenergetic case. Considering only one energy group results in all energy dependencies being removed (see equation (1.16) on page 12). Furthermore, due to the isotropic scatter and homogeneous material, all cross-sections become constants. The relation (1.2) then reduces to

$$\sigma \;=\; \sigma_c + \sigma_s + \sigma_f \ . \tag{2.1}$$

We apply vacuum boundary conditions for these problems. In such a reactor all neutrons travel with the same constant speed and no neutrons enter the reactor from outside. Despite imposing substantial simplifications these model problems are a good starting point for the analysis of neutron transport and several versions of them in different spatial dimensions have been studied in the literature (e.g. [6, 63, 82, 93]). All four of these papers consider the discretisation error of different discrete ordinates schemes in various geometries. We will discuss [93] in more detail in Section 2.4 and refer to the other papers when considering the respective geometries.

We perform the majority of the following analysis in the three dimensional case where $\mathbf{r} \in V \subset \mathbb{R}^3$ and $\boldsymbol{\Omega} \in \mathbb{S}^2$, and state the results for the 2D and 1D cases where similar arguments apply.

3D model

In the 3D case the above assumptions lead to the eigenvalue problem (1.5) taking the form: Find (λ, Ψ) such that

$$\boldsymbol{\Omega} \cdot \nabla \Psi(\mathbf{r}, \boldsymbol{\Omega}) \,+\, \sigma \Psi(\mathbf{r}, \boldsymbol{\Omega}) \,-\, \frac{\sigma_s}{4\pi} \int_{\mathbb{S}^2} \Psi(\mathbf{r}, \boldsymbol{\Omega}')\, \mathrm{d}\boldsymbol{\Omega}' \;=\; \lambda\, \frac{\nu \sigma_f}{4\pi} \int_{\mathbb{S}^2} \Psi(\mathbf{r}, \boldsymbol{\Omega}')\, \mathrm{d}\boldsymbol{\Omega}' \tag{2.2}$$

for all $(\mathbf{r}, \boldsymbol{\Omega}) \in V \times \mathbb{S}^2$, subject to

$$\Psi(\mathbf{r}, \boldsymbol{\Omega}) \;=\; 0 \quad \text{when} \quad \mathbf{n}(\mathbf{r}) \cdot \boldsymbol{\Omega} \;<\; 0\ , \quad \mathbf{r} \in \partial V\ . \tag{2.3}$$

Two subcases of this have been studied in the literature as discussed below. To describe them, let $(\theta, \varphi) \in [0, \pi] \times [0, 2\pi]$ denote the usual spherical polar coordinates on $\mathbb{S}^2$ (see Figure 2.1).

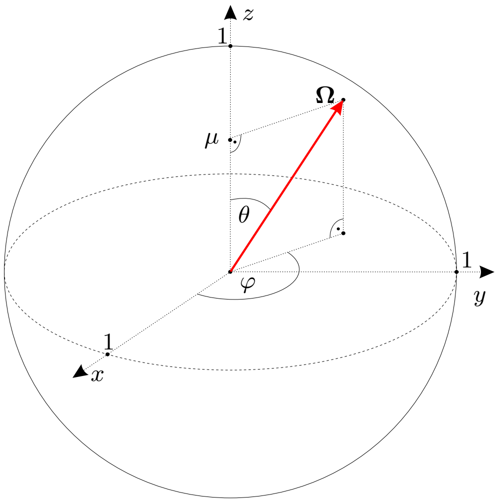


Figure 2.1: *The contribution of $\mathbf{\Omega}$ in the z-direction is $\mu = \cos\theta$.*

2D model

Here it is assumed that $\Psi(\mathbf{r},\mathbf{\Omega}) = \Psi(\widetilde{\mathbf{r}},\widetilde{\mathbf{\Omega}})$, where $\widetilde{\mathbf{r}} \in \widetilde{V} \subset \mathbb{R}^2$ and $\widetilde{\mathbf{\Omega}} = (\cos\varphi, \sin\varphi)$ lies on the unit circle $\mathbb{S}^1$. The resulting model problem becomes

$$\widetilde{\mathbf{\Omega}} \cdot \widetilde{\nabla}\Psi(\widetilde{\mathbf{r}},\widetilde{\mathbf{\Omega}}) + \sigma\Psi(\widetilde{\mathbf{r}},\widetilde{\mathbf{\Omega}}) - \frac{\sigma_s}{2\pi}\int_{\mathbb{S}^1}\Psi(\widetilde{\mathbf{r}},\widetilde{\mathbf{\Omega}}')\,\mathrm{d}\widetilde{\mathbf{\Omega}}' = \lambda\,\frac{\nu\sigma_f}{2\pi}\int_{\mathbb{S}^1}\Psi(\widetilde{\mathbf{r}},\widetilde{\mathbf{\Omega}}')\,\mathrm{d}\widetilde{\mathbf{\Omega}}'$$

for $(\widetilde{\mathbf{r}},\widetilde{\mathbf{\Omega}})$ on $\widetilde{V} \times \mathbb{S}^1$ (where $\widetilde{\nabla}$ denotes the 2D gradient). The vacuum boundary conditions then read

$$\Psi(\widetilde{\mathbf{r}},\widetilde{\mathbf{\Omega}}) = 0 \quad \text{when} \quad \widetilde{\mathbf{n}}(\widetilde{\mathbf{r}})\cdot\widetilde{\mathbf{\Omega}} < 0\,, \quad \widetilde{\mathbf{r}} \in \partial\widetilde{V}\,,$$

where $\widetilde{\mathbf{n}}(\widetilde{\mathbf{r}})$ again denotes the outward unit normal at $\widetilde{\mathbf{r}} \in \partial\widetilde{V}$ (see e.g. [6] and the references therein).

1D model

In the 1D case it is assumed that $\Psi(\mathbf{r},\mathbf{\Omega}) = \Psi(z,\mu)$, where $z \in [0,1]$ and $\mu = \cos\theta \in [-1,1]$. Then the problem (2.2), (2.3) reduces to

$$\mu\frac{\partial}{\partial z}\Psi(z,\mu) + \sigma\Psi(z,\mu) - \frac{\sigma_s}{2}\int_{-1}^{1}\Psi(z,\mu')\,\mathrm{d}\mu' = \lambda\,\frac{\nu\sigma_f}{2}\int_{-1}^{1}\Psi(z,\mu')\,\mathrm{d}\mu'\,, \tag{2.4}$$

which has to be solved on $[0,1] \times [-1,1]$, subject to

$$\Psi(0,\mu) = 0 \quad \text{when} \quad \mu > 0 \quad \text{and} \quad \Psi(1,\mu) = 0 \quad \text{when} \quad \mu < 0 \ . \tag{2.5}$$

This "1D slab geometry" model has received a lot of attention in the literature (e.g. in [82] and [93]).

2.2 Integral equation methods

In this section we establish the relation of the model problems introduced above to corresponding symmetric positive definite integral operator eigenvalue problems. Properties of the arising integral operator, including a vital norm estimate, are proved. The symmetry and compactness of the operator can be exploited by applying the Hilbert-Schmidt theorem to obtain an orthonormal basis for the corresponding function space, which we will use in later chapters to analyse the convergence of iterative methods.

2.2.1 Equivalence to an integral equation

The key to the reduction to a symmetric problem is to introduce another physical quantity, the *scalar flux* ϕ, which is obtained by integrating the angular flux Ψ over all directions, i.e.

$$\phi(\mathbf{r}) \ = \ (\mathcal{P}\Psi)(\mathbf{r}) \ := \ \frac{1}{4\pi} \int_{\mathbb{S}^2} \Psi(\mathbf{r}, \boldsymbol{\Omega}')\, \mathrm{d}\boldsymbol{\Omega}' \ . \tag{2.6}$$

To make the reduction mathematically precise, we introduce the usual Lebesgue space $L^2(V)$ with norm $\| \cdot \|_{L^2(V)}$. Also for any $1 \leq p \leq \infty$ we introduce the space

$$L^2(V, L^p(\mathbb{S}^2)) \ := \ \left\{ \Psi : V \times \mathbb{S}^2 \to \mathbb{R} \ : \ \int_V \|\Psi(\mathbf{r}, \cdot)\|^2_{L^p(\mathbb{S}^2)} \, \mathrm{d}\mathbf{r} \ < \ \infty \right\}$$

with norm $\|\Psi\|^2_{L^2(V, L^p(\mathbb{S}^2))} := \int_V \|\Psi(\mathbf{r}, \cdot)\|^2_{L^p(\mathbb{S}^2)} \, \mathrm{d}\mathbf{r}$, where $L^p(\mathbb{S}^2)$ is the usual L^p space of functions defined on $\mathbb{S}^2$. The operator $\mathcal{P}$ defined in (2.6) is then a bounded linear operator from $L^2(V, L^1(\mathbb{S}^2))$ to $L^2(V)$.

In the 2D and 1D settings $\mathcal{P}$ becomes, respectively,

$$\phi(\widetilde{\mathbf{r}}) \ := \ \frac{1}{2\pi} \int_{\mathbb{S}^1} \Psi(\widetilde{\mathbf{r}}, \widetilde{\boldsymbol{\Omega}}')\, \mathrm{d}\widetilde{\boldsymbol{\Omega}}' \quad \text{and} \quad \phi(z) \ := \ \frac{1}{2} \int_{-1}^{1} \Psi(z, \mu')\, \mathrm{d}\mu' \ .$$

In the following lemma, we make use of the notation

$$d(\mathbf{r},\mathbf{\Omega}) \;:=\; \inf\{s>0 \,:\, \mathbf{r}-s\mathbf{\Omega} \notin V\}\ ,$$

which describes the distance from $\mathbf{r}$ back to the boundary along the direction $-\mathbf{\Omega}$ (see Figure 2.2).

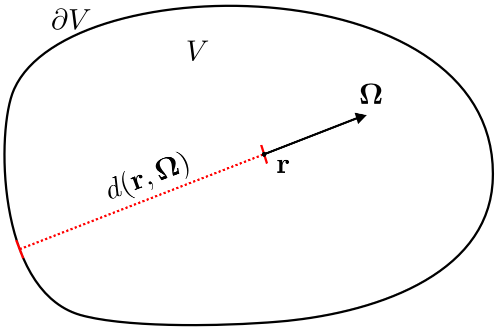


Figure 2.2: *Distance $d(\mathbf{r},\mathbf{\Omega})$ from $\mathbf{r}$ to the boundary ∂V along $-\mathbf{\Omega}$.*

Throughout we assume that V is a convex domain in $\mathbb{R}^3$ and for convenience we assume that its boundary ∂V is C^1, so that the normal direction $\mathbf{n}$ is a continuous function on ∂V. If $\mathbf{r}\in V$, it then follows that $\mathbf{\Omega}$ is an inward pointing direction at the boundary point $\mathbf{r}-d(\mathbf{r},\mathbf{\Omega})\mathbf{\Omega}\in\partial V$, and so by (2.3),

$$\Psi(\mathbf{r}-d(\mathbf{r},\mathbf{\Omega})\mathbf{\Omega},\mathbf{\Omega}) \;=\; 0\ . \tag{2.7}$$

Lemma 2.1. *Suppose $g\in L^2(V,L^\infty(\mathbb{S}^2))$ and consider the problem of solving*

$$\mathcal{T}\Psi(\mathbf{r},\mathbf{\Omega}) \;:=\; \mathbf{\Omega}\cdot\nabla\Psi(\mathbf{r},\mathbf{\Omega}) \,+\, \sigma\Psi(\mathbf{r},\mathbf{\Omega}) \;=\; g(\mathbf{r},\mathbf{\Omega}) \tag{2.8}$$

on $V\times\mathbb{S}^2$, together with the boundary conditions (2.3). *This problem has a unique solution $\Psi\in L^2(V,L^1(\mathbb{S}^2))$ given by*

$$\Psi(\mathbf{r},\mathbf{\Omega}) \;=\; \int_0^{d(\mathbf{r},\mathbf{\Omega})} \exp(-\sigma s)g(\mathbf{r}-s\mathbf{\Omega},\mathbf{\Omega})\,\mathrm{d}s \quad \textit{for} \quad (\mathbf{r},\mathbf{\Omega})\in V\times\mathbb{S}^2\ . \tag{2.9}$$

Proof. First observe that (2.8) is equivalent to the statement

$$-\frac{\mathrm{d}}{\mathrm{d}s}\big[\Psi(\mathbf{r}-s\mathbf{\Omega},\mathbf{\Omega})\exp(-\sigma s)\big] \;=\; g(\mathbf{r}-s\mathbf{\Omega},\mathbf{\Omega})\exp(-\sigma s) \tag{2.10}$$

for $(\mathbf{r},\mathbf{\Omega})\in V\times\mathbb{S}^2$ and $s>0$, provided $\mathbf{r}-s\mathbf{\Omega}\in V$.

To show that (2.9) is a solution of (2.8), observe that if $(\mathbf{r},\mathbf{\Omega})\in V\times\mathbb{S}^2$ and $s>0$

satisfy $\mathbf{r} - s\mathbf{\Omega} \in V$, then (2.9) implies

$$\Psi(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega}) \;=\; \int_0^{d(\mathbf{r}-s\mathbf{\Omega},\mathbf{\Omega})} \exp(-\sigma s') g(\mathbf{r} - (s+s')\mathbf{\Omega}, \mathbf{\Omega})\, \mathrm{d}s' \ .$$

Now making the change of variable $s'' = s' + s$ and observing that

$$d(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega}) + s \;=\; d(\mathbf{r}, \mathbf{\Omega}) \ ,$$

we obtain

$$\Psi(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega}) \exp(-\sigma s) \;=\; \int_s^{d(\mathbf{r},\mathbf{\Omega})} \exp(-\sigma s'') g(\mathbf{r} - s''\mathbf{\Omega}, \mathbf{\Omega})\, \mathrm{d}s'' \ ,$$

which implies (2.10).

Uniqueness of the solution to (2.8) follows since with $g = 0$, integrating (2.10) from $s = 0$ to $s = d(\mathbf{r}, \mathbf{\Omega})$ and using (2.7) shows that Ψ vanishes. The proof that $\Psi \in L^2(V, L^1(\mathbb{S}^2))$ is deferred to Remark 2.4. □

We now state a result from [65, Theorem 1, p. 324] which we will need to prove Lemma 2.3.

Lemma 2.2. *Consider* $L^p(D_1)$, $L^q(D_2)$ *with* $1 \le p$, $q \le \infty$, *where* D_1 *and* D_2 *are, respectively,* d_1*- and* d_2*-dimensional bounded regions in Euclidean space. Consider the integral operator*

$$(\mathcal{K}g)(\mathbf{r}) \;=\; \int_{D_1} k(\mathbf{r}, \mathbf{r}') g(\mathbf{r}')\, \mathrm{d}\mathbf{r}' \ , \quad \mathbf{r} \in D_2 \ ,$$

and assume the following conditions are satisfied:

$$\left[\int_{D_1} |k(\mathbf{r}, \mathbf{r}')|^u \, \mathrm{d}\mathbf{r}'\right]^{1/u} \;\le\; C_1 \ , \quad u > 0$$

for almost all $\mathbf{r} \in D_2$,

$$\left[\int_{D_2} |k(\mathbf{r}, \mathbf{r}')|^v \, \mathrm{d}\mathbf{r}\right]^{1/v} \;\le\; C_2 \ , \quad v > 0$$

for almost all $\mathbf{r}' \in D_1$,

$$q \;\ge\; p \ , \quad q \;\ge\; v \quad \textit{and} \quad \left(1 - \frac{v}{q}\right) p' \;\le\; u \ ,$$

where p' *is the conjugate exponent of* p *defined by* $1/p + 1/p' = 1$.

Then the integral operator $\mathcal{K}$ is a continuous linear operator mapping $L^p(D_1)$ into $L^q(D_2)$ and

$$\|\mathcal{K}\| \;\leq\; C_1^{1-v/q} C_2^{v/q} \;.$$

Lemma 2.3. *Consider* (2.8) *in the special case $g(\mathbf{r},\boldsymbol{\Omega}) = g(\mathbf{r})$ with $g \in L^2(V)$, and define $\phi := \mathcal{P}\Psi$, where Ψ is the solution from* (2.9)*. Then $\phi \in L^2(V)$ and*

$$\left.\begin{aligned} \phi(\mathbf{r}) \;=\; (\mathcal{K}_\sigma g)(\mathbf{r}) \;:=\; (\mathcal{P}\mathcal{T}^{-1}g)(\mathbf{r}) \;&=\; \int_V k_\sigma(\mathbf{r}-\mathbf{r}')g(\mathbf{r}')\,\mathrm{d}\mathbf{r}' \;, \\ \textit{where} \quad k_\sigma(\mathbf{x}) \;&:=\; \frac{1}{4\pi}\,\frac{\exp(-\sigma\|\mathbf{x}\|_2)}{\|\mathbf{x}\|_2^2} \;, \quad \mathbf{x}\in\mathbb{R}^3 \;. \end{aligned}\right\} \tag{2.11}$$

Proof. Using (2.9) and applying $\mathcal{P}$ yields

$$\phi(\mathbf{r}) \;=\; \int_{\mathbb{S}^2}\int_0^{d(\mathbf{r},\boldsymbol{\Omega})} \frac{\exp(-\sigma s)}{4\pi s^2}\, g(\mathbf{r}-s\boldsymbol{\Omega})\, s^2 \,\mathrm{d}s\,\mathrm{d}\boldsymbol{\Omega} \;.$$

Now, using spherical coordinates centred at $\mathbf{r}$ with $\mathbf{r}' = \mathbf{r} - s\boldsymbol{\Omega}$, we obtain (2.11). Finally, the assumptions of Lemma 2.2 are satisfied with $p = q = 2$, $D_1 = D_2 = V$ and $u = v = 1$, and hence $\mathcal{K}_\sigma$ is a continuous (bounded) linear operator from $L^2(V)$ to $L^2(V)$. Therefore it follows that $\phi \in L^2(V)$ which finishes the proof. □

Remark 2.4. *The fact that Ψ given by* (2.9) *lies in $L^2(V, L^1(\mathbb{S}^2))$ when the right-hand side $g \in L^2(V, L^\infty(\mathbb{S}^2))$ can now be proved by using* (2.9) *to obtain*

$$\begin{aligned} \|\Psi(\mathbf{r},\cdot)\|_{L^1(\mathbb{S}^2)} \;&\leq\; 4\pi \int_{\mathbb{S}^2}\int_0^{d(\mathbf{r},\boldsymbol{\Omega})} \frac{\exp(-\sigma s)}{4\pi s^2}\,\|g(\mathbf{r}-s\boldsymbol{\Omega},\cdot)\|_{L^\infty(\mathbb{S}^2)}\, s^2\,\mathrm{d}s\,\mathrm{d}\boldsymbol{\Omega} \\ &=\; 4\pi\,(\mathcal{K}_\sigma f)(\mathbf{r}) \;, \end{aligned}$$

where $f(\mathbf{r}) = \|g(\mathbf{r},\cdot)\|_{L^\infty(\mathbb{S}^2)} \in L^2(V)$. The result follows since $\mathcal{K}_\sigma$ is a bounded linear operator on $L^2(V)$.

This type of integral equation reformulation is well-known for neutron transport *source problems* (see e.g. [80, §5]). The integral form of the neutron transport equation is often used as a tool in the design of iterative schemes to solve source problems as, for example, in [48, 53], where the source iteration method is discussed. The paper [53] additionally presents a remedy to avoid negative fluxes in the computation.

However, the use of the reduction to analyse *eigenvalue* problems is, apart from obtaining discretisation error estimates for the critical eigenvalue (see [6, 93] for 2D and 1D examples), less well-known. In particular, the author does not know of any literature which exploits the structure derived below to provide a convergence analysis of eigenvalue iteration methods to compute the criticality as we do in Chapter 3.

Corollary 2.5. *If* (λ, Ψ) *with* $\Psi \in L^2(V, L^1(\mathbb{S}^2))$ *is an eigenpair for* (2.2), (2.3)*, then* (λ, ϕ)*, with* $\phi = \mathcal{P}\Psi \in L^2(V)$*, is an eigenpair of the reduced generalised eigenvalue problem*

$$\phi(\mathbf{r}) - \sigma_s \mathcal{K}_\sigma \phi(\mathbf{r}) \; = \; \lambda\, \nu\sigma_f \mathcal{K}_\sigma \phi(\mathbf{r}) \ , \quad \mathbf{r} \in V \ . \tag{2.12}$$

Conversely, if (λ, ϕ) *with* $\phi \in L^2(V)$ *is an eigenpair of the problem* (2.12)*, and if we define* $\Psi \in L^2(V, L^1(\mathbb{S}^2))$ *by solving*

$$\mathcal{T}\Psi(\mathbf{r}, \mathbf{\Omega}) \; = \; \sigma_s \phi(\mathbf{r}) + \lambda\, \nu\sigma_f \phi(\mathbf{r}) \ , \tag{2.13}$$

subject to the boundary conditions (2.3)*, then* (λ, Ψ) *is an eigenpair of the problem* (2.2)*.*

Proof. Suppose (λ, Ψ) is an eigenpair of (2.2) with $\Psi \in L^2(V, L^1(\mathbb{S}^2))$, then

$$\mathcal{T}\Psi(\mathbf{r}, \mathbf{\Omega}) - \sigma_s \phi(\mathbf{r}) \; = \; \lambda\, \nu\sigma_f \phi(\mathbf{r}) \ , \quad \text{where} \quad \phi(\mathbf{r}) \; = \; \mathcal{P}\Psi(\mathbf{r}, \mathbf{\Omega}) \ .$$

Now it follows from Lemma 2.3 and the linearity of $\mathcal{K}_\sigma$, that (2.12) holds in $L^2(V)$.

To prove the converse statement, let $\Psi \in L^2(V, L^1(\mathbb{S}^2))$ be the unique solution of (2.13) and set $\widetilde{\phi} := \mathcal{P}\Psi$. Lemma 2.3 and (2.12) imply

$$\widetilde{\phi}(\mathbf{r}) \; = \; \sigma_s \mathcal{K}_\sigma \phi(\mathbf{r}) + \lambda\, \nu\sigma_f \mathcal{K}_\sigma \phi(\mathbf{r}) \; = \; \phi(\mathbf{r}) \ .$$

Hence $\mathcal{T}\Psi \; = \; \sigma_s \widetilde{\phi} + \lambda\, \nu\sigma_f \widetilde{\phi} \; = \; \mathcal{S}\Psi + \lambda\, \mathcal{F}\Psi$, as required. □

Analogous arguments can be applied to the 2D and 1D model problems. For the 2D problem with $\widetilde{\mathbf{r}} \in \widetilde{V}$ and $\widetilde{\mathbf{\Omega}} \in \mathbb{S}^1$ the equivalent of (2.8) is

$$\left(\mathcal{T}^{-1} g\right)(\widetilde{\mathbf{r}}, \widetilde{\mathbf{\Omega}}) \; = \; \Psi(\widetilde{\mathbf{r}}, \widetilde{\mathbf{\Omega}}) \; = \; \int_0^{d(\widetilde{\mathbf{r}}, \widetilde{\mathbf{\Omega}})} \exp(-\sigma s) g(\widetilde{\mathbf{r}} - s\widetilde{\mathbf{\Omega}}, \widetilde{\mathbf{\Omega}})\, \mathrm{d}s \ .$$

The scalar flux eigenvalue problem is then

$$\phi(\widetilde{\mathbf{r}}) - \sigma_s \mathcal{K}_\sigma \phi(\widetilde{\mathbf{r}}) \; = \; \lambda\, \nu\sigma_f \mathcal{K}_\sigma \phi(\widetilde{\mathbf{r}}) \ ,$$

where

$$(\mathcal{K}_\sigma g)(\widetilde{\mathbf{r}}) \; := \; \int_{\widetilde{V}} k_\sigma(\widetilde{\mathbf{r}} - \widetilde{\mathbf{r}}') g(\widetilde{\mathbf{r}}')\, \mathrm{d}\widetilde{\mathbf{r}}' \quad \text{and} \quad k_\sigma(\mathbf{x}) \; := \; \frac{1}{2\pi} \frac{\exp(-\sigma\|\mathbf{x}\|_2)}{\|\mathbf{x}\|_2} \ , \quad \mathbf{x} \in \mathbb{R}^2 \ .$$

For the 1D problem the inverse of the transport operator is given by

$$\left(\mathcal{T}^{-1}g\right)(z,\mu) \;=\; \Psi(z,\mu) \;=\; \begin{cases} \frac{1}{\mu}\int\limits_0^z \exp\left(\frac{\sigma}{\mu}(t-z)\right) g(t,\mu)\,\mathrm{d}t \;, & \text{if } \mu > 0 \\ -\frac{1}{\mu}\int\limits_z^1 \exp\left(\frac{\sigma}{\mu}(t-z)\right) g(t,\mu)\,\mathrm{d}t \;, & \text{if } \mu < 0 \;, \end{cases} \tag{2.14}$$

and the equivalent eigenvalue problem of reduced dimension is

$$\phi(z) - \sigma_s \mathcal{K}_\sigma \phi(z) \;=\; \lambda\, \nu \sigma_f \mathcal{K}_\sigma \phi(z) \;,$$

with

$$(\mathcal{K}_\sigma g)(z) := \int_0^1 k_\sigma(z-z')g(z')\,\mathrm{d}z' \quad \text{and} \quad k_\sigma(x) := \frac{1}{2}\int_0^1 \exp\left(-\frac{\sigma|x|}{\mu}\right)\frac{\mathrm{d}\mu}{\mu} \;. \tag{2.15}$$

Remark 2.6. *The singularity in the kernel k_σ in* (2.15) *is well-understood and $\mathcal{K}_\sigma$ maps $L^2([0,1])$ to the space of continuous functions $C([0,1])$ (see, for example, [47]). Further properties of the solution* (2.14) *to the 1D problem, such as the differentiability of Ψ and the extension of Ψ to a Hölder continuous function, are discussed in [66].*

Remark 2.7. *In the 1D problem the kernel $k_\sigma(x)$ is related to the exponential integral*

$$\mathrm{E}_1(x) \;=\; \int_1^\infty \exp(-xt)\,\frac{\mathrm{d}t}{t} \;=\; -\frac{1}{2}\left(\ln(x) + \gamma + \sum_{k=1}^\infty \frac{x^k}{k\,k!}\right) \;, \quad x > 0 \;,$$

where $\gamma \approx 0.5772$ is the Euler-Mascheroni constant (see [1, equations 5.1.4 and 5.1.11, pages 228-229]). For real $x \neq 0$ we have

$$k_\sigma(x) \;=\; \frac{1}{2}\int_0^1 \exp\left(-\frac{|\sigma x|}{\mu}\right)\frac{\mathrm{d}\mu}{\mu} \;=\; \frac{1}{2}\int_1^\infty \exp\left(-|\sigma x|\eta\right)\frac{\mathrm{d}\eta}{\eta} \;=\; \frac{1}{2}\,\mathrm{E}_1(|\sigma x|) \;. \tag{2.16}$$

2.2.2 Properties of the integral operator

We now establish various mathematical properties of the operator $\mathcal{K}_\sigma$. In 1D, 2D and in 3D the kernel k_σ is positive and symmetric and we use this to prove that the operator $\mathcal{K}_\sigma$ is a positive definite self-adjoint operator.

Lemma 2.8. *The integral operator $\mathcal{K}_\sigma$ is* self-adjoint *on the Hilbert space $L^2(\bar{V})$, i.e. $(\mathcal{K}_\sigma f, g)_{L^2(\bar{V})} = (f, \mathcal{K}_\sigma g)_{L^2(\bar{V})}$ for all $f, g \in L^2(\bar{V})$, where $\bar{V} \in \{\, V,\, \widetilde{V},\, [0,1]\,\}$.*

Proof. Using the symmetry of the kernel k_σ we get for the 3D problem

$$\begin{aligned}(\mathcal{K}_\sigma f, g)_{L^2(V)} &= \int_V \int_V k_\sigma(\mathbf{r}-\mathbf{r}')f(\mathbf{r}')\,\mathrm{d}\mathbf{r}'\; g(\mathbf{r})\,\mathrm{d}\mathbf{r} \\ &= \int_V \int_V k_\sigma(\mathbf{r}'-\mathbf{r})g(\mathbf{r})\,\mathrm{d}\mathbf{r}\; f(\mathbf{r}')\,\mathrm{d}\mathbf{r}' \;=\; (f, \mathcal{K}_\sigma g)_{L^2(V)}\ .\end{aligned}$$

Analogous arguments hold for the 2D and 1D case. □

Next, we prove a norm estimate for $\mathcal{K}_\sigma$.

Theorem 2.9. *For $\bar{V} \in \{\, V,\, \widetilde{V},\, [0,1]\,\}$ we have*

$$\|\mathcal{K}_\sigma\|_{\mathscr{L}(L^2(\bar{V}))} \;\leq\; \frac{1}{\sigma}\ ,$$

where $\mathscr{L}\left(L^2(\bar{V})\right)$ denotes the space of linear operators from $L^2(\bar{V})$ to $L^2(\bar{V})$.

Proof. We consider $\mathcal{K}_\sigma g$ as a convolution and use Young's inequality for convolutions (e.g. [55, Theorem 20.18]) to get, for example, for the 3D case

$$\|\mathcal{K}_\sigma g\|_{L^2(V)} \;\leq\; \|k_\sigma * g^e\|_{L^2(\mathbb{R}^3)} \;\leq\; \|k_\sigma\|_{L^1(\mathbb{R}^3)}\|g^e\|_{L^2(\mathbb{R}^3)}\ ,$$

where $*$ denotes convolution and

$$g^e(\mathbf{r}) \;:=\; \begin{cases} g\ , & \text{if } \mathbf{r} \in V \\ 0\ , & \text{if } \mathbf{r} \notin V\ . \end{cases}$$

As $\|g^e\|_{L^2(\mathbb{R}^3)} = \|g\|_{L^2(V)}$ this gives the estimate

$$\|\mathcal{K}_\sigma\|_{\mathscr{L}(L^2(V))} \;\leq\; \|k_\sigma\|_{L^1(\mathbb{R}^3)}\ . \tag{2.17}$$

Now, to estimate the right-hand side of (2.17), we write $\mathbf{x} \in \mathbb{R}^3$ in polar coordinates $\mathbf{x} = s\mathbf{\Omega}\,,\ \mathbf{\Omega} \in \mathbb{S}^2$ to obtain

$$\begin{aligned}\|k_\sigma\|_{L^1(\mathbb{R}^3)} &= \int_{\mathbb{R}^3} \frac{1}{4\pi}\,\frac{\exp(-\sigma\|\mathbf{x}\|_2)}{\|\mathbf{x}\|_2^2}\,\mathrm{d}\mathbf{x} \\ &= \int_0^\infty \int_{\mathbb{S}^2} \frac{1}{4\pi}\,\frac{\exp(-\sigma\|s\mathbf{\Omega}\|_2)}{\|s\mathbf{\Omega}\|_2^2} s^2\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}s \\ &= \int_0^\infty \int_{\mathbb{S}^2} \frac{1}{4\pi}\,\exp(-\sigma s)\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}s \\ &= \int_0^\infty \exp(-\sigma s)\,\mathrm{d}s \\ &= \frac{1}{\sigma}\ .\end{aligned}$$

Therefore, with (2.17),

$$\|\mathcal{K}_\sigma\|_{\mathscr{L}(L^2(V))} \;\leq\; \frac{1}{\sigma}\;. \tag{2.18}$$

Although (2.17) holds in all three spatial dimensions, the derivation of (2.18) using polar coordinates only works in the 2D and 3D case (using similar arguments in the former). In 1D we use the positivity of k_σ to express $\|k_\sigma\|_{L^1(\mathbb{R})}$ as the evaluation of the Fourier transform of k_σ at zero.

$$\|k_\sigma\|_{L^1(\mathbb{R})} \;=\; \int_{-\infty}^{\infty} k_\sigma(x)\,\mathrm{d}x \;=\; \int_{-\infty}^{\infty} \exp(\mathrm{i}x\,0)k_\sigma(x)\,\mathrm{d}x \;=\; \hat{k}_\sigma(0)\;, \tag{2.19}$$

where the exponential Fourier transform of k_σ is given by

$$\hat{k}_\sigma(\xi) \;=\; \int_{-\infty}^{\infty} \exp(\mathrm{i}x\,\xi)k_\sigma(x)\,\mathrm{d}x\;.$$

To find an expression for $\hat{k}_\sigma$ we use that k_σ is related to the exponential integral E_1 with positive real argument $|\sigma x|$ (see Remark 2.7) for which the Fourier transform is known. Using that $k_\sigma(x) = 1/2\,\mathrm{E}_1(|\sigma x|)$ by (2.16), and performing a change of variables, we get

$$\begin{aligned}
\hat{k}_\sigma(\xi) \;&=\; \int_{-\infty}^{\infty} \exp(\mathrm{i}x\,\xi)k_\sigma(x)\,\mathrm{d}x \\
&=\; \frac{1}{2}\int_{-\infty}^{\infty} \exp(\mathrm{i}x\,\xi)\mathrm{E}_1(|\sigma x|)\,\mathrm{d}x \\
&=\; \frac{1}{2\sigma}\int_{-\infty}^{\infty} \exp\left(\mathrm{i}\xi\,\frac{\tilde{x}}{\sigma}\right)\mathrm{E}_1(|\tilde{x}|)\,\mathrm{d}\tilde{x} \\
&=\; \frac{1}{2\sigma}\,\hat{f}\left(\frac{\xi}{\sigma}\right)\;, \quad \text{where} \quad f(x) \;:=\; \mathrm{E}_1(|x|)\;.
\end{aligned}$$

Therefore, we need to find the Fourier transform of $f(x) = \mathrm{E}_1(|x|)$. As f is an even function, the Fourier transform $\hat{f}$ is given by $2\mathcal{F}_c\{f(x);y\}$ (e.g. [33, p. 117]), where $\mathcal{F}_c\{f(x);y\} := \int_0^\infty f(x)\cos(xy)\,\mathrm{d}x$ denotes the Fourier cosine transform. Erdelyi et al. [33, eq. (19), p. 42] now give the corresponding Fourier cosine transform

$$\mathcal{F}_c\{f(x);y\} \;=\; \mathcal{F}_c\{\mathrm{E}_1(x);y\} \;=\; \frac{\tan^{-1}(y)}{y}\;. \tag{2.20}$$

Note that the Fourier transform in equation (19) on page 42 of [33] is stated in terms of the exponential integral $\mathrm{Ei}(-x)$ and not explicitly in terms of $\mathrm{E}_1(x)$. However, the book uses the notation $\mathrm{E}_1(x) = -\mathrm{Ei}(-x)$ for real x (see [33, p. 386]) giving us (2.20). Finally, combining these results, we have

$$\hat{k}_\sigma(\xi) \;=\; \frac{1}{2\sigma}\, 2\mathcal{F}_c\left\{ f(x); \frac{\xi}{\sigma} \right\} \;=\; \frac{1}{\sigma}\, \frac{\tan^{-1}\left(\frac{\xi}{\sigma}\right)}{\frac{\xi}{\sigma}} \;,$$

and using l'Hôspital's rule to calculate the limit of $\hat{k}_\sigma$ as $\xi \to 0$, we obtain

$$\hat{k}_\sigma(0) \;=\; \frac{1}{\sigma} \;.$$

Therefore it follows from (2.19) that

$$\|k_\sigma\|_{L^1(\mathbb{R})} \;=\; \hat{k}_\sigma(0) \;=\; \frac{1}{\sigma} \;.$$

Combining this with the 1D version of (2.17), we get $\|\mathcal{K}_\sigma\|_{\mathscr{L}(L^2([0,1]))} \;\le\; 1/\sigma$. □

Remark 2.10. *The above norm estimate for $\mathcal{K}_\sigma$ does not appear to be well known in the literature. However, the result also follows, using different techniques, from the proof on page 32 in the appendix of [42], where the special case of $\alpha = 0$ has to be considered. In [34, p. 39] it is remarked that for a 1D problem with total cross-section $\sigma = 1$ the operator $\mathcal{K}_\sigma$ is even a contraction in $L^2([0,1])$, i.e. $\|\mathcal{K}_\sigma\|_{\mathscr{L}(L^2([0,1]))} < 1$.*

Next, we show that $\mathcal{K}_\sigma$ is compact. For this we require the first part of [65, Theorem 6, p. 332] which we now state applied to our case for completeness.

Lemma 2.11. *Consider $L^2(D)$, where D is a d-dimensional bounded region in Euclidean space, and define the* kernel of potential type

$$k(\mathbf{r}, \mathbf{r}') \;=\; \frac{b(\mathbf{r}, \mathbf{r}')}{\|\mathbf{r} - \mathbf{r}'\|_2^m} \;,$$

where $b(\mathbf{r}, \mathbf{r}')$ is a bounded function, continuous for $\mathbf{r} \neq \mathbf{r}'$. If

$$d \;>\; m \;, \tag{2.21}$$

then the integral operator

$$(\mathcal{K}g)(\mathbf{r}) \;=\; \int_D k(\mathbf{r}, \mathbf{r}') g(\mathbf{r}') \,\mathrm{d}\mathbf{r}' \;, \quad \mathbf{r} \in D \;,$$

is a compact operator mapping $L^2(D)$ into $L^2(D)$.

Using this lemma we can prove the following result.

Lemma 2.12. *$\mathcal{K}_\sigma$ is compact on $L^2(V)$.*

Proof. We apply Lemma 2.11 to our integral operator $\mathcal{K}_\sigma$. We then have $D = V \subset \mathbb{R}^3$, $b(\mathbf{r}, \mathbf{r}') = \exp(-\sigma\|\mathbf{r} - \mathbf{r}'\|_2)$, $m = 2$ and $d = 3$. These satisfy the assumptions of Lemma 2.11 and therefore $\mathcal{K}_\sigma$ is compact on $L^2(V)$. □

We can also apply the Lemma 2.11 to the 2D problem where the condition (2.21) is satisfied with $D = \widetilde{V} \subset \mathbb{R}^2$, $b(\mathbf{r}, \mathbf{r}') = \exp(-\sigma\|\mathbf{r} - \mathbf{r}'\|_2)$ and $m = 1$, $d = 2$.

However, in the 1D case the kernel is no longer of potential type and we need to use the following result from [56, Theorem 7, p. 51] to prove the compactness of $\mathcal{K}_\sigma$.

Lemma 2.13. *Let $k(x, y)$ be such that*

$$\int_0^1 \int_0^1 |k(x,y)|^2 \, \mathrm{d}x \, \mathrm{d}y \ < \ \infty \ . \tag{2.22}$$

Then the operator

$$(\mathcal{K} f)(x) \ = \ \int_0^1 k(x,y) f(y) \, \mathrm{d}y$$

is a compact operator on $L^2([0,1])$.

We can now prove the compactness of $\mathcal{K}_\sigma$ in the 1D case.

Lemma 2.14. *The operator $\mathcal{K}_\sigma$ in (2.15) is compact on $L^2([0,1])$.*

Proof. We have to show that the kernel k_σ satisfies the condition (2.22), i.e. that

$$\int_0^1 \int_0^1 |k_\sigma(z-t)|^2 \, \mathrm{d}t \, \mathrm{d}z \ < \ \infty \ .$$

For our k_σ we obtain with $\eta := \mu/|x|$, $x \in (0,1)$

$$k_\sigma(x) \ = \ \int_0^1 \exp\left(-\frac{\sigma|x|}{\mu}\right) \frac{\mathrm{d}\mu}{\mu} \ = \ \int_0^{1/|x|} \exp\left(-\frac{\sigma}{\eta}\right) \frac{\mathrm{d}\eta}{\eta} \ =: \ h\left(\frac{1}{|x|}\right) \ .$$

Now we get for $y := 1/|x| \in (1, \infty)$

$$h(y) \ = \ \int_0^y \exp\left(-\frac{\sigma}{\eta}\right) \frac{\mathrm{d}\eta}{\eta} \ = \ \int_0^1 \exp\left(-\frac{\sigma}{\eta}\right) \frac{\mathrm{d}\eta}{\eta} + \int_1^y \exp\left(-\frac{\sigma}{\eta}\right) \frac{\mathrm{d}\eta}{\eta} \ .$$

The first term on the left is just a constant which we denote with C_1. Bounding $\exp(-\sigma/\eta)$ in the second term from above by one gives then

$$h(y) \;=\; C_1 + \int_1^y \exp\left(-\frac{\sigma}{\eta}\right)\frac{\mathrm{d}\eta}{\eta} \;\le\; C_1 + \int_1^y \frac{1}{\eta}\,\mathrm{d}\eta \;=\; C_1 + \ln(y)\ .$$

Hence, for $|x| \in (0,1)$,

$$k_\sigma(x) \;\le\; C_1 + \ln\left(\frac{1}{|x|}\right)\ .$$

Now we can bound $\ln(1/|x|)$ for example by the following expressions, using that the function $x^{\frac{1}{4}}\ln(1/|x|)$ is increasing on $(0,\mathrm{e}^{-4})$ and $\ln(1/|x|)$ is bounded by $1/4$ on $(\mathrm{e}^{-4},1)$, i.e.

$$\ln\left(\frac{1}{|x|}\right) \;\le\; \begin{cases} C_2 x^{-\frac{1}{4}}\ , & \text{if } x \in (0,\mathrm{e}^{-4}) \\ \frac{1}{4}\ , & \text{if } x \in (\mathrm{e}^{-4},1)\ . \end{cases}$$

with $C_2 = 4/\mathrm{e}$, and therefore,

$$k_\sigma(z-t) \;\le\; \begin{cases} C_1 + C_2|z-t|^{-\frac{1}{4}}\ , & \text{if } |z-t| \in (0,\mathrm{e}^{-4}) \\ C_1 + \frac{1}{4}\ , & \text{if } |z-t| \in (\mathrm{e}^{-4},1)\ . \end{cases}$$

Finally,

$$\int_0^1\int_0^1 |k_\sigma(z-t)|^2\,\mathrm{d}t\,\mathrm{d}z \;\le\; \int_0^1\int_0^1 \left|C_1 + C_2|z-t|^{-\frac{1}{4}} + \frac{1}{4}\right|^2 \mathrm{d}t\,\mathrm{d}z \;<\; \infty\ ,$$

and the condition (2.22) is satisfied. Hence $\mathcal{K}_\sigma$ is a compact operator in 1D. □

In the final part of this section we consider whether the operator $\mathcal{K}_\sigma$ is positive. There are many different notions of positivity in the literature and, in fact, we will be using two different concepts ourselves. To distinguish these let us now define what we understand by *positive definite* operators (see also [30, p. 906]).

Definition 2.15. *Let H be a complex Hilbert space and let $\mathcal{A}$ be a bounded operator on H. $\mathcal{A}$ is called* positive definite *if it is self-adjoint and $(\mathcal{A}f,f) > 0$ for all $f \in H\backslash\{0\}$ where $(\,\cdot\,,\,\cdot\,)$ denotes the inner product on H.*

For compact operators we have the following equivalence using the remark in [56, p. 87]:

Lemma 2.16. *If H is a complex Hilbert space and $\mathcal{A}$ is a compact self-adjoint operator on H, then $\mathcal{A}$ is positive definite if and only if all eigenvalues of $\mathcal{A}$ are positive.*

Proof. As $\mathcal{A}$ is a compact self-adjoint operator, the Hilbert-Schmidt theorem (see e.g. [94, p. 268]) tells us that there exists an orthonormal basis $\{e_j\}_{j=1}^{\infty}$ of eigenfunctions of $\mathcal{A}$, so that we can write every $f \in H$ as

$$f \;=\; \sum_{j=1}^{\infty} \xi_j(f) e_j \;,$$

where $\xi_j(f) \;=\; (f, e_j)$. Using that the e_j are eigenfunctions, i.e. that $\mathcal{A}e_j = \omega_j e_j$, we obtain

$$\mathcal{A}f \;=\; \sum_{j=1}^{\infty} \omega_j \xi_j(f) e_j \;,$$

and therefore

$$(\mathcal{A}f, f) \;=\; \sum_{j=1}^{\infty} \omega_j |\xi_j(f)|^2 \;.$$

Now if $(\mathcal{A}f, f) > 0$ for all $f \in H\backslash\{0\}$, we get from using $f = e_j$ that $\omega_j > 0$ for all $j = 1, \ldots, \infty$.

Conversely, for every $f \neq 0$ there exists at least one $J \in \mathbb{N}$ such that $\xi_J(f) \neq 0$. If now $\omega_j > 0$ for all $j = 1, \ldots, \infty$, then this implies that $(\mathcal{A}f, f) > 0$ for all $f \in H\backslash\{0\}$ which finishes the proof. □

We apply this result now to our operator $\mathcal{K}_\sigma$.

Lemma 2.17. *All the eigenvalues of the operator $\mathcal{K}_\sigma$ are positive and hence $\mathcal{K}_\sigma$ is a positive definite operator.*

Proof. Suppose $\mathcal{K}_\sigma f \;=\; \omega f$ for some eigenvalue ω with corresponding eigenfunction $f \in L^2(V)$. As $\mathcal{K}_\sigma$ is self-adjoint, the eigenvalue ω must be real. Let Ψ be the solution of $\mathbf{\Omega} \cdot \nabla\Psi + \sigma\Psi = f$ on $V \times \mathbb{S}^2$, subject to vacuum boundary conditions (2.3). Then, by Lemma 2.3, the corresponding scalar flux satisfies $\phi \;=\; \mathcal{K}_\sigma f \;=\; \omega f$, and we have

$$\begin{aligned}
\omega f^2(\mathbf{r}) &= \phi(\mathbf{r}) f(\mathbf{r}) \\
&= \frac{1}{4\pi} \int_{\mathbb{S}^2} \Psi(\mathbf{r}, \mathbf{\Omega}) f(\mathbf{r}) \, \mathrm{d}\mathbf{\Omega} \\
&= \frac{1}{4\pi} \int_{\mathbb{S}^2} \Psi(\mathbf{r}, \mathbf{\Omega}) \left[\mathbf{\Omega} \cdot \nabla\Psi(\mathbf{r}, \mathbf{\Omega}) + \sigma\Psi(\mathbf{r}, \mathbf{\Omega})\right] \, \mathrm{d}\mathbf{\Omega} \\
&= \frac{1}{4\pi} \int_{\mathbb{S}^2} \mathbf{\Omega} \cdot \left[\Psi(\mathbf{r}, \mathbf{\Omega}) \nabla\Psi(\mathbf{r}, \mathbf{\Omega})\right] \mathrm{d}\mathbf{\Omega} \;+\; \frac{\sigma}{4\pi} \int_{\mathbb{S}^2} \Psi^2(\mathbf{r}, \mathbf{\Omega}) \, \mathrm{d}\mathbf{\Omega} \;.
\end{aligned}$$

Integrating over V and applying the divergence theorem, the first term on the right-hand side becomes

$$\begin{aligned}
\frac{1}{4\pi}\int_{\mathbb{S}^2}\boldsymbol{\Omega}\cdot\left[\int_V \Psi(\mathbf{r},\boldsymbol{\Omega})\nabla\Psi(\mathbf{r},\boldsymbol{\Omega})\,\mathrm{d}\mathbf{r}\right]\mathrm{d}\boldsymbol{\Omega} &= \frac{1}{8\pi}\int_{\mathbb{S}^2}\boldsymbol{\Omega}\cdot\left[\int_V \nabla\left[\Psi^2(\mathbf{r},\boldsymbol{\Omega})\right]\,\mathrm{d}\mathbf{r}\right]\mathrm{d}\boldsymbol{\Omega} \\
&= \frac{1}{8\pi}\int_{\mathbb{S}^2}\int_{\partial V}\Psi^2(\mathbf{r},\boldsymbol{\Omega})\ [\boldsymbol{\Omega}\cdot\mathbf{n}(\mathbf{r})]\ \mathrm{d}\mathbf{r}\,\mathrm{d}\boldsymbol{\Omega} \\
&\geq 0\ ,
\end{aligned}$$

where we used the boundary conditions (2.3) for the final estimate. Hence

$$\omega\int_V f^2(\mathbf{r})\,\mathrm{d}\mathbf{r}\ \geq\ \frac{\sigma}{4\pi}\int_V\int_{\mathbb{S}^2}\Psi^2(\mathbf{r},\boldsymbol{\Omega})\,\mathrm{d}\boldsymbol{\Omega}\,\mathrm{d}\mathbf{r}\ .$$

As $f \neq 0$, both integrals are positive and it follows that $\omega > 0$. Lemma 2.16 then implies that $\mathcal{K}_\sigma$ is positive definite.

For the 2D and 1D problems analogous arguments apply. □

2.2.3 Existence of an orthonormal basis

Now, by the Hilbert-Schmidt theorem for self-adjoint compact operators and Lemma 2.17, $\mathcal{K}_\sigma$ has a sequence of eigenpairs $\{(\omega_j, e_j)\}_{j=1}^{\infty}$, where the sequence $\{\omega_j\}$ is positive, monotone non-increasing and converges to zero as $j\to\infty$. Furthermore, the $\{e_j\}$ form a complete orthonormal basis in $L^2(\bar{V})$ and finally, from Lemma 2.9 and the fact that $\sigma = \sigma_c + \sigma_s + \sigma_f$ (and that all cross-sections are positive), we have

$$\|\sigma_s\mathcal{K}_\sigma\|_{\mathscr{L}(L^2(\bar{V}))}\ \leq\ \frac{\sigma_s}{\sigma}\ <\ 1\ . \qquad (2.23)$$

Combining this with Corollary 2.5 gives the following result.

Lemma 2.18. *The eigenvalues in Corollary 2.5 are*

$$\lambda_j\ =\ \frac{1-\sigma_s\omega_j}{\nu\sigma_f\omega_j}\qquad j=1,2,\dots\ . \qquad (2.24)$$

The sequence $\{\lambda_j\}$ is positive, non-decreasing and tends to infinity as $j\to\infty$.

The eigenvalue of physical interest is the smallest λ_j. Our labelling makes this λ_1, corresponding to w_1. Also of physical interest is the fact that the smallest eigenvalue should be simple and have a positive eigenfunction. We prove this fact for our model problems in the next section.

Remark 2.19. *The relation* (2.24) *between the eigenvalues ω_j of $\mathcal{K}_\sigma$ and the criticality described by λ_1, suggests that computing ω_1 (the largest eigenvalue of $\mathcal{K}_\sigma$) could be a* *viable alternative to computing λ_1. In practice, however, $\mathcal{K}_\sigma$ is not constructed explicitly (although integral transport methods do use a related approach by integrating the angular dependence out and then working on a scalar flux problem (see [80, §5] for further details)). We will use properties of the operator $\mathcal{K}_\sigma$ to obtain theoretical results about the eigenvalue of interest, but we do all our numerical calculations on the non-symmetric eigenvalue problem involving the operators $\mathcal{T}$, $\mathcal{S}$ and $\mathcal{F}$.*

2.3 Simplicity of the smallest eigenvalue

In this section we prove that the smallest eigenvalue λ_1 of our monoenergetic homogeneous model problem with isotropic scattering is simple and has a corresponding strictly positive eigenfunction. This result has been shown in [86] for a more general class of problems, but the analysis there considers the criticality problem via time-dependent problems and makes use of semigroup theory.

The approach presented here only uses simple analytical tools to study the time-independent problem. In [101] it has been stated that by applying the Perron-Frobenius theorem (the finite dimensional equivalent of the Krein-Rutman theorem that we will use in our analysis), the existence of a simple real and positive fundamental eigenvalue with a positive eigenvector can be guaranteed. The paper also mentions the Krein-Rutman theorem as a generalisation of the Perron-Frobenius theorem to the operator case but does not provide details of how to apply it to the criticality problem.

Our argument goes as follows. First, we show the existence of a smallest positive real eigenvalue with a non-negative eigenfunction using a Krein-Rutman argument. Unfortunately, this does not imply the simplicity of the eigenvalue. However, we can use the obtained result, together with the strict positivity of the kernel k_σ, to show that the eigenspace corresponding to λ_1 is one-dimensional.

To be able to do this we need the notion of positive operators on cones. These are different to positive definite operators which were introduced in Definition 2.15.

Definition 2.20 (Cone, see [26, p. 218]). *Let B be a Banach space. A* cone *$C \subset B$ is a closed convex set such that $\alpha f \in C$ for all $\alpha \geq 0$, and all $f \in C$, and $C \cap (-C) = \{0\}$.*

Definition 2.21 (Positive operator, see [69, p. 59]). *Let B be a Banach space with a cone $C \subset B$. The linear operator $\mathcal{A}$ is called* positive *on C if it transforms the cone C into itself, i.e. $\mathcal{A}(C) \subset C$.*

Definition 2.22 (Reproducing and total cones, see [26, Definition 19.1(a)]). *Let B be a Banach space with a cone $C \subset B$. The cone C is* reproducing *if $C - C = B$,* *i.e. if every element in B can be represented as the difference of two elements in C. If $\overline{C - C} = B$, the cone is* total.

In the following we will work with the cone of non-negative functions in $L^2(V)$. This is defined as $L^2_+(V) := \{f \in L^2(V) : f(\mathbf{x}) \geq 0 \ \text{ a.e.}\}$.

Lemma 2.23. *$L^2_+(V)$ is a reproducing and therefore total cone in the Hilbert space $L^2(V)$.*

Proof. Since we can write every function in $L^2(V)$ as the sum of its positive and negative parts, the cone $L^2_+(V)$ is reproducing in $L^2(V)$. Furthermore, every reproducing cone in a Banach space is also a total cone. □

We first show that the generalised eigenvalue problem (2.12) can be written as an eigenvalue problem in standard form.

Lemma 2.24. *The following two problems are equivalent in $L^2(V)$:*

$$\phi(\mathbf{r}) - \sigma_s \mathcal{K}_\sigma \phi(\mathbf{r}) = \lambda \, \nu \sigma_f \mathcal{K}_\sigma \phi(\mathbf{r}) \ , \tag{2.25}$$

$$\mathcal{A}_\sigma \phi(\mathbf{r}) = \frac{1}{\lambda} \phi(\mathbf{r}) \ , \quad \text{where} \quad \mathcal{A}_\sigma := \nu \sigma_f \sum_{n=0}^{\infty} (\sigma_s \mathcal{K}_\sigma)^n \mathcal{K}_\sigma \ . \tag{2.26}$$

Proof. We note that the left-hand side of (2.25) can be written as $(\mathcal{I} - \sigma_s \mathcal{K}_\sigma)\phi$, where $\mathcal{I}$ denotes the identity operator on $L^2(V)$. From (2.23) we know that $\sigma_s \mathcal{K}_\sigma$ is a contraction and we can therefore compute the inverse of the operator $(\mathcal{I} - \sigma_s \mathcal{K}_\sigma)$ using the convergent Neumann series

$$(\mathcal{I} - \sigma_s \mathcal{K}_\sigma)^{-1} = \sum_{n=0}^{\infty} (\sigma_s \mathcal{K}_\sigma)^n \ .$$

Rearranging the scalar flux eigenvalue problem (2.25) gives then the standard eigenvalue problem (2.26). □

In the following we show that there exists a largest positive eigenvalue $(\lambda_1)^{-1}$ of $\mathcal{A}_\sigma$ from (2.26), which then corresponds to a smallest eigenvalue λ_1 of (2.25). We start with the following observation.

Lemma 2.25. *$\mathcal{K}_\sigma$ is positive on $L^2_+(V)$, i.e. $\mathcal{K}_\sigma(L^2_+(V)) \subset L^2_+(V)$. Moreover, it maps non-zero non-negative functions to strictly positive functions, such that*

$$\mathcal{K}_\sigma(L^2_+(V) \backslash \{0\}) \subset L^2_{++}(V) \ ,$$

where $L^2_{++}(V) := \{f \in L^2(V) : f(\mathbf{x}) > 0 \;\; a.e.\} \subset L^2_+(V)$ *.*

Proof. As k_σ is a strictly positive kernel, the first part obviously holds. If in addition $g \in L^2_+(V)\backslash\{0\}$, then there exists a set of positive measure such that $g > 0$ on that set and

$$(\mathcal{K}_\sigma g)(\mathbf{r}) \;=\; \int_V k_\sigma(\mathbf{r}-\mathbf{r}')g(\mathbf{r}')\,\mathrm{d}\mathbf{r}' \;>\; 0 \quad \text{for all} \quad \mathbf{r} \in V \;,$$

which proves the second part of the lemma. □

Using that all cross-sections as well as ν are positive, we obtain that the operator $\mathcal{A}_\sigma$ given in (2.26) also maps non-zero non-negative functions to strictly positive functions.

Corollary 2.26. *The operator* $\mathcal{A}_\sigma$ *in* (2.26) *is positive on the cone* $L^2_+(V)$ *and satisfies*

$$\mathcal{A}_\sigma(L^2_+(V)\backslash\{0\}) \;\subset\; L^2_{++}(V) \;.$$

We now use the Krein-Rutman theorem as stated in [26] to prove that $\|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))}$ is the largest positive eigenvalue of $\mathcal{A}_\sigma$ and has an eigenfunction in $L^2_+(V)$.

Theorem 2.27 (Krein-Rutman theorem [26, Theorem 19.2])**.** *Let B be a Banach space, $C \subset B$ a total cone and $\mathcal{A} \in \mathscr{L}(B)$ be a compact and positive operator on C (in the sense of Definition 2.21, i.e. $\mathcal{A}(C) \subset C$) with spectral radius $r(\mathcal{A}) > 0$. Then $r(\mathcal{A})$ is an eigenvalue with an eigenfunction $f \in C$.*

Using this we can prove the following result.

Theorem 2.28. *The scalar flux eigenvalue problem*

$$\phi(\mathbf{r}) - \sigma_s \mathcal{K}_\sigma \phi(\mathbf{r}) \;=\; \lambda\, \nu \sigma_f \mathcal{K}_\sigma \phi(\mathbf{r}) \tag{2.27}$$

has a smallest positive real eigenvalue λ_1 with non-negative eigenfunction $\phi_1 \in L^2_+(V)$.

Proof. We apply Theorem 2.27 to the operator $\mathcal{A}_\sigma = \nu\sigma_f \sum_{n=0}^\infty (\sigma_s \mathcal{K}_\sigma)^n \mathcal{K}_\sigma$ from the standard eigenvalue problem (2.26) using the cone $C = L^2_+(V)$. By Corollary 2.26, $\mathcal{A}_\sigma$ is a positive operator on $L^2_+(V)$.

Since $\nu\sigma_f \sum_{n=0}^\infty (\sigma_s \mathcal{K}_\sigma)^n$ is a bounded operator and $\mathcal{K}_\sigma$ is compact, the composition $\mathcal{A}_\sigma$ is also compact. The set of non-zero eigenvalues for non-zero self-adjoint compact operators on Hilbert spaces is non-empty (e.g. [98, Theorem 7.33]), and hence the spectral radius $r(\mathcal{A}_\sigma)$ is positive.

Theorem 2.27 then yields that the largest eigenvalue $r(\mathcal{A}_\sigma) = \|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))} =: (\lambda_1)^{-1}$ is real and positive, and that there exists a non-negative scalar flux $\phi_1 \in L^2_+(V)$ corresponding to this eigenvalue. Lemma 2.24 finally implies that λ_1 is the smallest positive real eigenvalue of the scalar flux problem (2.27) with corresponding eigenfunction $\phi_1 \in L^2_+(V)$. □

We can extend this result to the angular flux problem.

Corollary 2.29. *The angular flux eigenvalue problem*

$$\mathbf{\Omega} \cdot \nabla\Psi(\mathbf{r},\mathbf{\Omega}) + \sigma\Psi(\mathbf{r},\mathbf{\Omega}) - \frac{\sigma_s}{4\pi}\int_{\mathbb{S}^2}\Psi(\mathbf{r},\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}' \;=\; \lambda\,\frac{\nu\sigma_f}{4\pi}\int_{\mathbb{S}^2}\Psi(\mathbf{r},\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}' \qquad (2.28)$$

has a smallest positive real eigenvalue λ_1 with corresponding non-negative eigenfunction $\Psi_1 \in L^2(V, L^1(\mathbb{S}^2))$.

Proof. We know by Theorem 2.28 that the scalar flux problem (2.27) has a smallest positive real eigenvalue λ_1 with corresponding non-negative eigenfunction $\phi_1 \in L^2_+(V)$. Now from Corollary 2.5 we obtain a solution $\Psi_1 \in L^2(V, L^1(\mathbb{S}^2))$ of

$$\mathcal{T}\Psi_1(\mathbf{r},\mathbf{\Omega}) \;=\; \sigma_s\mathcal{K}_\sigma\phi_1(\mathbf{r}) + \lambda_1\,\nu\sigma_f\mathcal{K}_\sigma\phi_1(\mathbf{r})\;, \qquad (2.29)$$

which is an eigenfunction for the angular flux problem (2.28) corresponding to the eigenvalue λ_1. As the right-hand side of (2.29) is non-negative, the formula (2.9) yields a non-negative function Ψ_1 which finishes the proof. □

Remark 2.30. *The above proof does not show that λ_1 is a* simple *eigenvalue. The stronger version of the Krein-Rutman theorem given by [26, Theorem 19.3], which would ensure simplicity of the eigenvalue, demands that the interior of the cone is non-empty. This is not satisfied for the cone $L^2_+(V)$ as for every $f \in L^2_+(V)$ and every $\epsilon > 0$, we can construct a function $\tilde{f}$ (by changing f on a small set of positive measure to negative values) which still ensures $\|f - \tilde{f}\|_{L^2} < \epsilon$ but is not contained in $L^2_+(V)$. This means that there exists no $\epsilon > 0$ such that the ball $B(f,\epsilon) := \{\, f' \in L^2(V) \;:\; \|f - f'\|_{L^2} < \epsilon\,\}$ is in $L^2_+(V)$. Hence there is no open neighbourhood of f contained in $L^2_+(V)$ and therefore the interior is empty.*

Fortunately, our integral operator $\mathcal{K}_\sigma$ has a strictly positive kernel and, as we saw in Corollary 2.26, $\mathcal{A}_\sigma$ therefore maps non-zero non-negative functions to *strictly* positive functions. This fact will help us to prove that λ_1 is simple.

We use for our argument a result about self-adjoint positive operators on cones from [116] which we adapt to our situation. In [116] real Hilbert spaces are considered, remarking that if H is a complex Hilbert space with inner product $(\cdot\,,\,\cdot)$, then H is also a real Hilbert space with a real inner product

$$(f,g)_{\mathrm{R}} \;:=\; \mathrm{Real}(f,g)\;, \quad f,g \in H\;.$$

Note that our operators $\mathcal{K}_\sigma$ and $\mathcal{A}_\sigma$ map real-valued functions to real-valued functions, they are self-adjoint and there exists a basis of real eigenfunctions in the space $L^2(V)$ that we consider. Hence, we may restrict ourselves to real Hilbert spaces without loss of generality. We keep the $(\cdot\,,\,\cdot)$ notation for the inner product assuming implicitly that it agrees with the real inner product.

Definition 2.31. *Let H be a real Hilbert space with inner product $(\cdot\,,\,\cdot)$ and let C be a reproducing cone. Then define for each $f \in H$ the* projection *$\pi_C(f)$ as the unique element of the cone C which is closer to f than any other element in C.*

Now, if $f \in H$ and $p \in C$, then $p = \pi_C(f)$ if and only if

$$(g-p, f-p) \;\le\; 0 \quad \textit{for all} \quad g \in C\;. \tag{2.30}$$

Analogously to [116, Lemma 1] we have the following equivalence.

Lemma 2.32. *If H is a real Hilbert space, C is a cone in H, $f \in H$ and $p \in C$, then $p = \pi_C(f)$ if and only if*

$$\sup_{g\in C}(g, f-p) \;=\; (p, f-p) \;=\; 0\;. \tag{2.31}$$

Proof. Let $p = \pi_C(f)$ so that (2.30) holds. Setting $g = 0$ and $g = 2p$ in (2.30), we deduce that $(p, f-p) = 0$. Now using this result in (2.30), we get

$$0 \;\ge\; (g-p, f-p) \;=\; (g, f-p) \quad \text{for all} \quad g \in C\;,$$

and therefore (2.31) holds. Conversely, if (2.31) holds, we have for all $g \in C$

$$(g-p, f-p) \;=\; (g, f-p) - (p, f-p) \;\le\; 0\;,$$

and (2.30) holds. Hence $p = \pi_C(f)$. □

The following result, analogous to [116, Lemma 2], will be key to our proof that the smallest eigenvalue of (2.27) is simple.

Lemma 2.33. *Let C be a reproducing cone in a real Hilbert space H and let $\mathcal{A}$ be a self-adjoint operator on H that maps C into itself. Furthermore, assume that the spectral radius of $\mathcal{A}$ is an eigenvalue, i.e. $\mathcal{A}f = \|\mathcal{A}\|f$ for some $f \neq 0$, and let $p = \pi_C(f)$. If $p \neq 0$, then p is also an eigenfunction for the eigenvalue $\|\mathcal{A}\|$, i.e.*

$$\mathcal{A}p \;=\; \|\mathcal{A}\|p \; . \tag{2.32}$$

Proof. We use the fact that $\mathcal{A}p \in C$. Then from (2.31) with $g = \mathcal{A}p$, and the self-adjointness of $\mathcal{A}$, we get

$$\begin{aligned} 0 \;&\geq\; (\mathcal{A}p, f - p) \\ &=\; (p, \mathcal{A}f) - (\mathcal{A}p, p) \\ &=\; \|\mathcal{A}\|(p, f - p + p) - (\mathcal{A}p, p) \\ &=\; \|\mathcal{A}\|\|p\|^2 - (\mathcal{A}p, p) \;, \end{aligned}$$

where the final equality is obtained by applying (2.31) again. Together with the Cauchy-Schwarz inequality this gives

$$\|\mathcal{A}\|\|p\|^2 \;\leq\; (\mathcal{A}p, p) \;\leq\; \|\mathcal{A}p\|\|p\| \;\leq\; \|\mathcal{A}\|\|p\|^2 \;,$$

and hence $\|\mathcal{A}\|\|p\|^2 = (\mathcal{A}p, p)$. Finally,

$$\begin{aligned} 0 \;&\leq\; \|(\mathcal{A}p - \|\mathcal{A}\|p)\|^2 \\ &=\; \|\mathcal{A}p\|^2 - 2\|\mathcal{A}\|(\mathcal{A}p, p) + \|\mathcal{A}\|^2\|p\|^2 \\ &=\; \|\mathcal{A}p\|^2 - \|\mathcal{A}\|^2\|p\|^2 \\ &\leq\; \|\mathcal{A}\|^2\|p\|^2 - \|\mathcal{A}\|^2\|p\|^2 \;=\; 0 \;, \end{aligned}$$

and therefore (2.32) holds. □

We can now prove our main result of this section.

Theorem 2.34. *The scalar flux eigenvalue problem* (2.27) *has a* simple *smallest positive real eigenvalue λ_1 with strictly positive eigenfunction $\phi_1 \in L^2_{++}(V)$.*

Proof. By Lemma 2.24 we know that (2.27) is equivalent to the standard eigenvalue problem (2.26). From Theorem 2.28 we obtained that the largest positive real eigenvalue $\|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))}$ of problem (2.26) has an eigenfunction in the cone $L^2_+(V)$. We now need to show that $\|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))}$ is a simple eigenvalue and the corresponding eigenfunction actually lies in $L^2_{++}(V)$.

Assume $\|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))}$ is not a simple eigenvalue. Then there exists a two-dimensional subspace of eigenfunctions. As $L^2_+(V)$ is a cone, $L^2_+(V) \cap (-L^2_+(V)) = \{0\}$, and we can construct a non-zero eigenfunction $\widetilde{\phi}$ outside the cone $L^2_+(V)$ such that

$$\mathcal{A}_\sigma \widetilde{\phi} \;=\; \|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))} \widetilde{\phi} \quad \text{and} \quad \pi_{L^2_+(V)}(\widetilde{\phi}) \;\neq\; 0 \;.$$

Now, since $\widetilde{\phi} \notin L^2_+(V)$, its projection $\pi_{L^2_+(V)}(\widetilde{\phi})$ is zero on a set of positive measure. On the other hand, applying Lemma 2.33, we get that

$$\mathcal{A}_\sigma(\pi_{L^2_+(V)}(\widetilde{\phi})) \;=\; \|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))} (\pi_{L^2_+(V)}(\widetilde{\phi})) \;. \tag{2.33}$$

Corollary 2.26 states that the left-hand side of (2.33) is in $L^2_{++}(V)$ and therefore strictly positive almost everywhere, but by construction the right-hand side of (2.33) is zero on a set of positive measure and we obtain a contradiction. Hence the eigenspace corresponding to the eigenvalue $\|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))}$ must be one-dimensional.

Let now $\phi_1 \in L^2_+(V)$ be an eigenfunction in the one-dimensional eigenspace. Then

$$\mathcal{A}_\sigma \phi_1 \;=\; \|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))} \phi_1 \;, \tag{2.34}$$

and applying Corollary 2.26 again yields that the left-hand side of (2.34) is in $L^2_{++}(V)$. As $\|\mathcal{A}_\sigma\|_{\mathscr{L}(L^2(V))} > 0$ this implies that $\phi_1 \in L^2_{++}(V)$.

Using Lemma 2.24 and the same argument as in Theorem 2.28, we therefore obtain that $\lambda_1 = \|\mathcal{A}_\sigma\|^{-1}_{\mathscr{L}(L^2(V))}$ is a simple smallest eigenvalue with strictly positive eigenfunction $\phi_1 \in L^2_{++}(V)$ for the scalar flux eigenvalue problem (2.27). □

Applying similar arguments as in the proof of Corollary 2.29 we get, due to the strict positivity of the integrand in (2.9), the final result of this section.

Corollary 2.35. *The smallest eigenvalue* λ_1 *of the eigenvalue problem* (2.28) *is positive, real and simple and has a corresponding eigenfunction* $\Psi_1(\mathbf{r}, \mathbf{\Omega}) \in L^2(V, L^1(\mathbb{S}^2))$ *which is strictly positive in the interior of* V*.*

Remark 2.36. *The simplicity of the smallest eigenvalue and the existence of a strictly positive eigenfunction are also proved in [118, Theorem II, p. 77] and [42, Theorem 5 with* $\alpha = 0$ *], but both references make use of a different Krein-Rutman theorem to the one applied here.*

2.4 Discretisation techniques

Several different discretisation schemes have been developed and discussed in the literature, some of which we already mentioned in Section 1.5. We now describe the discrete ordinates approach in more detail, indicating in Section 2.4.2 how a popular spatial discretisation scheme in 1D can fail to retain the underlying symmetry that we observed in the previous sections. We propose symmetry preserving discretisations and finish this section with discretisation error estimates for a scheme that we employ in our numerical tests. We focus here on one-dimensional problems while the 2D and 3D cases have been considered, for example, in [29, §8.1] and [19].

2.4.1 Discrete ordinates

As mentioned in Section 1.5, the idea of the discrete ordinates approach is to approximate the angular flux at a set of spatial mesh points and in a number of fixed directions. The directions are often denoted as ordinates giving the discretisation scheme its name.

We now discuss how to fully discretise the 1D criticality problem

$$\mu \frac{\partial}{\partial z} \Psi(z,\mu) \;+\; \sigma \Psi(z,\mu) \;-\; \frac{\sigma_s}{2} \int_{-1}^{1} \Psi(z,\mu')\,\mathrm{d}\mu' \;=\; \lambda\, \frac{\nu \sigma_f}{2} \int_{-1}^{1} \Psi(z,\mu')\,\mathrm{d}\mu' \tag{2.35}$$

with vacuum boundary conditions (2.5).

The first step is to perform an angular discretisation to approximate the integrals. A quadrature rule with $2N$ points $\{\mu_k\} \subset [-1,1]\backslash\{0\}$, and weights $\{w_k\}$ for indices $k = -N, -(N-1), \ldots, -1, 1, \ldots, (N-1), N$ is applied to approximate

$$\int_{-1}^{1} f(\mu)\,\mathrm{d}\mu \;\approx\; \sum_{|i|=1}^{N} w_i f(\mu_i) \;.$$

The quadrature points and weights are chosen symmetrically so that

$$\begin{aligned} 0 \;<\; \mu_1 \;<\; \mu_2 \;<\; \ldots \;<\; \mu_N \;\leq\; 1 \;, \\ \mu_i \;=\; -\mu_{-i} \;, \quad i = 1, \ldots, N \;, \quad \text{and} \\ w_i \;=\; w_{-i} \;, \quad i = 1, \ldots, N \;, \end{aligned} \tag{2.36}$$

which corresponds physically to treating fluxes in the positive and negative direction with equal importance. Furthermore, since we are working with non-negative functions, we demand that the weights are positive.

Standard approaches are Gaussian quadrature rules on either the two separate intervals $[-1,0]$ and $[0,1]$ or choosing $2N$ Gauss-Legendre points on $[-1,1]$. We use the latter for our numerical experiments.

Applying the angular discretisation to the 1D problem (2.35) and evaluating the resulting semidiscrete problem at the $2N$ quadrature points μ_k, $|k| = 1,\dots,N$, leads to the following system of ordinary differential equations for the semidiscrete solution $\Psi_N \in L^2([0,1])$:

$$\begin{aligned} \mu_k \frac{\partial}{\partial z}\Psi_N(z,\mu_k) \,+\, \sigma\Psi_N(z,\mu_k) \,-\, \frac{\sigma_s}{2}\sum_{|i|=1}^{N} w_i\Psi_N(z,\mu_i) \\ = \lambda\,\frac{\nu\sigma_f}{2}\sum_{|i|=1}^{N} w_i\Psi_N(z,\mu_i)\ , \quad \text{where} \end{aligned} \tag{2.37}$$

$$\Psi_N(0,\mu_k) \;=\; 0\ ,\ k = 1,\dots,N \quad \text{and} \quad \Psi_N(1,\mu_k) \;=\; 0\ ,\ k = -1,\dots,-N\ . \tag{2.38}$$

To arrive at a fully discrete problem we need to approximate the spatial derivative. Again, several different schemes have been applied in the literature (see e.g. [76, 77, 92] and the references therein). Usually, a finite difference or finite element method is used in space. A popular approach is the Crank-Nicolson scheme. This is one of the methods that we use for the numerical results in this thesis and now discuss in more detail. Define a spatial mesh z_j, $j = 0,\dots,M$, on $[0,1]$ with intervals $I_j = [z_{j-1}, z_j]$, $j = 1,\dots,M$, and lengths $h_j = z_j - z_{j-1}$.

We follow [93] for the application of the Crank-Nicolson scheme to (2.37), (2.38). Let $V^h = \{\, v \in C^0[0,1] \ :\ v|_{I_j} \in P_1(I_j),\ j = 1,\dots,M \,\}$ be the space of continuous piecewise linear functions on the interval $[0,1]$. The Crank-Nicolson scheme is to approximate Ψ_N from (2.37) with $\Psi_N^h \in V^h$, such that Ψ_N^h satisfies

$$\begin{aligned} \int_{I_j} \mu_k \frac{\partial}{\partial z}\Psi_N^h(z,\mu_k)\,\mathrm{d}z \,+\, \sigma\int_{I_j}\Psi_N^h(z,\mu_k)\,\mathrm{d}z \,-\, \frac{\sigma_s}{2}\sum_{|i|=1}^{N} w_i\int_{I_j}\Psi_N^h(z,\mu_i)\,\mathrm{d}z \\ = \lambda\,\frac{\nu\sigma_f}{2}\sum_{|i|=1}^{N} w_i\int_{I_j}\Psi_N^h(z,\mu_i)\,\mathrm{d}z\ , \quad j = 1,\dots,M\ , \quad \text{and} \\ \Psi_N^h(0,\mu_k) \;=\; 0\ ,\ k = 1,\dots,N \quad \text{and} \quad \Psi_N^h(1,\mu_k) \;=\; 0\ ,\ k = -1,\dots,-N\ . \end{aligned} \tag{2.39}$$

This gives the usual Crank-Nicolson discretisation of (2.37)

$$\begin{aligned}\mu_k\big(\Psi_N^h(z_j,\mu_k) - \Psi_N^h(z_{j-1},\mu_k)\big) \;+\; & \frac{\sigma h_j}{2}\big(\Psi_N^h(z_{j-1},\mu_k) + \Psi_N^h(z_j,\mu_k)\big) \\ & - \;\frac{\sigma_s h_j}{4}\sum_{|i|=1}^{N} w_i\big(\Psi_N^h(z_{j-1},\mu_i) + \Psi_N^h(z_j,\mu_i)\big) \\ & = \;\lambda\,\frac{\nu\sigma_f h_j}{4}\sum_{|i|=1}^{N} w_i\big(\Psi_N^h(z_{j-1},\mu_i) + \Psi_N^h(z_j,\mu_i)\big)\;.\end{aligned}$$

Together with the discretised boundary conditions (2.39) this can be written as a generalised eigenvalue problem in matrix form (see Section 3.3.3 for an example).

Note that the discrete ordinates scheme is not restricted to the criticality problem and that the same steps can be applied to obtain a discretisation for source problems.

2.4.2 Symmetry preserving discretisations

We established in Section 2.2 that the non-symmetric criticality problem (2.2) for the angular flux is equivalent to the *symmetric* scalar flux problem (2.12). We will see in Chapter 3 that it is advantageous for iterative eigenvalue solvers to preserve the symmetry in the continuous problem by the discretisation scheme. Unfortunately, the above Crank-Nicolson approach does not retain the underlying symmetry and this can result in slower convergence of iterative methods as we will observe in the numerical results in Section 3.4.

While a full study of symmetry-preserving discretisations is beyond the scope of this thesis, we show here by two examples, that natural symmetry-preserving discretisations do exist. First we consider the semidiscrete case of (2.4) and (2.5), where we discretise only with respect to the spatial variable z and leave the angular variable μ continuous. The discrete approximation to the operator $\mathcal{K}_\sigma = \mathcal{P}\mathcal{T}^{-1}$ is obtained by applying the inverse of the discrete version of $\mathcal{T}$ and then integrating over μ. This turns out to be symmetric in the discrete spatial variable when certain conditions are met. In addition we describe how to preserve the symmetry under further discretisation with respect to the angular variable μ. In both examples below, analogously to Lemma 2.1, we consider for any $g \in L^2([0,1])$ discrete versions of the problem

$$\mathcal{T}\Psi(z,\mu) \;=\; \mu\frac{\partial}{\partial z}\Psi(z,\mu) + \sigma\Psi(z,\mu) \;=\; g(z)\;, \quad z\in[0,1]\;, \quad \mu\in[-1,1]\;, \tag{2.40}$$

subject to the vacuum boundary conditions (2.5).

Symmetry preserving Euler scheme

For the first example we use a uniform spatial mesh $z_j = jh$, $j = 0,\ldots,M$ with $h = 1/M$, and an Euler-type method (i.e. a first order finite difference approximation of the derivative, also known as the step method [80, §3]). Integrating from left to right for $\mu > 0$ and from right to left for $\mu < 0$ gives

$$\begin{aligned}
\mu\,\frac{\Psi(z_j,\mu)-\Psi(z_{j-1},\mu)}{h} + \sigma\Psi(z_{j-1},\mu) &= g(z_{j-1}) \quad \text{for} \quad \mu > 0\ ,\ j = 1,\ldots,M\ ,\\
\mu\,\frac{\Psi(z_j,\mu)-\Psi(z_{j-1},\mu)}{h} + \sigma\Psi(z_j,\mu) &= g(z_j) \quad \text{for} \quad \mu < 0\ ,\ j = 1,\ldots,M\ ,
\end{aligned}$$

with $\Psi(z_0,\mu) = 0$ when $\mu > 0$ and $\Psi(z_M,\mu) = 0$ when $\mu < 0$.

These equations can be written as the two linear systems

$$A^+(\mu)\mathbf{\Psi}^+(\mu) \;=\; \mathbf{g}^+ \quad \text{for} \quad \mu > 0 \quad \text{and} \quad A^-(\mu)\mathbf{\Psi}^-(\mu) \;=\; \mathbf{g}^- \quad \text{for} \quad \mu < 0\ ,$$

where $\mathbf{\Psi}^+(\mu) := (\Psi(z_1,\mu),\ldots,\Psi(z_M,\mu))^T$, $\mathbf{g}^+ := (g(z_0),\ldots,g(z_{M-1}))^T$, $\mathbf{\Psi}^-(\mu) := (\Psi(z_0,\mu),\ldots,\Psi(z_{M-1},\mu))^T$ and $\mathbf{g}^- := (g(z_1),\ldots,g(z_M))^T$, and where $A^+(\mu)$ is a lower bidiagonal matrix and $A^-(\mu)$ is an upper bidiagonal matrix given respectively by

$$A^+(\mu) := \begin{bmatrix} \frac{\mu}{h} & & & \\ \left(\sigma - \frac{\mu}{h}\right) & \frac{\mu}{h} & & \\ & \ddots & \ddots & \\ & & \left(\sigma - \frac{\mu}{h}\right) & \frac{\mu}{h} \end{bmatrix}, \ A^-(\mu) := \begin{bmatrix} -\frac{\mu}{h} & \left(\sigma + \frac{\mu}{h}\right) & & \\ & \ddots & \ddots & \\ & & -\frac{\mu}{h} & \left(\sigma + \frac{\mu}{h}\right) \\ & & & -\frac{\mu}{h} \end{bmatrix}.$$

Note that $A^-(\mu)$ and $A^+(\mu)$ are both non-singular and

$$A^-(-\mu) \;=\; \left(A^+(\mu)\right)^T \quad \text{and so} \quad \left(A^-(-\mu)\right)^{-1} \;=\; \left(A^+(\mu)\right)^{-T}\ . \tag{2.41}$$

This condition plays a crucial role in our proof to show that this difference scheme preserves the underlying symmetry. With $\mathbf{\Psi}(\mu) := (\Psi(z_0,\mu),\ldots,\Psi(z_M,\mu))^T$ we reduce the problem now to one in terms of scalar fluxes. For $\boldsymbol{\phi} := (\phi(z_0),\ldots,\phi(z_M))^T$ and $\mathbf{g} := (g(z_0),\ldots,g(z_M))^T \in \mathbb{R}^{M+1}$ we get

$$\begin{aligned}
\boldsymbol{\phi} \;&=\; \int_{-1}^{1} \mathbf{\Psi}(\mu)\,\mathrm{d}\mu \;=\; \int_0^1 \begin{pmatrix} \mathbf{0} & \left(A^-(-\mu)\right)^{-1} \\ 0 & \mathbf{0}^T \end{pmatrix} \mathbf{g}\,\mathrm{d}\mu + \int_0^1 \begin{pmatrix} \mathbf{0}^T & 0 \\ \left(A^+(\mu)\right)^{-1} & \mathbf{0} \end{pmatrix} \mathbf{g}\,\mathrm{d}\mu \\
&=\; \widetilde{K}\mathbf{g}\ ,
\end{aligned}$$

where

$$\widetilde{K} \ := \ \left[\int_0^1 A(\mu)\,\mathrm{d}\mu\right] \quad \text{and} \quad A(\mu) \ := \ \begin{pmatrix} \mathbf{0} & \left(A^-(-\mu)\right)^{-1} \\ 0 & \mathbf{0}^T \end{pmatrix} + \begin{pmatrix} \mathbf{0}^T & 0 \\ \left(A^+(\mu)\right)^{-1} & \mathbf{0} \end{pmatrix} .$$

The symmetry of $\widetilde{K}$ then follows from (2.41).

Finally, let us now consider the full discretisation and suppose we choose a quadrature rule with points $\{\mu_k\} \subset [-1,1]\backslash\{0\}$, and weights $\{w_k\}$ that satisfy the conditions (2.36). We then obtain $\boldsymbol{\phi} = K\mathbf{g}$ with a symmetric $K := \sum_{k=1}^N w_k A(\mu_k)$.

Finite element scheme

As an alternative to the finite difference method above, consider an arbitrary mesh $0 = z_0 < z_1 < \ldots < z_M = 1$. For $i = 0,\ldots,M$, let φ_i denote the usual continuous piecewise linear "hat" functions on $[0,1]$, satisfying $\varphi_i(z_j) = \delta_{i,j}$. Furthermore, set $h_j = z_j - z_{j-1}$ and define $(\,\cdot\,,\,\cdot\,)$ to be the standard inner product on $L^2([0,1])$.

Now use the following approximation for (2.40). If $\mu > 0$, we approximate $\Psi(z,\mu)$ by $\Psi^+(z,\mu) := \sum_{j=1}^M \Psi_j^+(\mu)\varphi_j(z)$ and determine the coefficients $\Psi_j^+(\mu)$ by requiring

$$\left(\mu\frac{\partial}{\partial z}\Psi^+(\cdot,\mu) + \sigma\Psi^+(\cdot,\mu)\,,\,\varphi_{i-1}\right) \ = \ (g,\varphi_{i-1}) \ , \quad i = 1,\ldots,M \ .$$

This is equivalent to the $M \times M$ system

$$A^+(\mu)\boldsymbol{\Psi}^+(\mu) \ = \ \mathbf{g}^+ \ ,$$

where $\boldsymbol{\Psi}^+(\mu) := \left(\Psi_1^+(\mu),\ldots,\Psi_M^+(\mu)\right)^T$, $\mathbf{g}^+ := \left((g,\varphi_0),\ldots,(g,\varphi_{M-1})\right)^T$ and

$$\left(A^+(\mu)\right)_{i,j} \ := \ \mu(\varphi_j',\varphi_{i-1}) + \sigma(\varphi_j,\varphi_{i-1}) \ .$$

Similarly, for $\mu < 0$ approximate $\Psi(z,\mu)$ by $\Psi^-(z,\mu) := \sum_{j=0}^{M-1}\Psi_j^-(\mu)\varphi_j(z)$ defined by

$$\left(\mu\frac{\partial}{\partial z}\Psi^-(\cdot,\mu) + \sigma\Psi^-(\cdot,\mu)\,,\,\varphi_i\right) \ = \ (g,\varphi_i) \ , \quad i = 1,\ldots,M \ .$$

This is equivalent to

$$A^-(\mu)\boldsymbol{\Psi}^-(\mu) \ = \ \mathbf{g}^- \ ,$$

where $\boldsymbol{\Psi}^-(\mu) := \left(\Psi_0^-(\mu),\ldots,\Psi_{M-1}^-(\mu)\right)^T$, $\mathbf{g}^- := \left((g,\varphi_1),\ldots,(g,\varphi_M)\right)^T$ and

$$\left(A^-(\mu)\right)_{i,j} \ := \ \mu(\varphi_{j-1}',\varphi_i) + \sigma(\varphi_{j-1},\varphi_i) \ .$$

The matrices $A^{+}(\mu)$ are lower tridiagonal with positive diagonal while the matrices $A^{-}(\mu)$ are upper tridiagonal with positive diagonal and hence $A^{\pm}(\mu)$ are non-singular. Now we notice that by integration by parts $(\varphi'_{i-1}, \varphi_j) = -(\varphi'_j, \varphi_{i-1})$, and so the crucial condition (2.41) is also satisfied by the matrices $A^{\pm}(\mu)$. Therefore, this finite element method also has the property that the symmetry of the discrete version of the operator $\mathcal{P}\mathcal{T}^{-1}$ is preserved if the quadrature points are chosen according to (2.36).

2.4.3 Discretisation error estimates

We now briefly describe the discretisation error when using $2N$ Gauss-Legendre quadrature points on $[-1, 1]$ and a Crank-Nicolson scheme for the spatial approximation. The results that we state are a special case of Theorems 4.4 and 4.5 in [93]. The paper [93] also considers more general conditions on the quadrature rules as well as a discontinuous Galerkin scheme for the approximation of the spatial derivative. Discretisation error estimates for two dimensional problems have been studied, for example, in [63] with an extension to the analysis in [5] and [6].

The discretisation error results which are given below allow for non-uniform meshes, where $h := \max_{j=1,\ldots,M} h_j$ (with $h_j = z_j - z_{j-1}$) denotes the maximal distance between two spatial points. For a 1D source problem for the scalar flux [93, Theorem 4.4] gives the following result.

Theorem 2.37. *Let $\phi \in L^2([0,1])$ be the solution to the source problem*

$$\phi(z) - \gamma\mathcal{K}_\sigma\phi(z) \;=\; \mathcal{K}_\sigma Q(z)\ ,$$

which is the integral equation form of

$$\Psi(z,\mu) + \sigma\Psi(z,\mu) - \gamma\int_{-1}^{1}\Psi(z,\mu')\mathrm{d}\mu' \;=\; Q(z) \tag{2.42}$$

with source $Q \in C^{2+\epsilon}([0,1])$ for arbitrary $\epsilon > 0$. The notation $Q \in C^{2+\epsilon}([0,1])$ denotes that Q is twice continuously differentiable on $[0,1]$ and the second derivative is Hölder continuous with exponent ϵ.

Let $\gamma < \rho(\mathcal{K}_\sigma)$, where $\rho(\mathcal{K}_\sigma)$ is the spectral radius of the operator $\mathcal{K}_\sigma$, and define

$$\phi_N^h(z) \;=\; \sum_{|i|=1}^{N} w_i\Psi_N^h(z,\mu_i)\ ,$$

where Ψ_N^h is the discrete ordinates approximation to the solution of (2.42), which is *obtained using $2N$ Gauss-Legendre quadrature points on $[-1,1]$ with corresponding weights and the Crank-Nicolson scheme for the spatial discretisation.*

If the spatial mesh size $h = h(N)$ is chosen such that $h(N)|\log(N)| \to 0$ for $N \to \infty$, then the following estimate for the discretisation error of the scalar flux holds:

$$\|\phi - \phi_N^h\|_{L^2([0,1])} \;\leq\; C(\phi, Q)\left(N^{-\frac{3}{2}} + h^2 N^{\frac{1}{2}}\right) , \tag{2.43}$$

where $C(\phi, Q) = C_\epsilon\big(\|\phi\|_{L^2([0,1])} + \|Q\|_{C^{2+\epsilon}([0,1])}\big)$ for arbitrary $\epsilon > 0$.

Proof. The result is obtained by applying Theorem 4.4 in [93] for the case of the Crank-Nicolson method, $i = 1$ and $p = 2$. Note that the λ defined in the paper is in our notation γ and not related to our eigenvalue λ which in turn corresponds to $\lambda_{\min}$ in the paper. □

If we choose, for example, $\gamma = \sigma_s$, we have the source problem $(\mathcal{T} - \mathcal{S})\Psi = Q$ which includes neutron scatter but no fission. For the criticality problem [93, Theorem 4.5] provides the following result.

Theorem 2.38. *Let (λ, ϕ) be an eigenpair in $L^2([0,1])$ of the generalised eigenvalue problem (2.12), and let (λ_N^h, ϕ_N^h) be an eigenpair in $L^2([0,1])$ of the approximation*

$$\phi_N^h(\mathbf{r}) - \sigma_s K \phi_N^h(\mathbf{r}) \;=\; \lambda_N^h\, \nu\sigma_f K \phi_N^h(\mathbf{r}) \;, \quad \mathbf{r} \in V \;,$$

where K is the discrete version of $\mathcal{K}_\sigma$ after applying the $2N$ Gauss-Legendre quadrature rule on $[-1,1]$ and the Crank-Nicolson scheme for the spatial discretisation.

If the spatial mesh size $h = h(N)$ satisfies the same criterion as in Theorem 2.37, then the discretisation error estimate (2.43) holds for the eigenfunctions ϕ_N^h and we have the eigenvalue error estimate

$$|\lambda - \lambda_N^h| \;\leq\; C\left(N^{-2} + \log(N) h^2\right) .$$

These results show that the solution of the discrete problem converges to the true solution when the number of spatial and angular discretisation points tends to infinity. The estimates also suggest that the optimal relation between h and N is

$$h \;=\; h(N) \;=\; N^{-1} .$$

We will therefore choose the number of spatial and angular points of the same magnitude for the numerical tests in this thesis.

Chapter 3

Iterative methods for criticality computations

In the previous chapter we established the existence of a simple smallest positive real eigenvalue λ of the criticality problem

$$(\mathcal{T} - \mathcal{S})\Psi \quad = \quad \lambda \mathcal{F}\Psi \tag{3.1}$$

subject to vacuum boundary conditions (2.3), where the special form of the operators $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{F}$ from page 6 was given in Section 2.1.

This chapter now discusses iterative methods to compute the eigenvalue of interest. The first section describes four such algorithms and introduces the concept of inexact solves. Section 3.2 then provides a convergence analysis for the use of inexact inverse iteration to compute the criticality of neutron transport problems of the form (2.2), (2.3). This is done by exploiting the underlying symmetry of the problem that we have observed in the previous chapter and by transferring recently developed techniques for the analysis of matrix eigenvalue problems (in particular from [14]) to the continuous eigenvalue problem (3.1).

In Section 3.3 we present two 1D model problems of different complexity and give details of a discretisation which is used in our computations. The final section then provides numerical results that support the convergence theory from Section 3.2.

3.1 Description of iterative methods

In this section we describe four related iterative methods to compute the criticality of a nuclear reactor. As discussed in Section 1.2, the criticality is determined by the smallest positive real eigenvalue λ of the generalised eigenvalue problem (3.1). In the case of a discretisation of the problem we are usually working with large sparse linear systems, where so-called *direct* eigenvalue solvers (like the QR method), that approximate the whole eigenvalue spectrum, are expensive or not feasible to use. On the other hand matrix-vector multiplications can be performed efficiently for sparse matrices. Often we also have other ways of "cheaply" applying the operators $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{F}$ (for example in the case of the method of characteristics (see Section 1.5) or by using Monte Carlo techniques (see Chapter 4)) and we are, in general, only interested in one, i.e. the smallest, eigenvalue. Therefore, we focus on so-called *iterative methods* in this thesis.

These methods consist of an *outer iteration* in which the eigenvalue approximation is updated and where a linear (source) problem has to be solved. This solve is usually done iteratively itself, leading to an *inner iteration.* In the following we will discuss examples of these *inner-outer iteration methods* to solve the criticality problem (3.1).

Note that matrix free engineering techniques often only perform one (rough) inner solve (without necessarily satisfying a stopping criterion based on the residual of the source problem) before continuing the outer iteration. This is, for example, done in CACTUS, a module in the WIMS code developed by the ANSWERS software group of Serco Technical and Assurance Services[1]. The approach is based on the method of characteristics and can be used to compute the reactor criticality. Another example, where only one inner iteration is performed, is the version of the method of perturbation which we will describe in Section 5.5.

For theoretical purposes we focus on iterative methods where the outer (eigenvalue) iteration is separated from inner (source problem) iterations, and inner and outer tolerances are satisfied by the respective solvers. We now consider some well-known iterative methods for eigenvalue problems and apply them to the criticality problem (3.1).

3.1.1 Power method

The most basic method for solving eigenvalue problems is the *power method* (e.g. [45, 91, 99]). In order to apply this to our criticality problem we formally transform (3.1)

[1] http://www.sercoassurance.com/answers/

into the standard eigenvalue problem

$$(\mathcal{T}-\mathcal{S})^{-1}\mathcal{F}\Psi \;=\; \frac{1}{\lambda}\Psi\ . \tag{3.2}$$

The power method consists now of iteratively applying the operator on the left of (3.2) to the current eigenfunction approximation $\Psi^{(i)}$, followed by a normalisation of the result to give the next iterate $\Psi^{(i+1)}$.

Algorithm 1 Power method

```
Require: Starting guess $\Psi^{(0)}$ .
  for i=0,1,2,... do
    Compute $\widetilde{\Psi}^{(i+1)} = (\mathcal{T}-\mathcal{S})^{-1}\mathcal{F}\Psi^{(i)}$ .
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$ .
  end for
```

The outer iteration in Algorithm 1 – and in all other algorithms in this chapter – is stopped when the eigenvalue residual

$$\text{res}^{(i)} \;:=\; (\mathcal{T}-\mathcal{S}-\rho^{(i)}\mathcal{F})\Psi^{(i)} \tag{3.3}$$

is sufficiently small in some suitable norm, where $\rho^{(i)}$ is the standard Rayleigh quotient on $L^2(V\times\mathbb{S}^2)$ given by

$$\rho^{(i)} \;=\; \frac{\langle\Psi^{(i)},(\mathcal{T}-\mathcal{S})\Psi^{(i)}\rangle}{\langle\Psi^{(i)},\mathcal{F}\Psi^{(i)}\rangle}\ . \tag{3.4}$$

The inner product $\langle\,\cdot\,,\,\cdot\,\rangle$ denotes again integration over all independent variables.

For the matrix case it is well known (e.g. [99, Theorem 4.1]) that this approach converges to the eigenvector corresponding to the largest eigenvalue in modulus, provided that (i) this eigenvalue is simple; (ii) the initial guess $\boldsymbol{\Psi}^{(0)}$ contains a component in the eigendirection; and (iii) the linear systems are solved exactly.

We proved in Section 2.3 that our smallest eigenvalue λ of (3.1) (which corresponds to the largest $1/\lambda$ of (3.2)) is simple and has a corresponding eigenfunction that is strictly positive in the interior of the reactor. Hence, when using a strictly positive starting guess, the power method provides a suitable way of solving the criticality problem in neutron transport theory.

Let λ_j denote now the different eigenvalues of (3.1), where $\lambda_1<\lambda_2\le\lambda_3\le\ldots$. The convergence of the power method in Algorithm 1 to the desired λ_1 depends then on the ratio λ_1/λ_2 (see, for example, the proof of Theorem 4.1 in [99] for the matrix case) and is very slow for problems with $\lambda_1 \approx \lambda_2$. However, the convergence can be improved by using a shift that is close to the eigenvalue of interest.

This leads to our next method which was, according to [60], introduced in 1944 by Wielandt for computing eigenfunctions of linear operators [121] and subsequently turned by Wilkinson into a numerical method for computing eigenvectors of matrices. It is today generally called (shifted) inverse iteration [61], but in the context of Monte Carlo methods for neutron transport problems it is still referred to as the "Wielandt method" or "Wielandt acceleration" (e.g. [18, 125]).

3.1.2 Shifted inverse iteration

By applying the power method to the *shifted* problem $(\mathcal{T} - \mathcal{S} - \alpha\mathcal{F})\Psi = (\lambda - \alpha)\mathcal{F}\Psi$ and assuming that the shift α is chosen such that $0 < |\lambda_1 - \alpha| < |\lambda_2 - \alpha| \leq |\lambda_j - \alpha|$, $j > 2$, we obtain from Section 3.1.1 that this yields a method where the convergence speed is determined by $|\lambda_1 - \alpha|/|\lambda_2 - \alpha|$. Hence, for an appropriately chosen shift α, the convergence of this method can be faster than the standard power method. This method, called *shifted inverse iteration*, provides an important tool in practical applications as well as in the analysis of more complicated iterative eigenvalue methods as we will see later.

If we are solving criticality problems for reactor safety computations, we expect the desired smallest eigenvalue to be around unity and a fixed shift $\alpha = 1$ can be applied. Alternatively, a variable shift can be used. This is usually taken from a suitable eigenvalue approximation (e.g. a Rayleigh quotient) derived from $\Psi^{(i)}$, when $\Psi^{(i)}$ is rich in a certain eigendirection. One option is to choose the shift to be the standard Rayleigh quotient from (3.4). We will suggest an alternative choice for a non-standard Rayleigh quotient shift in Lemma 3.4 below.

Algorithm 2 Shifted inverse iteration

```
Require: Starting guess $\Psi^{(0)}$.
  for i=0,1,2,... do
    Define a shift $\alpha^{(i)}$.
    Compute $\widetilde{\Psi}^{(i+1)}$ such that $(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\widetilde{\Psi}^{(i+1)} = \mathcal{F}\Psi^{(i)}$.
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$.
  end for
```

Note that the power method in Algorithm 1 can be interpreted as a special case of shifted inverse iteration in Algorithm 2, where a fixed shift $\alpha^{(i)} = 0$ for all i is used.

As mentioned above the solutions of the linear systems which are to be solved within the different methods are usually computed iteratively. This often leads to only approximate solutions and we can formalise these *inexact methods* by replacing, for example, the linear system

$$(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\widetilde{\Psi}^{(i+1)} \;=\; \mathcal{F}\Psi^{(i)}$$

in Algorithm 2 by the requirement to seek $\widetilde{\Psi}^{(i+1)}$ such that

$$\|(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\widetilde{\Psi}^{(i+1)} \;-\; \mathcal{F}\Psi^{(i)}\| \;\leq\; \tau^{(i)} \;, \tag{3.5}$$

where $\tau^{(i)}$ is a given tolerance and $\|\cdot\|$ is a suitable norm.

Although standard iterative solvers often check the *relative* residual condition

$$\frac{\|(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\widetilde{\Psi}^{(i+1)} \;-\; \mathcal{F}\Psi^{(i)}\|}{\|\mathcal{F}\Psi^{(i)}\|} \;\leq\; \tau^{(i)} \;,$$

and not the absolute stopping criterion (3.5), we will restrict our theory to the case of an absolute residual condition to simplify the analysis.

Inexact iterative methods applied to matrix eigenvalue problems of the form

$$A\mathbf{x} \;=\; \lambda B\mathbf{x} \;, \quad A, B \in \mathbb{R}^{n\times n} \;, \tag{3.6}$$

have received a large amount of attention in recent years and substantial progress regarding the convergence analysis of these methods has been made. The paper [70], for example, contains a proof for linear convergence of an inexact inverse iteration method for non-Hermitian matrices A, B with diagonalisable $B^{-1}A$, when a fixed shift and decreasing tolerances for the inner problems are used. The technical proof expands the iterates $\mathbf{x}^{(i)}$ in terms of the eigenvectors of the problem and gives a condition on the size of the inner tolerances which depends on the distance of the shift to the second closest eigenvalue, an unknown quantity for most problems. The symmetric problem has been considered in [107], where a geometric approach is used for the analysis.

A well-known orthogonal splitting (see e.g. [91, p. 63]) for a unit vector $\mathbf{x}^{(i)} \in \mathbb{R}^n$ approximating the eigenvector $\mathbf{e}_1$ of (3.6) is

$$\mathbf{x}^{(i)} \;=\; c^{(i)}\mathbf{e}_1 + s^{(i)}\mathbf{u}^{(i)} \;,$$

where $A\mathbf{e}_j = \lambda_j B\mathbf{e}_j$, $j = 1, \ldots, n$, $\mathbf{u}^{(i)} \in \operatorname{span}(\mathbf{e}_2, \ldots, \mathbf{e}_n)$, $\mathbf{e}_1 \perp \mathbf{u}^{(i)}$ and $\|\mathbf{u}^{(i)}\| = 1$. Here $s^{(i)} = s(\mathbf{x}^{(i)})$ and $c^{(i)} = c(\mathbf{x}^{(i)})$ are the sine and cosine of the angle between $\mathbf{e}_1$ and the current iterate $\mathbf{x}^{(i)}$. They satisfy $(s^{(i)})^2 + (c^{(i)})^2 = 1$ and are used to define the tangent $t^{(i)} := |s^{(i)}|/|c^{(i)}|$. This splitting approach is, for example, employed in [14] for the real symmetric eigenvalue problem

$$A\mathbf{x} \;=\; \lambda\mathbf{x}\;, \quad A \in \mathbb{R}^{n\times n}\;.$$

The analysis presented covers the case of variable and fixed shifts, as well as the use of fixed and decreasing tolerances for the inner solve. First a one-step bound on $t^{(i+1)}$ is derived which is then applied to different choices of inner tolerances and shifts. This results, for example, in cubic convergence if a Rayleigh quotient shift and certain decreasing tolerances are used. Applying fixed inner tolerances the Rayleigh quotient iteration is still shown to converge quadratically for this symmetric problem. We will use similar ideas and a similar structure as in that paper in our convergence analysis in Section 3.2.

The paper [16] extends the results from [70] for the problem (3.6) with a non-symmetric matrix A but symmetric positive definite B to the case of variable shifts. The proof uses the orthogonal splitting

$$\mathbf{x}^{(i)} \;=\; \alpha^{(i)}(c^{(i)}\mathbf{e}_1 + s^{(i)}\mathbf{u}^{(i)})\;,$$

where $\alpha^{(i)}$ is a scalar defined by $\mathbf{x}^{(i)}$. The analysis provides convergence results for different shifts and inner tolerances, showing, for example, quadratic convergence for a fixed shift and suitably decreasing tolerances.

Improvements of the bounds in [16] and generalisations of the analysis that do not require B to be symmetric positive definite, or even non-singular, are given in [38]. The matrices in that paper are also allowed to be complex. The very general results use a decomposition of the eigenvector approximations in terms of the exact eigenvector $\mathbf{e}_1$ and a basis of an invariant subspace (see also [114, §4]).

Further papers that consider inexact inverse iteration and focus on the use of particular iterative solvers for the solution of the inner problems are, for example, [15] for GMRES, [97] for the conjugate gradient method and [39] for MINRES. The paper [39] also presents a tuned preconditioner which avoids the growth in the number of inner iterations as the outer iteration proceeds.

A different approach, that does not use an eigenvector expansion or orthogonal splitting employed by most other papers, is taken in [37]. There, convergence estimates of an inexact inverse iteration algorithm are obtained using the relation between inverse iteration and Newton's method. The suggested shift strategy and normalisation given in [37] allow a reformulation of inexact inverse iteration as a modified Newton's method and it is shown that for decreasing tolerances $\tau^{(i)} \leq C\|\mathrm{res}^{(i)}\|_\infty$ and a sufficiently "good" starting guess the (local) quadratic convergence of Newton's method can be retained. If a fixed inner tolerance is chosen, the given algorithm converges linearly.

A block version of inverse iteration, called inverse subspace iteration, that aims to compute an invariant subspace of a matrix, has been analysed in [95]. The arising linear systems

$$(A - \alpha I)\widetilde{X}^{(i+1)} \;=\; X^{(i)} \;, \quad A \in \mathbb{C}^{n\times n} \;,\; X^{(i)}, \widetilde{X}^{(i+1)} \in \mathbb{C}^{n\times p}$$

with fixed shift α are assumed to be solved inexactly. The paper gives conditions on the tolerances for the inner solves which ensure that linear convergence is retained. Furthermore, a tuning of the preconditioner for the block-GMRES inner solver is discussed that can avoid the increase in cost of the iterative solves as the outer iteration proceeds even though the inner tolerances are decreasing.

3.1.3 Inverse correction

The method that we introduce in this section is called *inverse correction*. This has been discussed, for example, in [96] and, as a form of inexact inverse iteration, in [46]. It is based on computing a *correction* $\Delta\Psi^{(i+1)}$ to the current iterate $\Psi^{(i)}$, such that $\widetilde{\Psi}^{(i+1)} = \Psi^{(i)} + \Delta\Psi^{(i+1)}$ yields a better approximation to the true eigenfunction than $\Psi^{(i)}$. In every iteration we have to solve the *correction equation*

$$(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\Delta\Psi^{(i+1)} \;=\; -\mathrm{res}^{(i)} \;, \tag{3.7}$$

where the eigenvalue residual $\mathrm{res}^{(i)}$ is given by (3.3).

It is important that the shift $\alpha^{(i)}$ is not exactly the Rayleigh quotient $\rho^{(i)}$ which is used for the computation of the residual $\mathrm{res}^{(i)}$. Otherwise (3.7) becomes

$$(\mathcal{T} - \mathcal{S} - \rho^{(i)}\mathcal{F})\Delta\Psi^{(i+1)} \;=\; -(\mathcal{T} - \mathcal{S} - \rho^{(i)}\mathcal{F})\Psi^{(i)} \;,$$

and for exact solves this results in $\Delta\Psi^{(i+1)} = -\Psi^{(i)}$. Computing $\widetilde{\Psi}^{(i+1)} = \Psi^{(i)} + \Delta\Psi^{(i+1)}$ then leads to $\widetilde{\Psi}^{(i+1)}$ becoming zero. In [96] a shift that is a slight perturbation of the Rayleigh quotient (e.g. $\alpha^{(i)} = 0.95\rho^{(i)}$) is suggested.

This method is closely related to inverse iteration with shift as the next lemma shows. To avoid confusion we denote the solution of the linear system in Algorithm 2 (inverse iteration) by $\widetilde{\Psi}_{\mathrm{II}}^{(i+1)}$ and the next iterate before normalisation in Algorithm 3 (inverse correction) by $\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)} = \Psi^{(i)} + \Delta\Psi^{(i+1)}$.

Algorithm 3 Inverse correction

```
Require: Starting guess $\Psi^{(0)}$.
  for i=0,1,2,... do
    Compute the residual $\mathrm{res}^{(i)} = (\mathcal{T} - \mathcal{S} - \rho^{(i)}\mathcal{F})\Psi^{(i)}$.
    Define a shift $\alpha^{(i)}$.
    Compute $\Delta\Psi^{(i+1)}$ such that $(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\Delta\Psi^{(i+1)} = -\mathrm{res}^{(i)}$.
    Define $\widetilde{\Psi}^{(i+1)} = \Psi^{(i)} + \Delta\Psi^{(i+1)}$.
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$.
  end for
```

Lemma 3.1. *Suppose Algorithms 2 and 3 use the same shifts $\alpha^{(i)}$ with $\alpha^{(i)} \neq \rho^{(i)}$. Furthermore, assume the linear systems are solved exactly and the same normalisation is applied. Then the algorithms produce, up to a different sign, the same iterates.*

Proof. The next iterate of Algorithm 3 is computed from

$$\begin{aligned}
\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)} &= \Psi^{(i)} + \Delta\Psi^{(i+1)} \\
&= \Psi^{(i)} - (\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})^{-1}(\mathcal{T} - \mathcal{S} - \rho^{(i)}\mathcal{F})\Psi^{(i)} \\
&= \Psi^{(i)} - (\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})^{-1}[(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\Psi^{(i)} + (\alpha^{(i)} - \rho^{(i)})\mathcal{F}\Psi^{(i)}] \\
&= (\rho^{(i)} - \alpha^{(i)})(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})^{-1}\mathcal{F}\Psi^{(i)} \\
&= (\rho^{(i)} - \alpha^{(i)})\widetilde{\Psi}_{\mathrm{II}}^{(i+1)} .
\end{aligned}$$

If in both methods the same normalisation is used, we therefore get

$$\Psi_{\mathrm{IC}}^{(i+1)} = \frac{\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)}}{\|\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)}\|} = \frac{\rho^{(i)} - \alpha^{(i)}}{|\rho^{(i)} - \alpha^{(i)}|}\frac{\widetilde{\Psi}_{\mathrm{II}}^{(i+1)}}{\|\widetilde{\Psi}_{\mathrm{II}}^{(i+1)}\|} = \pm\Psi_{\mathrm{II}}^{(i+1)} .$$

Hence Algorithms 2 and 3 produce the same iterates up to a different sign. □

Section 3.5 of the thesis [13] considers the relation between inverse iteration and inverse correction for inexact solves. Lemma 3.14 in that thesis provides conditions for the standard eigenvalue problem under which it is shown that the iterates of inexact inverse correction are contained in a set of solutions to the linear problems of inexact inverse iteration with an adjusted inner tolerance. Using the convergence theory for inverse iteration derived earlier in the thesis, it is then shown that inexact inverse correction converges to the desired eigenvalue.

We consider variations of Algorithms 2 and 3 in Section 5.2 of this thesis, where we show (in agreement with the result in [13]) that there exists an equivalence between the inexact forms of the two methods, provided that the inner tolerances are adjusted in a certain way. For further details about inexact inverse correction and its efficiency we refer the reader to [13, §3.5].

A recent method that follows the idea of inverse correction to generate a fast algorithm for the simultaneous computation of p eigenvalues and eigenvectors of the complex matrix eigenvalue problem

$$A\mathbf{x} \;=\; \lambda B\mathbf{x}\ , \quad A, B \in \mathbb{C}^{n\times n}$$

has been discussed in [124]. The new approach consists of a two-phase strategy for solving the inner block linear systems

$$A\widetilde{X}^{(i+1)} \;=\; BX^{(i)}\ , \quad X^{(i)}, \widetilde{X}^{(i+1)} \in \mathbb{C}^{n\times p} \tag{3.8}$$

by applying a *tuned* preconditioner (which changes in every outer iteration) to perform one single block-GMRES step and thereby obtain an approximate solution $\widetilde{X}_1^{(i+1)}$ to (3.8). Then the correction equation

$$A\Delta X^{(i+1)} \;=\; BX^{(i)} - A\widetilde{X}_1^{(i+1)} \tag{3.9}$$

is solved iteratively (second phase of the inner iteration process) using a *fixed* preconditioner until $\widetilde{X}^{(i+1)} = \widetilde{X}_1^{(i+1)} + \Delta X^{(i+1)}$ satisfies the stopping criterion for the inner solve. This approach allows further strategies to reduce the costs for the inner solve (such as obtaining a special starting vector for (3.9) as a linear combination of previous solutions).

3.1.4 Simplified Jacobi-Davidson

Another approach that follows the idea of computing a correction to the current iterate is the *Jacobi-Davidson method* (see e.g. [105, 106]). It additionally demands that the correction is orthogonal to the space that is spanned by all previous iterates. The *simplified Jacobi-Davidson method* reduces this requirement by enforcing orthogonality only to the current iterate.

The latter approach is also known as the *Newton-Grassmann method* and is, for example, discussed in [104]. There it is proved that, for the standard eigenvalue problem with Hermitian matrices, this method is equivalent to Rayleigh quotient iteration if no preconditioner and exact solves of the linear problem are used. This equivalence also holds for inexact solves with the same Galerkin-Krylov subspace method (e.g. the CG method). The paper [40] remarks that the result also extends to the non-Hermitian case of the standard eigenvalue problem. In addition [40] presents a tuning for the preconditioner of the Rayleigh quotient iteration using which it is shown that the two methods are equivalent even if preconditioned inexact solves are performed.

An extension to generalised eigenvalue problems is given in [38, 41], where the link between Rayleigh quotient iteration and the simplified Jacobi-Davidson method is exploited to obtain convergence results for the latter. The reference [38] also provides a sample algorithm (Algorithm 6.1 in [38]) which we adapted to our criticality problem (3.1) to obtain Algorithm 4.

Algorithm 4 Simplified Jacobi-Davidson

```
Require: Starting guess $\Psi^{(0)}$ .
  for i=0,1,2,... do
    Compute the residual $\mathrm{res}^{(i)} = (\mathcal{T} - \mathcal{S} - \rho^{(i)}\mathcal{F})\Psi^{(i)}$ .
    Define a shift $\alpha^{(i)}$ and projections $\mathcal{Q}_l^{(i)}$ and $\mathcal{Q}_r^{(i)}$ so that for all $g \in L^2(V, L^1(\mathbb{S}^2))$
    $\mathcal{Q}_l^{(i)} g = g - \frac{\langle g, \mathcal{F}\Psi^{(i)}\rangle}{\langle \mathcal{F}\Psi^{(i)}, \mathcal{F}\Psi^{(i)}\rangle}(\mathcal{F}\Psi^{(i)})$ and $\mathcal{Q}_r^{(i)} g = g - \frac{\langle \mathcal{F}\Psi^{(i)}, \mathcal{F}g\rangle}{\langle \mathcal{F}\Psi^{(i)}, \mathcal{F}\Psi^{(i)}\rangle}\Psi^{(i)}$ .
    Compute $\Delta\Psi^{(i+1)}$ such that $\mathcal{Q}_l^{(i)}(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\mathcal{Q}_r^{(i)}\Delta\Psi^{(i+1)} = -\mathrm{res}^{(i)}$ .
    Define $\widetilde{\Psi}^{(i+1)} = \Psi^{(i)} + \Delta\Psi^{(i+1)}$ .
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$ .
  end for
```

The orthogonal projections $\mathcal{Q}_l^{(i)}$ and $\mathcal{Q}_r^{(i)}$ in Algorithm 4 are used to find a solution $\Delta\Psi^{(i+1)}$ such that $\langle \mathcal{F}\Delta\Psi^{(i+1)}, \mathcal{F}\Psi^{(i)}\rangle = 0$, i.e. that $\mathcal{F}\Delta\Psi^{(i+1)}$ is orthogonal to $\mathcal{F}\Psi^{(i)}$.

A review of the Jacobi-Davidson method including recent developments is given in [58]. The Jacobi-Davidson method has also lately been applied to compute generalized singular values and singular vectors of large sparse matrices (see [57]).

Numerical comparison of the iterative methods

Tables 3.1 and 3.2 now show numerical results for the four iterative methods described above when applied to a discretisation of the control rod insertion model problem which we will describe in Section 3.3.2. The resulting generalised matrix eigenvalue problem is non-symmetric with a singular right-hand side. The first and second column of the table contain the name of the method and the shift that is applied. As remarked above the power method can be considered as an application of inverse iteration with a fixed shift $\alpha^{(i)} = 0$.

The columns "outer" and "inner" denote the number of outer iterations needed for the solution of the eigenvalue problem and the average number of iterations that were needed to solve the inner source problems. The final column contains the average computing time of the method taken from 10 consecutive runs.

For the inner solves we used the GMRES function in MATLAB with an LU factorisation of the matrix representing the transport operator $\mathcal{T}$ as preconditioner. In the first test the tolerance for the inner solves is fixed to $\tau_0 = 10^{-14}$. This is also used as outer tolerance.

When applying the preconditioner in the simplified Jacobi-Davidson method, we need to take care to also apply the matrix equivalents of the orthogonal projections $\mathcal{Q}_l^{(i)}$ and $\mathcal{Q}_r^{(i)}$, i.e.

$$Q_l^{(i)} = I - \frac{(F\mathbf{\Psi}^{(i)})(F\mathbf{\Psi}^{(i)})^T}{(F\mathbf{\Psi}^{(i)})^T(F\mathbf{\Psi}^{(i)})} \quad \text{and} \quad Q_r^{(i)} = I - \frac{\mathbf{\Psi}^{(i)}(F\mathbf{\Psi}^{(i)})^T F}{(F\mathbf{\Psi}^{(i)})^T(F\mathbf{\Psi}^{(i)})} ,$$

correctly to the preconditioner. To achieve this we have to adjust the GMRES function in MATLAB for this case. Details about the orthogonalisation of the preconditioner for the standard eigenvalue problem are given in [40] and [9, §4].

	$\alpha^{(i)}$	outer	inner	time
Power method (inverse iteration)	0	6	22	28.4
Inverse iteration	1	4	23	23.3
Inverse iteration	$\rho^{(i)}$	3	26	21.3
Inverse correction	1	4	26	23.7
Inverse correction	$0.95\rho^{(i)}$	4	27	23.8
Simplified Jacobi-Davidson	1	4	25	21.4
Simplified Jacobi-Davidson	$\rho^{(i)}$	3	19	18.7

Table 3.1: *Comparison of the numerical methods in Algorithms 1 to 4 for a fixed inner tolerance $\tau_0 = 10^{-14}$ applied to the control rod problem presented in Section 3.3.2.*

The results in Table 3.1 show that all methods need very similar numbers of inner and outer iterations when the linear systems are solved almost exactly. However, when solving the inner systems only inexactly, we obtain different results as seen in Table 3.2. The tolerance for the (outer) eigenvalue residual remained 10^{-14} while the inner tolerances were increased to $\tau_0 = 0.1$.

The power method and inverse iteration with a fixed shift $\alpha^{(i)} = 1$ failed to converge in this case while the number of outer iterations for the other methods increased. Rayleigh quotient iteration needed the smallest number of outer iterations but with an average of 14 inner iterations the inner solver had to work hardest. The correction based approaches (inverse correction and simplified Jacobi-Davidson) only needed on average four inner iterations to satisfy the stopping criterion. The reason for this appears to

	$\alpha^{(i)}$	outer	inner	time
Power method (inverse iteration)	0	-	-	-
Inverse iteration	1	-	-	-
Inverse iteration	$\rho^{(i)}$	6	14	22.7
Inverse correction	1	13	4	30.2
Inverse correction	$0.95\rho^{(i)}$	12	4	29.0
Simplified Jacobi-Davidson	1	13	4	22.1
Simplified Jacobi-Davidson	$\rho^{(i)}$	9	4	20.0

Table 3.2: *Comparison of the numerical methods in Algorithms 1 to 4 applied to the control rod problem from Section 3.3.2 using a fixed inner tolerance* $\tau_0 = 0.1$.

be that the default starting vector for the inner solver in GMRES is the zero vector which provides a good initial guess to the solution of the correction equation.

In the next section we give a convergence analysis for the operator case of inexact inverse iteration for the criticality problem.

3.2 Convergence theory for inexact inverse iteration

The key to analysing the convergence properties of iterative methods for the criticality problem as described in Section 2.1, is to use the link of the non-symmetric problem (2.2) for the angular flux to the self-adjoint scalar flux problem (2.12). We can then take a similar approach to the one in [14] to obtain convergence results for inexact inverse iteration.

Note that we are working here with operators on infinite dimensional function spaces whereas [14], and the other references mentioned in Section 3.1, considered the case of finite-dimensional matrices. Fortunately, due to $\mathcal{K}_\sigma$ being self-adjoint and compact on $L^2(V)$, we have analogous tools to the case in [14] at our access.

Using that the eigenfunctions $\{e_j\}_{j=1}^{\infty}$ of $\mathcal{K}_\sigma$, which we introduced in Section 2.2.3, form an orthonormal basis of $L^2(V)$, we can write any $\phi \in L^2(V)$ as

$$\phi \;=\; \sum_{j=1}^{\infty} \xi_j(\phi) e_j \;, \tag{3.10}$$

where $\xi_j(\phi) \;=\; (\phi, e_j)_{L^2(V)}$. Parseval's equality gives that

$$\|\phi\|^2_{L^2(V)} \;=\; \sum_{j=1}^{\infty} |\xi_j(\phi)|^2 \;=\; c(\phi)^2 + s(\phi)^2 \;, \tag{3.11}$$

where $c(\phi) := |\xi_1(\phi)|$ and $s(\phi)^2 := \sum_{j=2}^{\infty} |\xi_j(\phi)|^2$. Analogous to the finite dimensional case $s(\phi)$ and $c(\phi)$ can be interpreted as generalised sine and cosine of the angle between ϕ and e_1. Finally, we define the generalised tangent $t(\phi) := s(\phi)/c(\phi)$, which we will use to estimate the convergence of the iterative method. Note that if $\phi \to e_1$, then $s(\phi) \to 0$ and $c(\phi) \to 1$, so that $t(\phi) \to 0$.

In Algorithm 5 we present inexact inverse iteration for the criticality problem (3.1). When approximately solving the linear system for the next iterate (step (†) below), we measure the residual of the linear system using the following scalar quantity. For $v \in L^2(V, L^\infty(\mathbb{S}^2))$ we set

$$\|v\|_* \;=\; \|\mathcal{P}\mathcal{T}^{-1}v\|_{L^2(V)} \ , \tag{3.12}$$

which is well-defined by Lemma 2.1. Moreover, Lemma 2.3 tells us that if $v(\mathbf{r}, \mathbf{\Omega}) = v(\mathbf{r})$, for all $(\mathbf{r}, \mathbf{\Omega}) \in V \times \mathbb{S}^2$, we have $\|v\|_* = \|\mathcal{K}_\sigma v\|_{L^2(V)}$, so that $\|\cdot\|_*$ acts as a norm on the subspace of all functions in $L^2(V, L^\infty(\mathbb{S}^2))$ that are constant with respect to their second argument.

Algorithm 5 Inexact shifted inverse iteration

```
Require: Starting guess $\Psi^{(0)}$ .
  for i=0,1,2,... do
    Choose a shift $\alpha^{(i)}$ and an inner tolerance $\tau^{(i)} \geq 0$ .
    Compute $\widetilde{\Psi}^{(i+1)}$ such that $\|(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\widetilde{\Psi}^{(i+1)} - \mathcal{F}\Psi^{(i)}\|_* \leq \tau^{(i)}$ .    (†)
    Normalise $\Psi^{(i+1)} = \widetilde{\Psi}^{(i+1)}/\|\mathcal{P}\widetilde{\Psi}^{(i+1)}\|_{L^2(V)}$ .
  end for
```

In this algorithm we implicitly require $\Psi^{(i)}, \widetilde{\Psi}^{(i)} \in L^2(V, L^1(\mathbb{S}^2))$. A simple application of Lemma 2.3 proves the following lemma.

Lemma 3.2. *If $\widetilde{\Psi}^{(i)}$ and $\Psi^{(i)}$ are computed by Algorithm 5, and if we introduce the corresponding scalar fluxes $\widetilde{\phi}^{(i)} := \mathcal{P}\widetilde{\Psi}^{(i)}$ and $\phi^{(i)} := \mathcal{P}\Psi^{(i)}$, then*

$$\|(\mathcal{I} - (\sigma_s + \alpha^{(i)}\nu\sigma_f)\mathcal{K}_\sigma)\widetilde{\phi}^{(i+1)} - \nu\sigma_f\mathcal{K}_\sigma\phi^{(i)}\|_{L^2(V)} \;\leq\; \tau^{(i)} \ , \quad \textit{and} \tag{3.13}$$

$$\phi^{(i+1)} \;=\; \frac{\widetilde{\phi}^{(i+1)}}{\|\widetilde{\phi}^{(i+1)}\|_{L^2(V)}} \ . \tag{3.14}$$

Thus, when $\Psi^{(i)}$ is close to an eigenfunction corresponding to the minimal eigenvalue of (1.4), then $\phi^{(i)}$ is predominantly in the direction e_1 and $t(\phi^{(i)})$ will be close to zero. The following theorem gives a mechanism for bounding $t(\phi^{(i+1)})$. This theorem will be used in Corollaries 3.5 and 3.6 to obtain convergence properties of several variants of Algorithm 5. For convenience we will discuss an abstract version of (3.13), (3.14) where the superscripts are suppressed.

Theorem 3.3. *Suppose* $s(\phi) \neq 0$,

$$\|(\mathcal{I} - (\sigma_s + \alpha\,\nu\sigma_f)\mathcal{K}_\sigma)\widetilde{\phi} \;-\; \nu\sigma_f\mathcal{K}_\sigma\phi\|_{L^2(V)} \;\leq\; \tau \ , \tag{3.15}$$

$$\textit{and set} \quad \phi' \;=\; \frac{\widetilde{\phi}}{\|\widetilde{\phi}\|_{L^2(V)}} \ . \tag{3.16}$$

Then, if $\tau \;<\; \nu\sigma_f\omega_1 c(\phi)$, *we have with constant* $C_1 \;=\; 1/(\nu\sigma_f\omega_2)$,

$$t(\phi') \;\leq\; \left(\frac{s(\phi) + C_1\,\tau}{c(\phi) - C_1\,\tau}\right)\left|\frac{\lambda_1 - \alpha}{\lambda_2 - \alpha}\right| \ . \tag{3.17}$$

Proof. To make the notation simpler, we set, without loss of generality, $\nu = 1$ in the proof. First observe that if $\widetilde{\phi} = 0$ in (3.15), then, since $s(\phi) \neq 0$, we have

$$\tau \;\geq\; \sigma_f\|\mathcal{K}_\sigma\phi\|_{L^2(V)} \;=\; \sigma_f\left\{\sum_{j=1}^{\infty}\omega_j^2|\xi_j(\phi)|^2\right\}^{1/2} \;>\; \sigma_f\omega_1 c(\phi) \ ,$$

which contradicts the assumption. So $\widetilde{\phi} \neq 0$ and the normalisation (3.16) is well-defined.

To obtain the bound on $t(\phi')$, set

$$R \;:=\; (\mathcal{I} - (\sigma_s + \alpha\sigma_f)\mathcal{K}_\sigma)\widetilde{\phi} \;-\; \sigma_f\mathcal{K}_\sigma\phi \ . \tag{3.18}$$

Because the (ω_j, e_j) are eigenpairs of $\mathcal{K}_\sigma$, we have (using (3.10), (3.16) and (2.24)), for all $j \geq 1$,

$$\begin{aligned}
\xi_j(R) \;&=\; \big((\mathcal{I} - (\sigma_s + \alpha\sigma_f)\mathcal{K}_\sigma)\widetilde{\phi} \;-\; \sigma_f\mathcal{K}_\sigma\phi, e_j\big)_{L^2(V)} \\
&=\; (1 - (\sigma_s + \alpha\sigma_f)\omega_j)\xi_j(\widetilde{\phi}) - \sigma_f\omega_j\xi_j(\phi) \\
&=\; (1 - \sigma_s\omega_j - \alpha\sigma_f\omega_j)\|\widetilde{\phi}\|_{L^2(V)}\xi_j(\phi') - \sigma_f\omega_j\xi_j(\phi) \\
&=\; \sigma_f\omega_j\left[\|\widetilde{\phi}\|_{L^2(V)}(\lambda_j - \alpha)\xi_j(\phi') - \xi_j(\phi)\right] \ .
\end{aligned} \tag{3.19}$$

Now using (3.11) and (3.15), we have

$$\tau \;\geq\; \|R\|_{L^2(V)} \;\geq\; |\xi_1(R)| \;\geq\; \sigma_f\omega_1\left[c(\phi) - \|\widetilde{\phi}\|_{L^2(V)}\,|\lambda_1 - \alpha|\,c(\phi')\right] \ ,$$

and a rearrangement of this yields,

$$\frac{1}{c(\phi')} \;\leq\; \left(\frac{\sigma_f\omega_1}{\sigma_f\omega_1 c(\phi) - \tau}\right)\,|\lambda_1 - \alpha|\,\|\widetilde{\phi}\|_{L^2(V)} \ . \tag{3.20}$$

On the other hand, rearranging (3.19) gives for $j \geq 2$

$$\xi_j(\phi')\|\widetilde{\phi}\|_{L^2(V)} \;=\; \left(\frac{1}{\lambda_j - \alpha}\right)\left[\frac{\xi_j(R)}{\sigma_f\omega_j} + \xi_j(\phi)\right] \, . \tag{3.21}$$

Now recall (2.24) which leads with $\nu = 1$ to

$$\sigma_f\omega_j(\lambda_j - \alpha) \;=\; 1 - (\sigma_s + \sigma_f\alpha)\omega_j \, .$$

As ω_j tends to zero, this shows that $\sigma_f\omega_j(\lambda_j - \alpha)$ increases with j (as does $\lambda_j - \alpha$). We can now square (3.21), sum over $j = 2, \ldots, \infty$, take the square root and use the triangle inequality for the Euclidean norm to get

$$\begin{aligned} s(\phi')\|\widetilde{\phi}\|_{L^2(V)} \;&=\; \left(\sum_{j=2}^{\infty}\left|\frac{\xi_j(R)}{\sigma_f\omega_j(\lambda_j - \alpha)} + \frac{\xi_j(\phi)}{\lambda_j - \alpha}\right|^2\right)^{1/2} \\ &\leq\; \frac{1}{\sigma_f\omega_2|\lambda_2 - \alpha|}\left(\sum_{j=2}^{\infty}|\xi_j(R)|^2\right)^{1/2} \;+\; \frac{1}{|\lambda_2 - \alpha|}\left(\sum_{j=2}^{\infty}|\xi_j|^2\right)^{1/2} \, , \end{aligned}$$

where we used the decrease of $|\lambda_j - \alpha|^{-1}$ and $|\sigma_f\omega_j(\lambda_j - \alpha)|^{-1}$. Applying (3.15), recalling the definition of R in (3.18) and of $s(\phi)$, we then obtain

$$s(\phi')\|\widetilde{\phi}\|_{L^2(V)} \;\leq\; \frac{1}{|\lambda_2 - \alpha|}\left(\frac{\tau}{\sigma_f\omega_2} + s(\phi)\right) \, . \tag{3.22}$$

Finally, by rearranging the product of (3.20) and (3.22), and using the definition of C_1, we obtain the result. □

The estimate (3.17) contains a great deal of information about the convergence of Algorithm 5. For example, if $\alpha^{(i)}$ converges quadratically to λ_1, then, with a fixed choice of $\tau^{(i)} = \tau_0$ (satisfying the assumption of Theorem 3.3), the algorithm will converge quadratically. We define a possible candidate for $\alpha^{(i)}$ in the following lemma.

Lemma 3.4. *Given $\Psi^{(i)} \in L^2(V, L^1(\mathbb{S}^2))$, consider the non-standard Rayleigh quotient*

$$\widetilde{\rho}^{(i)} \;:=\; \frac{(\mathcal{P}\Psi^{(i)}, \mathcal{P}\mathcal{T}^{-1}(\mathcal{T} - \mathcal{S})\Psi^{(i)})_{L^2(V)}}{(\mathcal{P}\Psi^{(i)}, \mathcal{P}\mathcal{T}^{-1}\mathcal{F}\Psi^{(i)})_{L^2(V)}} \;=\; \frac{(\phi^{(i)}, (\mathcal{I} - \sigma_s\mathcal{K}_\sigma)\phi^{(i)})_{L^2(V)}}{(\phi^{(i)}, \nu\sigma_f\mathcal{K}_\sigma\phi^{(i)})_{L^2(V)}} \, . \tag{3.23}$$

This satisfies the estimate

$$\widetilde{\rho}^{(i)} \;=\; \lambda_1 + \mathcal{O}(s(\phi^{(i)})^2) \, . \tag{3.24}$$

Proof. We write

$$\phi^{(i)} \;=\; c(\phi^{(i)})e_1 \;+\; s(\phi^{(i)})u^{(i)} \;, \tag{3.25}$$

where $\|u^{(i)}\|_{L^2(V)} = 1$ and $(u^{(i)}, e_1)_{L^2(V)} = 0$. Now using (2.24), we see that

$$\widetilde{\rho}^{(i)} \;=\; \frac{(1-\sigma_s\omega_1)\, c(\phi^{(i)})^2 \;+\; \mathcal{O}(s(\phi^{(i)})^2)}{\nu\sigma_f\omega_1\, c(\phi^{(i)})^2 \;+\; \mathcal{O}(s(\phi^{(i)})^2)} \;=\; \lambda_1 \;+\; \mathcal{O}(s(\phi^{(i)})^2) \;,$$

which proves the estimate (3.24). □

Using Theorem 3.3 and Lemma 3.4, we obtain the following convergence result for Algorithm 5.

Corollary 3.5. *Suppose that for every step in Algorithm 5 the conditions of Theorem 3.3 are satisfied and the shift* $\alpha^{(i)} = \widetilde{\rho}^{(i)}$ *(Rayleigh Quotient Iteration) is applied. Then*

$$t(\phi^{(i+1)}) \;\leq\; \left(\frac{s(\phi^{(i)}) + C_1\,\tau^{(i)}}{c(\phi^{(i)}) - C_1\,\tau^{(i)}}\right) \left|\frac{C_2}{\lambda_2 - \lambda_1}\right| \; t(\phi^{(i)})^2 \;, \quad C_2 \text{ constant} \;.$$

Hence, Algorithm 5 converges quadratically. The convergence rate is even cubic if the tolerances decrease with rate

$$\tau^{(i)} \;\leq\; C_3 s(\phi^{(i)}) \;, \quad C_3 \text{ constant} \;, \tag{3.26}$$

or if the linear systems are solved exactly.

On the other hand, using (3.17) for a fixed shift and decreasing tolerances, we get the following corollary.

Corollary 3.6. *If in every iteration of Algorithm 5 the conditions of Theorem 3.3 are met and fixed shifts* $\alpha^{(i)} = \alpha_0$ *, as well as tolerances satisfying* (3.26) *are used, then for small enough* C_3 *in* (3.26)

$$t(\phi^{(i+1)}) \;\leq\; \left(\frac{1 + C_1C_3}{1 - C_1C_3t(\phi^{(i)})}\right) \left|\frac{\lambda_1 - \alpha_0}{\lambda_2 - \alpha_0}\right| \; t(\phi^{(i)}) \;.$$

Hence, provided the shift α_0 *is close enough to* λ_1*, we obtain linear convergence of the algorithm.*

Note that this analysis gives no guarantee that Algorithm 5 converges when we use a fixed shift and constant tolerances. We investigate this question numerically in Section 3.4. Furthermore, to satisfy the condition (3.26), we can use the following lemma.

Lemma 3.7. *If we choose the tolerances such that*

$$\tau^{(i)} \ \leq \ \|\widetilde{\mathrm{res}}^{(i)}\|_* \ ,$$

where $\widetilde{\mathrm{res}}^{(i)} := (\mathcal{T} - \mathcal{S} - \widetilde{\rho}^{(i)}\mathcal{F})\Psi^{(i)}$ *, then* $\tau^{(i)}$ *satisfies the estimate* (3.26)*.*

Proof. Using the definitions of the residual $\widetilde{\mathrm{res}}^{(i)}$ and the $*$-norm we have

$$\begin{aligned}
\|\widetilde{\mathrm{res}}^{(i)}\|_* &= \|\mathcal{P}\mathcal{T}^{-1}(\mathcal{T} - \mathcal{S} - \widetilde{\rho}^{(i)}\mathcal{F})\Psi^{(i)}\|_{L^2(V)} \\
&= \|(\mathcal{I} - \sigma_s\mathcal{K}_\sigma - \widetilde{\rho}^{(i)}\nu\sigma_f\mathcal{K}_\sigma)\phi^{(i)}\|_{L^2(V)} \\
&\leq \|(\mathcal{I} - \sigma_s\mathcal{K}_\sigma - \lambda_1\nu\sigma_f\mathcal{K}_\sigma)\phi^{(i)}\|_{L^2(V)} + \|(\lambda_1 - \widetilde{\rho}^{(i)})\nu\sigma_f\mathcal{K}_\sigma\phi^{(i)}\|_{L^2(V)} \, . \quad (3.27)
\end{aligned}$$

For the first term in (3.27) we obtain by applying the orthogonal splitting (3.25) to $\phi^{(i)}$ and using that $(\mathcal{I} - \sigma_s\mathcal{K}_\sigma)e_1 = \lambda_1\nu\sigma_f\mathcal{K}_\sigma e_1$

$$\|(\mathcal{I} - \sigma_s\mathcal{K}_\sigma - \lambda_1\nu\sigma_f\mathcal{K}_\sigma)(c(\phi^{(i)})e_1 \ + \ s(\phi^{(i)})u^{(i)})\|_{L^2(V)} \ = \ C_4 s(\phi^{(i)}) \ ,$$

where $C_4 \ = \ \|(\mathcal{I} - \sigma_s\mathcal{K}_\sigma \ - \ \lambda_1\nu\sigma_f\mathcal{K}_\sigma)u^{(i)}\|_{L^2(V)}$. The second term in (3.27) can be estimated using Lemma 3.4 which gives

$$|\widetilde{\rho}^{(i)} - \lambda_1|\|\nu\sigma_f\mathcal{K}_\sigma\phi^{(i)}\|_{L^2(V)} \ = \ \mathcal{O}(s(\phi^{(i)})^2) \ .$$

Inserting these two results into (3.27), we obtain, for an appropriately chosen C_3, equation (3.26). □

The above analysis will also extend to other iterative methods such as Jacobi-Davidson type methods using techniques applied, for example, in [36, 38, 40].

Remark 3.8. *Note that the convergence analysis in this section is given only for the* continuous *problem* (1.4)*. However, it provides a guide to how iterations behave in* discrete *cases as we will see in Section 3.4, where we investigate two 1D model problems of different complexity. We emphasise here that the obtained numerical solutions contain a* discretisation error *(which was briefly discussed in Section 2.4.3). We do not investigate this difference between the solution of the discrete problem and the continuous solution further in this thesis.*

3.3 Model problems for numerical computations

The model problems we consider here are criticality computations in one-dimensional slab geometry. An example of such a geometry is given in Figure 3.3.

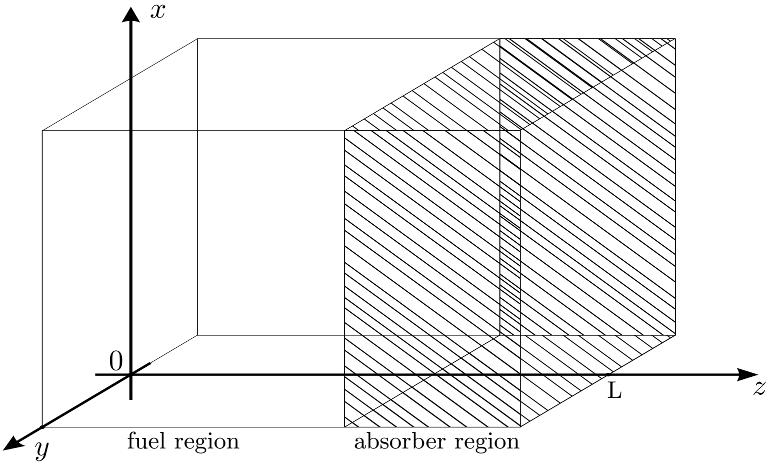


Figure 3.3: *Slab reactor (infinite and homogeneous in x- and y-direction) with fuel and absorber region.*

The neutron flux Ψ is then a function of z, E and $\mu = \cos\theta$ only, where θ denotes the angle between the z-axis and the flux direction $\mathbf{\Omega}$ (see Figure 2.1).

For this geometry the eigenvalue problem (1.5) reduces to

$$\begin{aligned} & \mu \frac{\partial \Psi(z,E,\mu)}{\partial z} + \sigma(z,E)\Psi(z,E,\mu) \\ & - \frac{1}{2}\int_0^\infty \int_{-1}^1 \sigma_s(z,E',E,\mu',\mu)\Psi(z,E',\mu')\,\mathrm{d}\mu'\,\mathrm{d}E' \\ = & \ \lambda\,\frac{\chi(E)}{2}\int_0^\infty \nu(z,E')\sigma_f(z,E')\int_{-1}^1 \Psi(z,E',\mu')\,\mathrm{d}\mu'\,\mathrm{d}E' \ . \end{aligned} \tag{3.28}$$

We now describe two slab model problems of different complexity.

3.3.1 Model problem from the Los Alamos benchmark test set

This model problem is taken from a collection of benchmark tests [109, 110] produced at Los Alamos National Laboratory. The test set contains several benchmark problems with analytic eigenvalue solutions to the neutron transport equation in 1D geometries which are taken from peer-reviewed journal articles.

The problem that we consider here (number 2 of the benchmark test set) is a slab reactor with only one homogeneous region. The energy dependence is removed by assuming that all neutrons are of the same speed (monoenergetic case, see (1.16)) and isotropic scattering is considered. In this case all cross-sections reduce to constants and equation (3.28) simplifies to

$$\mu \frac{\partial \Psi(z,\mu)}{\partial z} \; + \; \sigma \Psi(z,\mu) \; = \; \frac{\sigma_s}{2} \int_{-1}^{1} \Psi(z,\mu')\, \mathrm{d}\mu' \; + \; \lambda\, \nu \frac{\sigma_f}{2} \int_{-1}^{1} \Psi(z,\mu')\, \mathrm{d}\mu' \; .$$

The test problem uses vacuum boundary conditions and is therefore an example of the 1D problem (2.4), (2.5). Further specifications of the problem are given in Table 3.4.

σ	σ_s	σ_f	ν
0.32640	0.225216	0.081600	3.24
slab length: L = 3.707444cm			

Table 3.4: *Data for the problem from the Los Alamos benchmark test set (problem number 2).*

3.3.2 Control rod insertion model problem

This model problem simulates the insertion of a control rod into the core of a nuclear reactor. Usually the cells of a reactor are arranged in a lattice structure or in rings surrounding a central pin. In order to simulate the existence of neighbouring identical cells in 1D, we consider half of a cell and enforce reflective boundary conditions.

The reflective boundary conditions demand in the 1D case that the flux for the incoming angle represented by μ is the same as the outgoing flux for the angle represented by $-\mu$. Hence, with $V = [0, L]$ and E denoting the energy, the boundary conditions for this problem are

$$\begin{aligned} \Psi(0,E,\mu) \; &= \; \Psi(0,E,-\mu) \; , \quad \forall\; \mu > 0 \; , \\ \Psi(L,E,\mu) \; &= \; \Psi(L,E,-\mu) \; , \quad \forall\; \mu < 0 \; . \end{aligned} \tag{3.29}$$

We model the problem as a slab reactor with two regions (see Figure 3.5). The only variation in material properties is in the z-direction. The first part of the slab contains the fuel region, a homogenised mix of fuel (typically uranium or plutonium), water and cladding. The second part, the absorber region, models the control rod and consists of a homogenised mix of the absorber material in the control rod and the remaining water if the control rod is not fully inserted.

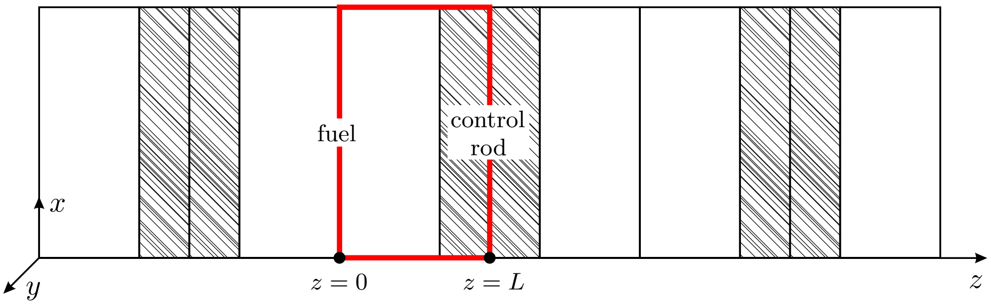


Figure 3.5: *Criticality computations on large lattice structures can be done approximately by treating them as infinite lattices and modelling a part of them (highlighted) using reflective boundary conditions at the sides.*

Depending on the insertion depth of the rod the ratio of absorber to water varies and therefore the material properties in the absorber region change. These are determined by the cross-sections σ, σ_s and σ_f as well as the neutron yield ν and the fission neutron distribution χ. Within each region we assume the material cross-sections to be constant.

The energy spectrum of the model problem is constrained to two groups, i.e. neutrons of high and low energy, corresponding to the angular fluxes Ψ_h and Ψ_l, so that (3.28) can be written as a system of two equations. We assume isotropic scatter and use the corresponding group cross-sections (for example $\sigma_{s,h\to l}(z)$ denotes the probability of a scatter from the high to the low energy group). We obtain two equations which we can write in the vector format

$$\begin{aligned} &\mu \frac{\partial}{\partial z} \begin{pmatrix} \Psi_h(z,\mu) \\ \Psi_l(z,\mu) \end{pmatrix} + \begin{pmatrix} \sigma_h(z) & 0 \\ 0 & \sigma_l(z) \end{pmatrix} \begin{pmatrix} \Psi_h(z,\mu) \\ \Psi_l(z,\mu) \end{pmatrix} \\ &- \begin{pmatrix} \sigma_{s,h\to h}(z) & \sigma_{s,l\to h}(z) \\ \sigma_{s,h\to l}(z) & \sigma_{s,l\to l}(z) \end{pmatrix} \frac{1}{2} \int_{-1}^{1} \begin{pmatrix} \Psi_h(z,\mu') \\ \Psi_l(z,\mu') \end{pmatrix} \mathrm{d}\mu' \\ = \ &\lambda \begin{pmatrix} \chi_h \nu_h(z) \sigma_{f,h}(z) & \chi_h \nu_l(z) \sigma_{f,l}(z) \\ \chi_l \nu_h(z) \sigma_{f,h}(z) & \chi_l \nu_l(z) \sigma_{f,l}(z) \end{pmatrix} \frac{1}{2} \int_{-1}^{1} \begin{pmatrix} \Psi_h(z,\mu') \\ \Psi_l(z,\mu') \end{pmatrix} \mathrm{d}\mu' \ . \end{aligned} \tag{3.30}$$

3.3.3 Discrete ordinates discretisation for the control rod problem

We now describe details of the application of the discrete ordinates approach from Section 2.4.1 to the control rod problem from Section 3.3.2, where the first discretisation with respect to the energy has already been included to arrive at the two-group equation (3.30). We now consider the further steps to arrive at a fully discrete matrix eigenvalue problem.

First, we apply a Gauss-Legendre quadrature with $2N$ points $\{\mu_k\} \subset [-1,1]\backslash\{0\}$, and weights $\{w_k\}$ with $k = -N, \ldots, -1$ and $k = 1, \ldots, N$ to approximate

$$\int_{-1}^{1} f(\mu)\mathrm{d}\mu \;\approx\; \sum_{|k|=1}^{N} w_k f(\mu_k) \;.$$

Applying the angular discretisation and evaluating the resulting semidiscrete problem at the different quadrature points μ_k, $|k| = 1, \ldots, N$, gives the following system of $4N$ ordinary differential equations:

$$\begin{aligned}
&\mu_k \frac{\mathrm{d}}{\mathrm{d}z} \begin{pmatrix} \Psi_h(z,\mu_k) \\ \Psi_l(z,\mu_k) \end{pmatrix} \;+\; \begin{pmatrix} \sigma_h(z) & 0 \\ 0 & \sigma_l(z) \end{pmatrix} \begin{pmatrix} \Psi_h(z,\mu_k) \\ \Psi_l(z,\mu_k) \end{pmatrix} \\
&- \;\begin{pmatrix} \sigma_{s,h\to h}(z) & \sigma_{s,l\to h}(z) \\ \sigma_{s,h\to l}(z) & \sigma_{s,l\to l}(z) \end{pmatrix} \sum_{|i|=1}^{N} w_i \begin{pmatrix} \Psi_h(z,\mu_i) \\ \Psi_l(z,\mu_i) \end{pmatrix} \\
= \;&\lambda \;\begin{pmatrix} \chi_h \nu_h(z)\sigma_{f,h}(z) & \chi_h \nu_l(z)\sigma_{f,l}(z) \\ \chi_l \nu_h(z)\sigma_{f,h}(z) & \chi_l \nu_l(z)\sigma_{f,l}(z) \end{pmatrix} \sum_{|i|=1}^{N} w_i \begin{pmatrix} \Psi_h(z,\mu_i) \\ \Psi_l(z,\mu_i) \end{pmatrix} \;.
\end{aligned} \tag{3.31}$$

We now choose spatial mesh points z_j, $j = 0, \ldots, M$, such that the material boundaries (i.e. the jumps in the piecewise constant cross-sections) coincide with mesh points, and introduce the shorthand notation

$$\Psi_h^{j,k} \;=\; \Psi_h(z_j,\mu_k) \quad \text{and} \quad \Psi_l^{j,k} \;=\; \Psi_l(z_j,\mu_k) \;.$$

A fully discrete approximation of (3.30) can then be constructed by approximating the spatial derivative in (3.31) using a Crank-Nicolson scheme on the intervals $I_j := [z_{j-1}, z_j]$, $j = 1, \ldots, M$, with lengths $h_j := z_j - z_{j-1}$ (see also page 44). We approximate the material properties by their value at the interval midpoints $\bar{z}_j := \frac{z_{j-1}+z_j}{2}$ and obtain

$$\begin{aligned}
&\left[\frac{h_j}{2} \begin{pmatrix} \sigma_h(\bar{z}_j) & 0 \\ 0 & \sigma_l(\bar{z}_j) \end{pmatrix} - \mu_k I_2 \right] \begin{pmatrix} \Psi_h^{j-1,k} \\ \Psi_l^{j-1,k} \end{pmatrix} \\
&+ \left[\frac{h_j}{2} \begin{pmatrix} \sigma_h(\bar{z}_j) & 0 \\ 0 & \sigma_l(\bar{z}_j) \end{pmatrix} + \mu_k I_2 \right] \begin{pmatrix} \Psi_h^{j,k} \\ \Psi_l^{j,k} \end{pmatrix} \\
&- \frac{h_j}{2} \begin{pmatrix} \sigma_{s,h\to h}(\bar{z}_j) & \sigma_{s,l\to h}(\bar{z}_j) \\ \sigma_{s,h\to l}(\bar{z}_j) & \sigma_{s,l\to l}(\bar{z}_j) \end{pmatrix} \sum_{|i|=1}^{N} w_i \left[\begin{pmatrix} \Psi_h^{j,i} \\ \Psi_l^{j,i} \end{pmatrix} + \begin{pmatrix} \Psi_h^{j-1,i} \\ \Psi_l^{j-1,i} \end{pmatrix} \right] \\
= \;&\lambda \frac{h_j}{2} \begin{pmatrix} \chi_h \nu_h(\bar{z}_j)\sigma_{f,h}(\bar{z}_j) & \chi_h \nu_l(\bar{z}_j)\sigma_{f,l}(\bar{z}_j) \\ \chi_l \nu_h(\bar{z}_j)\sigma_{f,h}(\bar{z}_j) & \chi_l \nu_l(\bar{z}_j)\sigma_{f,l}(\bar{z}_j) \end{pmatrix} \sum_{|i|=1}^{N} w_i \left[\begin{pmatrix} \Psi_h^{j,i} \\ \Psi_l^{j,i} \end{pmatrix} + \begin{pmatrix} \Psi_h^{j-1,i} \\ \Psi_l^{j-1,i} \end{pmatrix} \right] \;.
\end{aligned} \tag{3.32}$$

This holds for $j = 1, \ldots, M$ and $|k| = 1, \ldots, N$, and therefore we have $4MN$ equations which contain $4(M+1)N$ unknowns (2 energy groups, $2N$ angles and $M+1$ spatial points). They can be written as a large matrix-vector system which is currently underdetermined.

However, the boundary conditions (3.29) yield $4N$ additional conditions ($2N$ for each of the two energy groups) which allow us to remove the same number of unknowns from the system since

$$\begin{aligned}
(\Psi_h^{0,N}, \ldots, \Psi_h^{0,1})^T &= (\Psi_h^{0,-N}, \ldots, \Psi_h^{0,-1})^T \ , \\
(\Psi_h^{M,-1}, \ldots, \Psi_h^{M,-N})^T &= (\Psi_h^{M,1}, \ldots, \Psi_h^{M,N})^T \ , \\
(\Psi_l^{0,N}, \ldots, \Psi_l^{0,1})^T &= (\Psi_l^{0,-N}, \ldots, \Psi_l^{0,-1})^T \ , \quad \text{and} \\
(\Psi_l^{M,-1}, \ldots, \Psi_l^{M,-N})^T &= (\Psi_l^{M,1}, \ldots, \Psi_l^{M,N})^T \ .
\end{aligned}$$

Using the $4MN$ equations and $4MN$ unknowns, we can set up the corresponding matrices. We represent the first two rows (the transport part) of (3.32) by a matrix T, the scattering part by S and the right-hand side (the fission source) by a matrix F. The vector of unknowns is denoted by $\mathbf{\Psi}$, where we define

$$\mathbf{\Psi} \ := \ (\mathbf{\Psi}_h^0, \ldots, \mathbf{\Psi}_h^j, \ldots, \mathbf{\Psi}_h^M, \mathbf{\Psi}_l^0, \ldots, \mathbf{\Psi}_l^j, \ldots, \mathbf{\Psi}_l^M)^T$$

and

$$\begin{aligned}
\mathbf{\Psi}_h^0 &:= (\Psi_h^{0,-1}, \ldots, \Psi_h^{0,-N})^T \ , \\
\mathbf{\Psi}_h^j &:= (\Psi_h^{j,N}, \ldots, \Psi_h^{j,1}, \Psi_h^{j,-1}, \ldots, \Psi_h^{j,-N})^T, \quad j = 2, \ldots, M-1 \ , \\
\mathbf{\Psi}_h^M &:= (\Psi_h^{M,N}, \ldots, \Psi_h^{M,1})^T \ , \\
\mathbf{\Psi}_l^0 &:= (\Psi_l^{0,-1}, \ldots, \Psi_l^{0,-N})^T \ , \\
\mathbf{\Psi}_l^j &:= (\Psi_l^{j,N}, \ldots, \Psi_l^{j,1}, \Psi_l^{j,-1}, \ldots, \Psi_l^{j,-N})^T, \quad j = 2, \ldots, M-1 \ , \quad \text{and} \\
\mathbf{\Psi}_l^M &:= (\Psi_l^{M,N}, \ldots, \Psi_l^{M,1})^T \ .
\end{aligned}$$

The shortened vectors for the boundary points ($j = 0$ and $j = M$) result from the use of the boundary conditions to reduce the number of unknowns. The ordering of the entries in $\mathbf{\Psi}$ is visualised in Figure 3.6

We obtain the generalised matrix eigenvalue problem

$$(T - S)\mathbf{\Psi} \ = \ \lambda F \mathbf{\Psi} \ ,$$

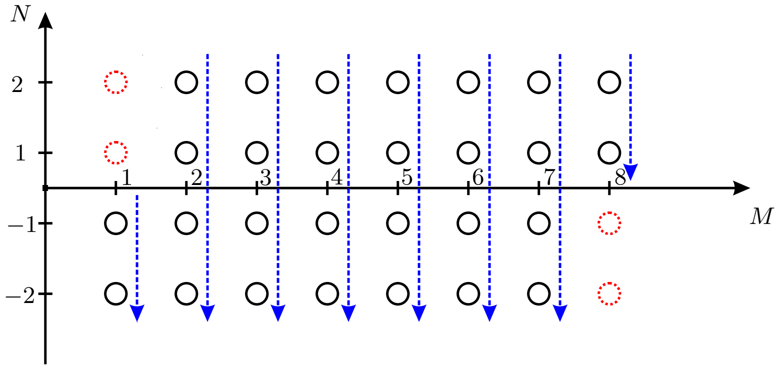


Figure 3.6: *Ordering of the flux elements in the vector* $\mathbf{\Psi}$ *with respect to the spatial and angular mesh points.*

where the matrix T has the structure

$$T = \begin{pmatrix} T_h & 0 \\ 0 & T_l \end{pmatrix} ,$$

with T_h and T_l being sparse matrices with only one upper and one lower diagonal.

The matrices S and F have a block format with equal rows in each block. In practice, instead of setting them up explicitly, we exploit their simple structure to compute efficient matrix vector products.

3.4 Numerical results for criticality computations

We now consider numerical experiments for the two model problems from Section 3.3 using inexact inverse iteration as described in Algorithm 5.

3.4.1 Los Alamos benchmark test set problem

Our first numerical tests are performed using the Los Alamos problem from Section 3.3.1. To obtain a discrete matrix eigenvalue problem we apply a Gauss-Legendre quadrature with $2N$ quadrature points on $[-1, 1]$. The spatial discretisation uses the Euler scheme described in Section 2.4.2 with $M + 1$ mesh points, which preserves the symmetry under reduction to the scalar flux. We also apply a Crank-Nicolson scheme for the spatial approximation and will see that this does not preserve the underlying symmetry. Note that none of our matrices T, S, and F itself is symmetric. Furthermore, we will see that the symmetry preserving Euler scheme only achieves a faster convergence rate if we choose the non-standard Rayleigh quotient $\widetilde{\rho}$ as shift.

The continuous problem has a dominant eigenvalue of $\lambda_1 = 1$ (see [109, 110]). We note again that we ignore the discretisation error in the numerical results and focus on the convergence rate of the iterative scheme by considering the discrete eigenvalue residual. We use $M = 128$ equally sized spatial intervals and $2N = 128$ angular Gauss points leading to non-symmetric generalised matrix eigenvalue problems of dimension 16384×16384. The eigenvalues nearest zero of the discrete problems are $\lambda_1^{\text{Eul}} \approx 0.99570$, $\lambda_2^{\text{Eul}} \approx 2.60907$ and $\lambda_1^{\text{CN}} \approx 1.00003$, $\lambda_2^{\text{CN}} \approx 2.60530$. Our stopping criterion for the outer iteration is $\|\mathbf{res}^{(i)}\|_2 < 10^{-14}$, where $\mathbf{res}^{(i)} := (T - S - \rho^{(i)}F)\mathbf{\Psi}^{(i)}$ with T, S, F, and $\mathbf{\Psi}$ being the discrete versions of $\mathcal{T}$, $\mathcal{S}$, $\mathcal{F}$, and Ψ, respectively. The eigenvalue approximation is given by the standard Rayleigh quotient

$$\rho(\mathbf{\Psi}^{(i)}) := \frac{\langle \mathbf{\Psi}^{(i)}, (T-S)\mathbf{\Psi}^{(i)} \rangle}{\langle \mathbf{\Psi}^{(i)}, F\mathbf{\Psi}^{(i)} \rangle}, \tag{3.33}$$

where $\langle \cdot , \cdot \rangle$ represents the ℓ_2 inner product over all spatial and angular discrete variables. Note that we compute the eigenvalue residual $\mathbf{res}^{(i)}$ in the full, spatially and angular dependent space.

Problem (†) in Algorithm 5 is solved using the GMRES function in MATLAB 2009b with an LU factorisation of T as preconditioner. When the preconditioner is not applied, the GMRES solver fails to converge. Note that the stopping criterion of MATLAB's GMRES function checks the *relative* residual norm (and not the absolute norm as we assumed in our analysis).

As starting guess $\mathbf{\Psi}^{(0)}$ for Algorithm 5, we use a normalised vector with equal positive entries. To measure the convergence rate of the algorithm, we consider the eigenvalue error $\Delta^{(i)} := |\lambda_1 - \rho^{(i)}|$, where λ_1 is the computed eigenvalue when the iteration terminates.

Table 3.7 shows the numerical results for fixed shifts $\alpha_0 = 0.9$ and $\alpha_0 = 0.99$. We used decreasing tolerances $\tau^{(i)} \leq 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$ for the inner solves, where P denotes the discrete version of the projection operator $\mathcal{P}$. The results clearly show linear convergence in both cases with a faster linear rate for $\alpha = 0.99$, agreeing with Corollary 3.6.

When replacing the symmetry preserving Euler scheme from Section 2.4.2 with the Crank-Nicolson discretisation (which does not preserve the symmetry of the reduction), we obtain very similar results to those in Table 3.7. This suggests that the convergence for the fixed shift iteration is not influenced by retaining the underlying symmetry if the tolerances decrease sufficiently fast and the fixed shift is chosen close enough to the desired eigenvalue.

	$\alpha_0 = 0.9$			$\alpha_0 = 0.99$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	2.6E-04	8.0E-03	2.5E-01	1.4E-05	4.2E-04	1.3E-02
2	5.4E-06	2.0E-02	7.7E+01	2.0E-08	1.4E-03	1.0E+02
3	1.3E-07	2.4E-02	4.6E+03	3.0E-11	1.5E-03	7.7E+04
4	3.3E-09	2.6E-02	2.0E+05	4.5E-14	1.5E-03	5.1E+07
5	8.6E-11	2.6E-02	7.7E+06	0.0E+00	0.0E+00	0.0E+00
6	2.2E-12	2.6E-02	3.0E+08			
7	5.8E-14	2.6E-02	1.2E+10			
8	0.0E+00	0.0E+00	0.0E+00			

Table 3.7: *Numerical results for the inexact inverse iteration method in Algorithm 5 with decreasing tolerances* $\tau^{(i)} \leq 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$ *when using the symmetry preserving Euler scheme.*

Surprisingly, even for a *fixed* shift $\alpha_0 = 0.3$ and a *fixed* inner tolerance $\tau_0 = 0.1$, we still obtained linear convergence. This appears to be due to the fact that for sufficiently large i of the outer iteration, the inner solver GMRES is observed to converge after one iteration and the accuracy of the GMRES solves for the linear systems (†) increases. This then results effectively in a slowly decreasing inner tolerance, leading to linear convergence of the method.

Tables 3.8 and 3.9 concern the variable shift case, comparing the convergence for $\alpha^{(i)} = \rho^{(i)}$, the standard Rayleigh quotient in (3.33), and $\alpha^{(i)} = \widetilde{\rho}^{(i)}$, the special Rayleigh quotient from (3.23).

In Table 3.8 we obtain only *linear* convergence for the symmetry preserving Euler scheme and fixed inner tolerances when using the standard Rayleigh quotient $\rho^{(i)}$ as shift, but the numerical results suggest *quadratic* convergence for the special Rayleigh quotient $\widetilde{\rho}^{(i)}$. This agrees with our theory and so we recommend the use of the non-standard Rayleigh quotient $\widetilde{\rho}^{(i)}$ as shift.

	$\alpha^{(i)} = \rho^{(i)}$			$\alpha^{(i)} = \widetilde{\rho}^{(i)}$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	9.7E-05	3.0E-03	9.1E-02	9.4E-05	2.9E-03	8.8E-02
2	2.4E-08	2.4E-04	2.5E+00	2.4E-11	2.5E-07	2.7E-03
3	1.6E-11	6.8E-04	2.9E+04	0.0E+00	0.0E+00	0.0E+00
4	1.1E-15	7.0E-05	4.4E+06			
5	0.0E+00	0.0E+00	0.0E+00			

Table 3.8: *Numerical results for Algorithm 5 with constant tolerances* $\tau_0 = 0.1$ *and two different Rayleigh quotient shifts for matrices arising from the application of the Euler scheme described in Section 2.4.2.*

However, when applying the Crank-Nicolson scheme for the spatial approximation (see

Table 3.9), the underlying symmetry gets lost in the discretisation and neither of the variable shifts achieves quadratic convergence. We used twice as many angular and spatial discretisation points as in Table 3.8 to produce Table 3.9. In this case the convergence rates are clearer to establish from the numerical results. Both shifts give only *linear* convergence, emphasising the benefits of using a symmetry preserving discretisation with the special Rayleigh quotient $\widetilde{\rho}^{(i)}$. However, using the special Rayleigh quotient $\widetilde{\rho}^{(i)}$ in the non-symmetric case is not disadvantageous, but actually leads to slightly faster (but still linear) convergence.

	$\alpha^{(i)} = \rho^{(i)}$			$\alpha^{(i)} = \widetilde{\rho}^{(i)}$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	7.8E-05	2.4E-03	7.3E-02	7.4E-05	2.3E-03	6.9E-02
2	9.7E-09	1.3E-04	1.6E+00	8.3E-10	1.1E-05	1.5E-01
3	3.1E-12	3.1E-04	3.2E+04	1.3E-14	1.6E-05	1.9E+04
4	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00

Table 3.9: *Numerical results for Algorithm 5 with constant tolerances* $\tau_0 = 0.1$ *and Rayleigh quotient shifts using a Crank-Nicolson scheme for the spatial approximation.*

Due to reaching machine precision so quickly, we were not able to clearly establish the predicted cubic convergence for a Rayleigh quotient shift and decreasing tolerances when using the symmetric Euler discretisation and our special Rayleigh quotient shift $\widetilde{\rho}^{(i)}$.

When applying decreasing tolerances to the other three variable shift cases that we considered in Tables 3.8 and 3.9, the numerical results suggest the gain of an additional order in the convergence rate leading to *quadratic* convergence for those problems as Table 3.10 indicates.

	$\alpha^{(i)} = \rho^{(i)}$			$\alpha^{(i)} = \widetilde{\rho}^{(i)}$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	8.0E-05	2.4E-03	7.2E-02	7.2E-05	2.2E-03	6.5E-02
2	1.7E-09	2.1E-05	2.7E-01	1.3E-10	1.8E-06	2.5E-02
3	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00

Table 3.10: *Numerical results as in Table 3.9 but in this case applying decreasing tolerances* $\tau^{(i)} \leq 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$.

The following numerical results are now for the more realistic control rod problem with two different material regions and neutrons of two energy levels.

3.4.2 Control rod insertion model problem

Our second example is the control rod model problem described in Section 3.3.2. We apply a Gauss quadrature and Crank-Nicolson scheme with 128 uniform spatial intervals in the fuel region and 8 equally sized intervals in the absorber part of the problem (resolving the material boundary) and giving a total of $M = 136$ spatial intervals. We use again $2N = 128$ angles, leading to a system of size 34816×34816.

The convergence behaviour of Algorithm 5 is investigated with respect to three different material compositions in the absorber region: (i) The pure absorber case; (ii) a mix of 10% absorber and 90% water; and (iii) a homogeneous case, where the absorber and fuel region have the same cross-sections. The principal eigenvalues in cases (i)–(iii) are $\lambda_1 \approx 1.18$, 0.92, and 0.85, respectively, and the problem details are given in Table 3.11.

	properties of the fuel in (i)-(iii) and absorber in (iii)				
	σ	σ_s		σ_f	ν
h	2.11228E-01	1.90001E-01	1.16636E-05	3.01008E-04	2.48225
l	7.23458E-01	1.85926E-02	7.04384E-01	1.01367E-02	2.43832

	absorber properties for (i)			absorber properties for (ii)		
	σ	σ_s		σ	σ_s	
h	3.96908E-02	1.76684E-02	1.75847E-06	1.78882E-01	1.39293E-01	9.30325E-06
l	1.74551E-01	1.12667E-05	1.60722E-02	1.03217E+00	3.37989E-02	1.00381E+00

problem length: L = 5.25cm (fuel region: 5.0cm, absorber region: 0.25cm)

Table 3.11: *Data for the control rod insertion model problem; the scatter cross-sections are arranged as in* (3.30).

The theory does not apply directly to (3.30) and even the homogeneous problem (iii) does not have an obvious symmetric reduction. Moreover, we assumed vacuum boundary conditions for our analysis above, while this model problem has reflective boundary conditions. However, the numerical results are nevertheless interesting and give an indication for possible extensions of our analysis.

For the first test we used the same starting vector and stopping criterion as in the Los Alamos problem but changed the fixed shift to $\alpha_0 = 0.5$. With this and a constant inner tolerance $\tau_0 = 0.1$, we failed to converge to our demanded accuracy in all of cases (i) to (iii). The first five columns in Table 3.12 show that the norm of the residual and the error in the eigenvalue do not decrease any further between 200 and 2000 iterations. The increasing accuracy of the inner GMRES solves, that we saw for the Los Alamos problem, was not observed here.

	$\tau_0 = 0.1$				$\tau_0 = 10^{-12}$		
	$\Delta^{(200)}$	$\Vert\texttt{res}^{(200)}\Vert_2$	$\Delta^{(2000)}$	$\Vert\texttt{res}^{(2000)}\Vert_2$	i	$\Delta^{(i)}$	$\Vert\texttt{res}^{(i)}\Vert_2$
pure absorber	5.5E-02	2.0E-04	5.5E-02	2.0E-04	6	0.0E+00	3.2E-15
absorber & water mix	9.3E-03	2.4E-04	9.3E-03	2.4E-04	5	0.0E+00	3.9E-16
homogeneous case	3.9E-03	1.5E-04	3.9E-03	1.5E-04	1	0.0E+00	1.4E-15

Table 3.12: *Fixed shift* $\alpha_0 = 0.5$ *; for* $\tau_0 = 10^{-12}$ *the problems converge within i iterations.*

We recovered convergence only by decreasing the fixed tolerance τ_0 to 10^{-12} as the final columns in Table 3.12 show. These small tolerances resulted in almost exact solves of the linear system so that the convergence is not greatly surprising. The statement that the homogeneous problem was solved in only one iteration (last row in Table 3.12) is no typing error but is due to the fact that our starting vector with equal entries is almost an eigenvector in this case (the solution to a homogeneous slab problem with reflective boundary conditions is a constant flux in each of the energy groups).

Therefore, in order not to give problem (iii) an advantage for the remaining numerical tests, we changed our starting vector to one whose entries were chosen randomly in $(0, 1)$, i.e. from the uniform distribution $U(0, 1)$. Repeating the previous test for the homogeneous case with a random starting vector increased the number of iterations needed to converge from one to five.

Table 3.13 gives numerical results for the cases (i) to (iii) using a fixed shift and decreasing tolerances. We obtain – as in the Los Alamos problem – *linear* convergence. Apart from the first iterate, the convergence speed for all three cases appears to be similar. This suggests that the heterogeneity does not impair the convergence in this case.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	1.7E-04	1.9E-04	2.1E-04	1.1E-05	1.2E-05	1.4E-05	4.2E-06	5.0E-06	5.9E-06
2	1.0E-06	6.0E-03	3.6E+01	8.8E-08	8.2E-03	7.7E+02	4.6E-09	1.1E-03	2.6E+02
3	4.7E-09	4.7E-03	4.6E+03	2.6E-10	2.9E-03	3.3E+04	3.7E-11	8.2E-03	1.8E+06
4	2.4E-11	5.2E-03	1.1E+06	1.0E-12	4.0E-03	1.5E+07	1.7E-13	4.4E-03	1.2E+08
5	2.3E-13	9.5E-03	3.9E+08	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
6	0.0E+00	0.0E+00	0.0E+00						

Table 3.13: *Numerical results for the control rod problem using a fixed shift* $\alpha_0 = 0.5$ *and decreasing inner tolerances* $\tau^{(i)} \leq 0.1 \, \Vert PT^{-1} \, \mathbf{res}^{(i)} \Vert_2$ *for different materials in the absorber region.*

Table 3.14 illustrates the convergence properties using constant tolerances $\tau_0 = 0.1$ and variable shifts $\alpha^{(i)}$ chosen to be the non-standard Rayleigh quotients $\widetilde{\rho}^{(i)}$. As in the Los Alamos problem for the Crank-Nicolson discretisation, we obtain *linear* but not quadratic convergence. The numerical results suggest that for the use of a Rayleigh quotient shift, the heterogeneity in the first two problems may influence the speed of the linear convergence.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	1.1E-02	1.3E-02	1.4E-02	1.7E-05	1.9E-05	2.2E-05	1.9E-05	2.3E-05	2.7E-05
2	1.6E-04	1.4E-02	1.2E+00	1.6E-06	9.4E-02	5.5E+03	8.4E-08	4.4E-03	2.3E+02
3	1.4E-05	8.9E-02	5.6E+02	9.4E-09	5.8E-03	3.6E+03	2.4E-10	2.8E-03	3.3E+04
4	4.7E-08	3.3E-03	2.3E+02	3.8E-09	4.0E-01	4.3E+07	7.6E-13	3.3E-03	1.4E+07
5	5.2E-10	1.1E-02	2.4E+05	5.8E-11	1.6E-02	4.1E+06	0.0E+00	0.0E+00	0.0E+00
6	1.0E-11	1.9E-02	3.6E+07	0.0E+00	0.0E+00	0.0E+00			
7	0.0E+00	0.0E+00	0.0E+00						

Table 3.14: *Numerical results for the control rod problem with* $\alpha^{(i)} = \widetilde{\rho}^{(i)}$ *and constant tolerances* $\tau_0 = 0.1$.

Solving the same fixed tolerance problems with the standard Rayleigh quotient $\rho^{(i)}$ gave similar convergence results to those in Table 3.14 without indicating superiority of one Rayleigh quotient over the other.

Finally, using Rayleigh quotient shifts and decreasing tolerances, the convergence rates for the two variable shift cases improve. Table 3.15 contains numerical results for the standard Rayleigh quotient which suggest that *quadratic* convergence is achieved. Applying the non-standard Rayleigh quotient $\widetilde{\rho}^{(i)}$ leads to similar results.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	1.3E-05	1.5E-05	1.6E-05	2.7E-05	3.0E-05	3.4E-05	1.0E-05	1.2E-05	1.4E-05
2	1.3E-11	1.0E-06	7.5E-02	2.2E-11	8.2E-07	3.0E-02	2.4E-12	2.3E-07	2.3E-02
3	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00

Table 3.15: *Numerical results for the control rod problem using the standard Rayleigh quotient shift* $\rho^{(i)}$ *and decreasing tolerances* $\tau^{(i)} \leq 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$.

The final numerical results in this chapter consider as shift for the control rod problem the two-sided Rayleigh quotient

$$\rho_q^{(i)} \;=\; \frac{\mathbf{y}^{(i)^T}(T-S)\mathbf{\Psi}^{(i)}}{\mathbf{y}^{(i)^T}F\mathbf{\Psi}^{(i)}} \;,$$

where $\mathbf{y}^{(i)}$ is an approximation to the left eigenvalue of $\mathbf{y}^T(T-S) = \lambda\,\mathbf{y}^T F$. This method is known as Ostrowski's Two-Sided Iteration and discussed in [90], where it is shown that $\rho_q^{(i)}$ approximates λ_1 quadratically and that for exact solves cubic convergence of the iteration is obtained. For our inexact solves with fixed tolerance $\tau_0 = 0.1$ we expect a quadratic convergence rate which is supported by Table 3.16.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	3.1E-03	3.4E-03	3.8E-03	8.9E-03	1.0E-02	1.1E-02	3.1E-03	3.6E-03	4.3E-03
2	7.7E-06	2.5E-03	8.3E-01	2.3E-06	2.5E-04	2.9E-02	6.9E-06	2.3E-03	7.4E-01
3	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00

Table 3.16: *Numerical results for the control rod problem using the a quadratic Rayleigh quotient shift $\rho_q^{(i)}$ and fixed tolerances $\tau_0 = 0.1$ for different materials in the absorber region.*

In order to compute the Rayleigh quotient shift $\rho_q^{(i)}$ we need to solve in every outer iteration two linear systems (the forward and the adjoint problem) which roughly doubles the computational demand since the solution of the source problem is the dominating work in the iterative process. Often this is not worth the effort.

Nevertheless, this well-known approach has recently regained some interest in conjunction with Monte Carlo methods (see e.g. [74]), where the adjoint solution is obtained from a cheap (but coarse) diffusion solution while a Monte Carlo solver is used to obtain the solution of the forward problem. We will consider Monte Carlo methods for neutron transport problems in the next chapter.

Chapter 4

Monte Carlo methods in neutron transport theory

Advances in nuclear reactor design lead to more and more complex geometries and the need to model an increasing amount of detail. Deterministic methods as described in Section 1.5 are either by nature (for example in the diffusion approximation) or due to the huge computational demands (S_N and P_N approach) not able to provide such detail for complex problems.

Monte Carlo statistical sampling methods on the other hand use no (or very few) approximations to the true physics and are able to capture details in complex three dimensional geometries. Figure 4.1 is a VisualWorkshop[1] image of a model for the Monte Carlo code MCBEND, provided by the ANSWERS Software Service of Serco Technical and Assurance Services. It shows part of a model of a Magnox reactor at the Oldbury nuclear power station in South Gloucestershire, UK. Capturing such a high level of detail in a 3D problem using deterministic methods is currently not possible. Furthermore, due to their inherent ease of parallelisation and the growth in computer power, the statistical sampling methods are becoming more and more popular.

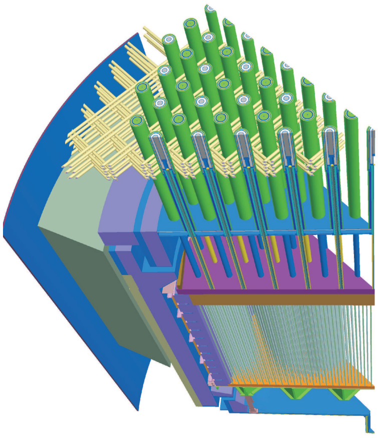


Figure 4.1: *Model of a Magnox reactor which is used for Monte Carlo calculations.*

[1] VisualWorkshop is jointly developed between Serco Technical and Assurance Services and Sellafield Ltd.

Monte Carlo methods give rise to a large research area on their own and discussing all different techniques and their background in detail would go well beyond the scope of this thesis. The objective of this chapter is to indicate how Monte Carlo techniques are applied to problems arising in neutron transport and to point out where the advantages and drawbacks of these solution methods lie. In particular the latter then motivate the use of a new iterative scheme for solving the criticality problem, which we will discuss in Chapter 5.

Furthermore, we link the steps of the statistical sampling technique and the underlying mathematics for the neutron transport equation together, providing a new angle on the Monte Carlo method in this context while only using simple tools and avoiding the probabilistic framework of random walks in references such as [111]. The idea of using a Neumann series to solve source problems containing scatter has been mentioned briefly in [29, §2.2.2] and [31, §3.4]. We will use our theoretical framework from Chapter 2 to make this link rigorous and obtain a convergence result for the source iteration method.

We start this chapter with a brief historical review of Monte Carlo methods and particularly their relation to neutron transport problems which motivated significant advances. Then a short review on elementary probability theory that is needed in the following sections is provided. We next describe how source problems can be solved using Monte Carlo techniques, and finally discuss the solution of the criticality problem using statistical sampling, before finishing this chapter with numerical results to support the theory.

4.1 History of the Monte Carlo method

The name “Monte Carlo” was first used in the article [85] by Nicolas Metropolis and Stanislav Ulam in 1949. However, simple statistical sampling tests were already performed many years before that. Generally the first reference is that of an experiment known as “Buffon’s needle” from 1777 (see [25]), which, as Laplace later showed in [72], could be used to estimate π.

A lot of progress and research on Monte Carlo techniques was done during and after the second world war. Key characters of that time were Metropolis, Ulam and John von Neumann who performed Monte Carlo calculations on the ENIAC machine at the Los Alamos laboratory. According to [32, 88], the idea of obtaining an approximate answer to a (physical) process by means of simulation came to Ulam when playing a simple patience card game and wondering what the chances to win are. He thought that playing the game often enough would give him a good estimate for these. He then discussed this idea with von Neumann and Metropolis and they applied it to neutron transport problems that they had been working on at Los Alamos.

With the rapid growth of computing power Monte Carlo approaches received an increasing interest over time and their application to neutron transport has been studied in several books (e.g. [64, 71, 108, 111]). Today, Monte Carlo methods play a very important role in the neutron transport community. They are used to produce benchmarking results for shielding and criticality problems (e.g. [22, 119]) and many different computer codes have been developed (see [67] for a recent overview of Monte Carlo radiation transport codes).

4.2 Background on probability theory

In order to understand how Monte Carlo methods work and in what sense they converge to the solution of neutron transport problems, we need some background on probability theory. While there exist many references with further details on probability theory (e.g. [50, 68]), we try to keep this section as short and simple as possible and follow the exposition in [80, §7].

Let us start by considering a *real random variable* X. This is a measurable map from a probability space to $\mathbb{R}$ (see, for example, [68]), i.e. X takes an element from the probability space and maps it to a real number. We then denote the *probability* that X attains a value between a and b by $P[a \leq X \leq b]$, and using this, we can define the *probability density function* $f(x)$ as the limit of

$$f(x)\Delta x \;=\; P[x \leq X \leq x + \Delta x] \tag{4.1}$$

when $\Delta x \downarrow 0$. It follows that $f(x) \geq 0$ for all $x \in \mathbb{R}$ and, moreover,

$$\int_{x_-}^{x_+} f(x)\,\mathrm{d}x \;=\; 1 \;,$$

where x_- and x_+ denote the lower and upper values (possibly $-\infty$ and $+\infty$) of the range of X.

Now the corresponding *cumulative distribution function* $F(x)$ denotes the probability that a random variable X attains values less than or equal to x. It is given by

$$F(x) \;:=\; P[X \leq x]$$

and satisfies

$$F(x) \ = \ \int_{x_-}^{x} f(x')\,\mathrm{d}x' \ .$$

This implies that $F(x)$ is an increasing real function with

$$\lim_{x \to x_-} F(x) \ = \ 0 \ , \tag{4.2}$$

$$\lim_{x \to x_+} F(x) \ = \ 1 \ , \quad \text{and} \tag{4.3}$$

$$P[a \leq X \leq b] \ = \ F(b) - F(a) \ .$$

A major challenge when performing Monte Carlo calculations is to obtain realisations of random variables X_j that are distributed according to a given probability density function $f(x)$. To achieve this the cumulative density function $F(x)$, which by (4.2) and (4.3) attains values between zero and one, can be used together with a random number generator that produces numbers ξ_j that are uniformly distributed between zero and one. A *realisation* x_j of the random variable X_j can then be computed for a given ξ_j via

$$x_j \ = \ F^{-1}(\xi_j) \ . \tag{4.4}$$

Note that the x_j in (4.4) is not a random variable but its image, i.e. a real value that describes the outcome of the random variable for a certain sample from the probability space. There exist several different techniques to perform the sampling of realisations of random variables (see e.g. [29, p. 542] for examples in neutron transport).

We denote by

$$E[X] \ := \ \int_{x_-}^{x_+} x f(x)\mathrm{d}x$$

the *expected value* (or *expectation*) of the random variable X, where $f(x)$ is the corresponding probability distribution function.

Let X now be a random variable whose expectation describes a quantity that we would like to estimate (for example the average scalar flux in a certain spatial region or the criticality of a nuclear reactor). Furthermore, let X_j be independent and identically distributed copies of X (i.e. also random variables). An estimator for the expectation of X is given by the *sample mean*

$$\hat{X} \ := \ \frac{1}{n}\sum_{j=1}^{n} X_j \ . \tag{4.5}$$

We use the Monte Carlo method to obtain realisations x_j of the random variables X_j

to compute a realisation $\hat{x}$ of the estimator $\hat{X}$ in (4.5) from n simulated *histories*. These histories are in our case the trajectories of the neutrons from their “birth” to their “death”, and the x_j are the contributions for each of these trajectories to the quantity that we would like to estimate. Loosely speaking, the idea of the Monte Carlo method is now to approximate $E[X]$ by $\hat{x}$. For $n \to \infty$ the estimate $\hat{x}$ converges in a probabilistic way (see (4.9)) to $E[X]$. This can be made more rigorous as we will see in the following.

For independent and identically distributed copies X_j of X, their expectation is

$$E[X_j] \;=\; E[X] \quad \text{for all} \quad j = 1, \ldots, n \;.$$

Then, by the linearity of the integral, we get

$$E[\hat{X}] \;=\; E\left[\frac{1}{n}\sum_{j=1}^{n} X_j\right] \;=\; \frac{1}{n}\sum_{j=1}^{n} E[X_j] \;=\; E[X] \;,$$

which tells us that the sample mean is a so-called unbiased estimator of $E[X]$. Note that this does not say that every realisation x_j of X_j, or even the estimate $\hat{x}$ of the sample mean $\hat{X}$, will be equal to the true expected value $E[X]$. However, the realisations will be correct on average and vary around the true value. The question is now how this spread can be estimated and controlled.

The common way to measure the accuracy of the sample results is to consider the *variance* $\sigma^2[X]$ (sometimes also denoted as Var[X]) of the random variable X, which is defined as

$$\sigma^2[X] \;:=\; E[(X - E[X])^2] \;. \tag{4.6}$$

Note that the σ in (4.6) is not related to the total cross-section from (2.1) that we dealt with so far. However, as both notations are standard in the literature, we do not want to alter them and it should be clear from the context which definition is to be used.

As the expectation is a linear operator,

$$\sigma^2[X] \;=\; E[X^2] - (E[X])^2 \;,$$

and it can be shown (see, for example, [80, p. 316]) that we have for the variance of the sample mean $\hat{X}$

$$\sigma^2[\hat{X}] \;=\; E[(\hat{X} - E[\hat{X}])^2] \;=\; \frac{\sigma^2[X]}{n} \;.$$

From this we obtain

$$\sigma[\hat{X}] \;=\; \frac{\sigma[X]}{\sqrt{n}} \;, \tag{4.7}$$

where $\sigma[X]$, the square root of the variance, is the *standard deviation* of X.

Hence, let us assume that, using (4.5), we construct a realisation $\hat{x}$ of $\hat{X}$ from n realisations x_j of copies X_j of the random variable X. Equation (4.7) tells us now that if we constructed many estimates $\hat{x}$ with this approach, then the spread in these realisations $\hat{x}$ around $E[X]$ is proportional to $\sigma[X]$ and falls of with the square root of n. Therefore, if we increase n, i.e. if we use a larger number of particle histories, then the computed $\hat{x}$ will lie closer around the expectation $E[X]$.

To obtain an estimate of the spread in $\hat{X}$ in practice, we need an estimate for $\sigma[X]$. An unbiased estimate for $\sigma^2[X]$ is provided by the *sample variance*

$$S^2 \;:=\; \frac{1}{n-1}\sum_{j=1}^{n}(X_j - \hat{X})^2 \;,$$

which, as shown in [80, p. 318], satisfies $E[S^2] = \sigma^2[X]$. This is then used to compute an estimate of $\sigma[X]$ by the *sample standard deviation*

$$S \;=\; \left(\frac{n}{n-1}\right)^{\frac{1}{2}} \left(\frac{1}{n}\sum_{j=1}^{n} X_j^2 - \hat{X}^2\right)^{\frac{1}{2}} \;. \tag{4.8}$$

In the next step we apply the central limit theorem (see, for example, [35, 68]). Let us assume that we perform the Monte Carlo computations for a fixed n several times and consider the resulting realisations $\hat{x}$ of the sample mean $\hat{X}$. Then the central limit theorem tells us that if the number of samples n approaches infinity, the corresponding probability function $f_n(\hat{x})$ of $\hat{X}$ tends to the normal distribution with expectation $E[X]$ and standard deviation $\sigma[X]/\sqrt{n}$.

This result allows us to obtain for large enough n a probability that states that the realisations of the sample mean $\hat{X}$ lie within an interval of a certain size around the true expectation $E[X]$. For a given probability the size of the interval depends on the standard deviation and the number of samples. We get, for example, that

$$P\left[E[X] - \frac{\sigma[X]}{\sqrt{n}} \;\le\; \hat{X} \;\le\; E[X] + \frac{\sigma[X]}{\sqrt{n}}\right] \;\approx\; 0.6826 \;, \tag{4.9}$$

stating that the computed estimates for the sample mean $\hat{X}$ will lie with a probability of about 68% in the interval $\left(E[X] - \frac{\sigma[X]}{\sqrt{n}}, E[X] + \frac{\sigma[X]}{\sqrt{n}}\right)$. For further details of how to obtain this result see, for example, [80, p. 319]. If the width of the interval is increased by considering $2\sigma[X]$ or $3\sigma[X]$ instead of $\sigma[X]$, the probability of a realisation of $\hat{X}$ lying in the larger intervals increases to 0.954 and 0.997 respectively. In practice we use the sample standard deviation S, defined in (4.8), to estimate $\sigma[X]$ in (4.9).

The aim is now to obtain a high probability at the same time as a small interval width. The simplest way to achieve this is by increasing n and therefore sampling more particles. However, this comes at the expense of computing time and finding a balance between the accuracy of the sample mean and the computing time needed is an important problem in the application of Monte Carlo methods.

As an alternative to simulating more particles, there exist many different *variance reduction techniques* for neutron transport problems which lead to a smaller $\sigma[X]$ and therefore a better estimate in (4.9) (see, for example, [80, §7-6]). In order to keep this chapter concise we will not discuss these techniques here. Also note that for the central limit theorem to hold true, a large enough n must be taken. In addition, the histories of the Monte Carlo calculations have to be independent which turns out to cause some difficulty when the method is used to solve the criticality problem (see e.g. [17]).

Let us summarise the above results using a description from [80] before we give details of an actual Monte Carlo method. Although the probability density function $f(x)$ is not a Gaussian distribution, the probability density function $f_n(\hat{x})$, that describes the distribution of the mean values $\hat{X}$ computed from n histories, resembles a normal distribution for sufficiently large n. Furthermore, the standard deviation of $f_n(\hat{x})$ decreases proportional to $1/\sqrt{n}$. Hence, if we perform several Monte Carlo calculations for large enough n, we expect the realisations $\hat{x}$ of the sample mean $\hat{X}$ to be spread closely around the true expected value $E[X]$ which we want to estimate.

4.3 Monte Carlo techniques for solving source problems

Monte Carlo methods are often divided into two different classes. The first one is Monte Carlo in the sense of statistical sampling and is sometimes called "Monte Carlo simulation" [88] or "analog Monte Carlo sampling" [80, §7-3]. The basic idea of this approach is to obtain an approximation to the physical quantity of interest (in this section it will be the neutron flux) by collecting information from the particle histories during the course of simulation.

The second class of methods uses Monte Carlo techniques to evaluate integrals or integral equations. Both approaches are applied in the context of neutron transport [29, p. 553].

4.3.1 Pure absorber case

To simplify the discussion let us start with considering a monoenergetic source problem in a homogeneous region without any scatter and fission. We assume vacuum boundary conditions and the problem becomes

$$\mathcal{T}\Psi \;=\; Q \tag{4.10}$$

subject to

$$\Psi(\mathbf{r},\mathbf{\Omega}) \;=\; 0 \quad \text{when} \quad \mathbf{n}(\mathbf{r})\cdot\mathbf{\Omega} \;<\; 0\,, \quad \mathbf{r}\in\partial V\,. \tag{4.11}$$

The extension of the Monte Carlo method to more general cases is relatively straightforward. We will discuss in Section 4.3.2 how to include scatter events and on page 106 how to deal with heterogeneous media.

Let us now consider the statistical sampling approach and motivate the Monte Carlo steps by establishing links to the mathematics. We pick n neutrons and track one particle after the other through the numerical experiment. Note that this allows easy parallelisation since the different neutrons do not influence each other.

The initial positions and directions for the n neutrons are determined according to the source distribution Q. For a cuboid reactor and a spatially uniform isotropic source, for example, the position of each neutron can be obtained by choosing three random numbers $\xi_i \in U(0,1)$, $i = 1,2,3$, where $U(0,1)$ denotes the uniform distribution on $(0,1)$. These ξ_i are then multiplied with the length, width and height of the system to give the (x,y,z)-coordinates of the starting position.

We then draw two further random numbers $\xi_4, \xi_5 \in U(0,1)$ to compute the direction of travel by setting $\mu = 2\xi_4 - 1$ and $\varphi = 2\pi\xi_5$, with μ being the cosine of the azimuthal angle θ and φ denoting the polar angle, see Figure 4.2.

We now have to find out how far the neutron travels, which can be done by using the probability

$$P[X \geq d] \;=\; \exp(-\sigma d) \tag{4.12}$$

for a neutron travelling a distance $X \geq d$ without having a collision. To justify (4.12) consider the transmission of a beam of particles through a purely absorbing material without any scatter and sources. For our model problem a purely absorbing system

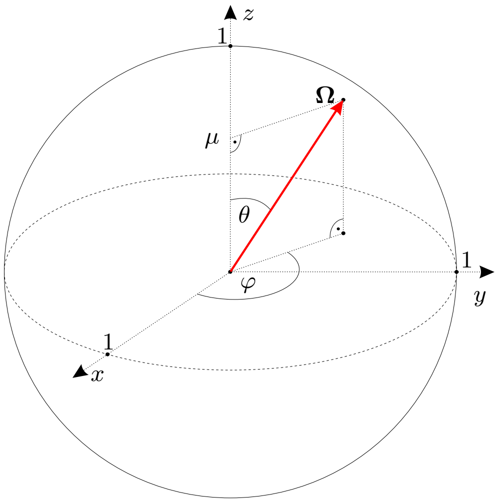


Figure 4.2: *Choosing μ and φ determines the travel direction $\mathbf{\Omega} \in \mathbb{S}^2$.*

corresponds to

$$\mathbf{\Omega} \cdot \nabla\Psi(\mathbf{r}, \mathbf{\Omega}) \;+\; \sigma\Psi(\mathbf{r}, \mathbf{\Omega}) \;=\; 0 \;. \tag{4.13}$$

Along the line from $\mathbf{r}$ to $\mathbf{r} + d\mathbf{\Omega}$ equation (4.13) can be written as

$$\frac{\mathrm{d}}{\mathrm{d}s}\Psi(\mathbf{r} + s\mathbf{\Omega}, \mathbf{\Omega}) \;+\; \sigma\Psi(\mathbf{r} + s\mathbf{\Omega}, \mathbf{\Omega}) \;=\; 0 \;, \quad s \in [0, d] \;,$$

and by using the factor $\exp(\sigma s)$, integrating from 0 to d, and rearranging, we obtain

$$\Psi(\mathbf{r} + d\mathbf{\Omega}, \mathbf{\Omega}) \;=\; \exp(-\sigma d)\Psi(\mathbf{r}, \mathbf{\Omega}) \;. \tag{4.14}$$

If we interpret (4.14) now physically, we see that the neutron flux at position $\mathbf{r} + d\mathbf{\Omega}$, i.e. at a distance d along the line, is the flux at the starting point $\mathbf{r}$ multiplied with the probability $\exp(-\sigma d)$ that the neutrons have no collision between $\mathbf{r}$ and $\mathbf{r} + d\mathbf{\Omega}$, i.e. we have justified (4.12).

Using (4.12) to look at the probability of a neutron having a collision between $\mathbf{r}$ and $\mathbf{r} + \epsilon\mathbf{\Omega}$ where ϵ is small, we get from a Taylor expansion around $-\sigma d$

$$\begin{aligned} P[d \le X \le d + \epsilon] \;&=\; P[X \ge d] \;-\; P[X \ge d + \epsilon] \\ &=\; \exp(-\sigma d) \;-\; \exp(-\sigma(d + \epsilon)) \\ &=\; \sigma\exp(-\sigma d)\epsilon \;+\; \mathcal{O}(\epsilon^2) \;. \end{aligned}$$

By the definition (4.1) the probability distribution function for having a collision at a distance d is therefore given by

$$f(d) \;=\; \sigma \exp(-\sigma d) \; . \tag{4.15}$$

This satisfies the condition $f(d) \geq 0$ for all $d \geq 0$ and integrates to one on the interval $[0, \infty)$ that we are considering. The corresponding cumulative density function is

$$F(d) \;=\; 1 - \exp(-\sigma d) \; . \tag{4.16}$$

Using now *inverse sampling*, i.e. (4.4), we can pick a random number $\xi_6 \in U(0,1)$ which we set equal to our $F(d)$, and, by rearranging (4.16), we obtain the travel distance

$$d \;=\; -\frac{1}{\sigma} \ln(1 - \xi_6) \; .$$

However, as the uniform distribution is symmetric in $(0, 1)$, we can replace $1 - \xi_6$ by ξ_6 to get

$$d \;=\; -\frac{1}{\sigma} \ln(\xi_6) \; . \tag{4.17}$$

We use (4.17) to let the neutron undergo a collision at the distance d along the direction $\mathbf{\Omega}$ which allows us to compute the coordinates of the collision point. If this point lies outside of the reactor dimensions, the neutron left the system and we move on to the next neutron. If the collision point was inside the system boundaries, we now relate this collision event to the angular flux Ψ.

From Lemma 2.1 we know that the solution to (4.10) is given by

$$\Psi(\mathbf{r}, \mathbf{\Omega}) \;=\; \int_0^{d(\mathbf{r},\mathbf{\Omega})} \exp(-\sigma s) Q(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega}) \, \mathrm{d}s \quad \text{for} \quad (\mathbf{r}, \mathbf{\Omega}) \in V \times \mathbb{S}^2 \; ,$$

where

$$d(\mathbf{r}, \mathbf{\Omega}) \;=\; \inf\{s > 0 \, : \, \mathbf{r} - s\mathbf{\Omega} \notin V\} \; .$$

Using (4.15) this can be written as

$$\Psi(\mathbf{r}, \mathbf{\Omega}) \;=\; \frac{1}{\sigma} \int_0^{d(\mathbf{r},\mathbf{\Omega})} f(s) Q(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega}) \, \mathrm{d}s \quad \text{for} \quad (\mathbf{r}, \mathbf{\Omega}) \in V \times \mathbb{S}^2 \; . \tag{4.18}$$

Now $Q(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega})$ determines the number of neutrons that are launched at $\mathbf{r} - s\mathbf{\Omega}$ in the direction $\mathbf{\Omega}$ and $f(s)$ gives the probability to travel a distance s. Hence the integrand $f(s)Q(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega})$ denotes the amount of collisions at the point $\mathbf{r}$, caused by neutrons that started at $\mathbf{r} - s\mathbf{\Omega}$ and travelled in the direction $\mathbf{\Omega}$. Integrating this over $s \in [0, d(\mathbf{r}, \mathbf{\Omega})]$ therefore results in counting all the collisions at $\mathbf{r}$ of neutrons that travelled in the direction $\mathbf{\Omega}$.

If we define the *collision density*

$$c(\mathbf{r}, \mathbf{\Omega}) \;=\; \int_0^{d(\mathbf{r},\mathbf{\Omega})} f(s)Q(\mathbf{r} - s\mathbf{\Omega}, \mathbf{\Omega})\,\mathrm{d}s$$

for $(\mathbf{r}, \mathbf{\Omega}) \in V \times \mathbb{S}^2$, then we obtain from (4.18) that

$$\Psi(\mathbf{r}, \mathbf{\Omega}) \;=\; \frac{1}{\sigma} c(\mathbf{r}, \mathbf{\Omega}) \;. \tag{4.19}$$

The relationship (4.19) between the angular flux and the collision density is well known (e.g. [80, p. 18]), but it is usually not derived via the solution of a source problem.

When estimating the angular flux in practice, we introduce a spatial and angular mesh, and it is common to call the individual mesh elements *bins*. Whenever there is a collision of a neutron that travelled in the cone of directions $\mathrm{d}\mathbf{\Omega}$ about $\mathbf{\Omega}$ in the mesh element centred at $\mathbf{r}$ with volume $\mathrm{d}V$, we add $w/(\sigma \mathrm{d}V \mathrm{d}\mathbf{\Omega})$ to the corresponding bin, where $\mathrm{d}V\mathrm{d}\mathbf{\Omega}$ denotes the incremental volume in the spatial and angular space and

$$w \;=\; \frac{1}{n} \int_V \int_{\mathbb{S}^2} |Q(\mathbf{r}', \mathbf{\Omega}')|\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}\mathbf{r}'$$

represents the particle weight. This particle weight is a scaling factor to ensure that the flux scales correctly with the number of neutrons used as well as the magnitude of the source. By repeating the tracking for all n neutrons, we subsequently build up a flux profile. The final tally of these values then represents an estimate of the angular flux in the spatial and angular element around $\Psi(\mathbf{r}, \mathbf{\Omega})$.

A sufficiently large number of particles is needed to obtain meaningful results and if we wanted to increase the detail in our results by choosing a finer mesh, we would need to increase the number of neutrons. We will consider this question in the numerical results in Section 4.5.

Remark 4.1. *The equation* (4.18) *also allows a different interpretation of the Monte Carlo method described above. If we define* $F(s) := 1 - \exp(-\sigma s)$, *then* $F'(s) = f(s)$ *with* f *defined in* (4.15). *Now* (4.18) *becomes after a change of variable with* $x = F(s)$

$$\Psi(\mathbf{r}, \mathbf{\Omega}) \;=\; \frac{1}{\sigma} \int_0^{F(d(\mathbf{r},\mathbf{\Omega}))} Q\big(\mathbf{r} - F^{-1}(x)\mathbf{\Omega}, \mathbf{\Omega}\big)\,\mathrm{d}x \quad \textit{for} \quad (\mathbf{r}, \mathbf{\Omega}) \in V \times \mathbb{S}^2 \;.$$

Using Monte Carlo techniques for the evaluation of the right-hand side then gives

$$\Psi(\mathbf{r},\mathbf{\Omega}) \;\cong\; \frac{1}{\sigma n}\sum_{j=1}^{n} Q(\mathbf{r}-F^{-1}(x_j)\mathbf{\Omega},\mathbf{\Omega}) \;,$$

with x_j uniformly distributed randomly chosen points, which is very close to the method described above.

Remark 4.2. *When working with large real problems it is very expensive on the computer storage to save all the angular fluxes. As the scalar flux often contains sufficient information for the engineer, commercial computer codes tend to only store the* scalar *flux. In that case only spatial bins exist (see also [80, p. 310]) and the storage demand is considerably reduced.*

Remark 4.3. *Source problems with a (partly) negative source (see Chapter 5 for examples) can be solved efficiently by changing the weight of a particle that represents the negative source at a position* $\mathbf{r}$ *and direction* $\mathbf{\Omega}$ *to*

$$w \;=\; -\frac{1}{n}\int_V\int_{\mathbb{S}^2} |Q(\mathbf{r},\mathbf{\Omega})|\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}\mathbf{r} \;.$$

Adjusting and scaling the weight of the particles is a frequent method in variance reduction techniques (see, for example, [108]), which we mentioned in Section 4.2.

4.3.2 Absorber and scatter case

We now extend the previous results to include isotropic scatter. The source problem is then

$$(\mathcal{T}-\mathcal{S})\Psi \;=\; Q \;, \tag{4.20}$$

where the scatter operator is given by

$$\mathcal{S}\Psi \;=\; \frac{\sigma_s}{4\pi}\int_{\mathbb{S}^2}\Psi(\mathbf{r},\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}' \;. \tag{4.21}$$

We demand again vacuum boundary conditions (4.11).

A frequent technique for the solution of source problems of the form (4.20), which is also used by deterministic solvers (e.g. the method of characteristics, see [8, 115]), is the so-called *source iteration* [48, 53]. The scatter "source" $\mathcal{S}\Psi$ is moved to the right-hand side and subsequent "transport sweeps" to invert $\mathcal{T}$ are performed before adjusting the source on the right-hand side, leading to an iteration of the form

$$\mathcal{T}\Psi^{(i+1)} \;=\; \widetilde{Q}^{(i)}, \quad \text{where} \quad \widetilde{Q}^{(i)} \;=\; Q \,+\, \mathcal{S}\Psi^{(i)} \;.$$

The same approach is taken in splitting methods for solving linear systems, as, for example, in the Jacobi method. Let us assume we want to find the solution of the linear system

$$A\mathbf{x} \;=\; \mathbf{b}\;, \quad A \in \mathbb{R}^{n\times n}\,,\; \mathbf{b},\mathbf{x} \in \mathbb{R}^n\;. \tag{4.22}$$

The matrix A is then split up into its diagonal and lower + upper triangular parts ($A = D + L + U$) and the linear system is rearranged to formulate the iteration

$$D\mathbf{x}^{(i+1)} \;=\; \mathbf{b} - (L+U)\mathbf{x}^{(i)}\;.$$

Under certain conditions – the standard criterion is that the spectral radius of $D^{-1}(L+U)$ is less than unity – this scheme then converges to the solution $\mathbf{x}$ of (4.22) (see e.g. [45, 100] for further details).

In the source iteration for (4.20) the splitting is done by moving the scatter part on to the right-hand side of the equation. We now give a convergence result for this iteration. An equivalent result for a 1D discretisation, exploiting the structure of the matrices involved, has been shown in [48, Theorem 9.2.1].

Lemma 4.4. *Suppose that in order to solve the problem*

$$(\mathcal{T} - \mathcal{S})\Psi \;=\; Q \tag{4.23}$$

for some fixed source $Q \in L^2(V, L^\infty(\mathbb{S}^2))$*, we choose a starting guess* $\Psi^{(0)}$ *and apply the source iteration*

$$\mathcal{T}\Psi^{(i+1)} \;=\; Q \,+\, \mathcal{S}\Psi^{(i)}\;, \quad i = 0,1,2,\dots\;. \tag{4.24}$$

We then obtain for the corresponding scalar fluxes $\phi^{(i)} = \mathcal{P}\Psi^{(i)}$*, with* $\mathcal{P}$ *defined in* (2.6) *that*

$$\|\phi - \phi^{(i+1)}\|_{L^2(V)} \;\le\; \frac{\sigma_s}{\sigma}\;\|\phi - \phi^{(i)}\|_{L^2(V)}\;,$$

where $\phi \in L^2(V)$ *is the solution for the scalar flux problem*

$$(\mathcal{I} - \sigma_s\mathcal{K}_\sigma)\phi \;=\; \mathcal{P}\mathcal{T}^{-1}Q\;. \tag{4.25}$$

Hence, when $0 < \sigma_s < \sigma$ *, we obtain* $\|\phi - \phi^{(i)}\|_{L^2(V)} \to 0$ *for* $i \to \infty$ *.*

Proof. Using (4.24), the existence of $\mathcal{T}^{-1}$ from Lemma 2.1 and that $\mathcal{S}\Psi = \sigma_s \mathcal{P}\Psi = \sigma_s \phi$, we have

$$\begin{aligned} \Psi - \Psi^{(i+1)} &= \Psi - (\mathcal{T}^{-1}Q + \mathcal{T}^{-1}\mathcal{S}\Psi^{(i)}) \\ &= \Psi - (\mathcal{T}^{-1}(\mathcal{T} - \mathcal{S})\Psi + \mathcal{T}^{-1}\mathcal{S}\Psi^{(i)}) \\ &= \mathcal{T}^{-1}\mathcal{S}(\Psi - \Psi^{(i)}) \\ &= \sigma_s \mathcal{T}^{-1}(\phi - \phi^{(i)}) \ . \end{aligned} \tag{4.26}$$

Hence applying the operator $\mathcal{P}$ from (2.6) and using that $\mathcal{P}\mathcal{T}^{-1}g(\mathbf{r}) = \mathcal{K}_\sigma g(\mathbf{r})$ as shown in Lemma 2.3, we get

$$\|\phi - \phi^{(i+1)}\|_{L^2(V)} \ = \ \|\sigma_s \mathcal{P}\mathcal{T}^{-1}(\phi - \phi^{(i)})\|_{L^2(V)} \ = \ \|\sigma_s \mathcal{K}_\sigma(\phi - \phi^{(i)})\|_{L^2(V)} \ .$$

The norm estimate from Lemma 2.9 now yields

$$\|\phi - \phi^{(i+1)}\|_{L^2(V)} \ \leq \ \sigma_s \|\mathcal{K}_\sigma\|_{\mathscr{L}(L^2(V))} \|\phi - \phi^{(i)}\|_{L^2(V)} \ \leq \ \frac{\sigma_s}{\sigma} \|\phi - \phi^{(i)}\|_{L^2(V)} \ .$$

The convergence of $\phi^{(i)}$ to ϕ for $i \to \infty$ follows from $\sigma_s/\sigma < 1$. □

Remark 4.5. *The assumption* $0 < \sigma_s < \sigma$ *is satisfied for physically relevant systems as for these* $\sigma_c > 0$ *in the nuclear reactor and therefore* (2.1) *implies that the scatter cross-section is strictly smaller than the total cross-section.*

However, note that Lemma 4.4 suggests that the convergence of the source iteration depends on the scattering ratio σ_s/σ *and could be slow for problems where this is close to unity. We will consider this question numerically in Section 4.5.*

In a similar way we can obtain convergence for the angular fluxes $\Psi^{(i)}$.

Corollary 4.6. *Under the assumptions of Lemma 4.4, the angular fluxes* $\Psi^{(i)}$ *generated by the source iteration* (4.24) *converge in* $L^2(V, L^1(\mathbb{S}))$ *to the solution* Ψ *of* (4.23)*.*

Proof. From (4.26) and Lemma 2.1 we have

$$\Psi - \Psi^{(i+1)} \ = \ \sigma_s \mathcal{T}^{-1}(\phi - \phi^{(i)}) \ = \ \sigma_s \int_0^{d(\mathbf{r},\mathbf{\Omega})} \exp(-\sigma s)(\phi - \phi^{(i)})(\mathbf{r} - s\mathbf{\Omega})\, \mathrm{d}s$$

for $(\mathbf{r}, \mathbf{\Omega}) \in V \times \mathbb{S}^2$. Now taking the absolute value leads to

$$|(\Psi - \Psi^{(i+1)})(\mathbf{r}, \mathbf{\Omega})| \ \leq \ \sigma_s \int_0^{d(\mathbf{r},\mathbf{\Omega})} \exp(-\sigma s)|(\phi - \phi^{(i)})(\mathbf{r} - s\mathbf{\Omega})| \,\mathrm{d}s \ ,$$

and if we integrate over $\mathbb{S}^2$, we obtain by using analogous steps as in the proof for Lemma 2.3

$$\int_{\mathbb{S}^2} |(\Psi - \Psi^{(i+1)})(\mathbf{r}, \mathbf{\Omega})| \,\mathrm{d}\mathbf{\Omega} \ \leq \ \sigma_s \mathcal{K}_\sigma(|\phi - \phi^{(i)}|)(\mathbf{r}) \ .$$

Finally, we take the L^2-norm on both sides of this inequality and apply Lemma 2.9 to get

$$\|\Psi - \Psi^{(i+1)}\|_{L^2(V, L^1(\mathbb{S}))} \ \leq \ \sigma_s \|\mathcal{K}_\sigma(|\phi - \phi^{(i)}|)\|_{L^2(V)} \ \leq \ \frac{\sigma_s}{\sigma} \|\phi - \phi^{(i)}\|_{L^2(V)} \ . \qquad (4.27)$$

Since we showed in Lemma 4.24 that $\|\phi - \phi^{(i)}\|_{L^2(V)} \to 0$ for $i \to \infty$, it follows from (4.27) that under the same assumptions $\Psi^{(i+1)} \to \Psi$ for $i \to \infty$ in $L^2(V, L^1(\mathbb{S}))$. □

Furthermore, we have an equivalence between the solution of the angular flux source problem and its scalar flux counterpart as the following lemma shows.

Lemma 4.7. *If Ψ is a solution in $L^2(V, L^1(\mathbb{S}^2))$ of (4.20) with $Q \in L^2(V, L^\infty(\mathbb{S}^2))$ and vacuum boundary conditions (4.11), then $\phi = \mathcal{P}\Psi$, is a solution in $L^2(V)$ of the reduced source problem (4.25). Conversely, if $\phi \in L^2(V)$ is a solution of problem (4.25), and if we define Ψ by solving*

$$\mathcal{T}\Psi \ = \ Q + \sigma_s \phi \qquad (4.28)$$

subject to vacuum boundary conditions (4.11), then $\Psi \in L^2(V, L^1(\mathbb{S}^2))$ is a solution for (4.20).

Proof. If Ψ solves (4.20) in $L^2(V, L^1(\mathbb{S}^2))$, then

$$\mathcal{T}\Psi \ = \ Q + \sigma_s \phi \ .$$

Now applying $\mathcal{T}^{-1}$ via Lemma 2.1 and integrating over $\mathbf{\Omega} \in \mathbb{S}^2$, we obtain

$$\phi \ = \ \mathcal{P}\mathcal{T}^{-1}Q + \sigma_s \mathcal{P}\mathcal{T}^{-1}\phi \ .$$

Since $\phi \in L^2(V)$, we can use the same steps in the proof of Lemma 2.3 to obtain that $\sigma_s \mathcal{P}\mathcal{T}^{-1}\phi = \sigma_s \mathcal{K}_\sigma \phi$ and therefore (4.25) holds.

To prove the converse statement, let Ψ be the unique solution of (4.28) with ϕ solving (4.25), and set $\widetilde{\phi} := \mathcal{P}\Psi$. Applying $\mathcal{P}\mathcal{T}^{-1}$ to (4.28) and using the steps from Lemma 2.3 as well as (4.25), we obtain

$$\widetilde{\phi} \;=\; \mathcal{P}\mathcal{T}^{-1}Q + \sigma_s\mathcal{K}_\sigma\phi \;=\; \phi\ .$$

Hence $\mathcal{T}\Psi = Q + \sigma_s\widetilde{\phi} = Q + \mathcal{S}\Psi$, as required. The fact that $\Psi \in L^2(V, L^1(\mathbb{S}^2))$ follows from the same arguments as in Remark 2.4 with $f(\mathbf{r}) = \|Q(\mathbf{r},\cdot)\|_\infty + \sigma_s\phi(\mathbf{r})$. □

We now establish a link between the source iteration and a Neumann series solution for (4.25). This link has been briefly mentioned in [29, §2.2.2] and [31, §3.4], but we are now able to provide a condition for the convergence of the Neumann series. We obtain from using (4.24) recursively that

$$\begin{aligned}
\Psi^{(i+1)} &= \mathcal{T}^{-1}Q \,+\, \mathcal{T}^{-1}\mathcal{S}\Psi^{(i)} \\
&= \mathcal{T}^{-1}Q \,+\, \mathcal{T}^{-1}\mathcal{S}(\mathcal{T}^{-1}Q \,+\, \mathcal{T}^{-1}\mathcal{S}\Psi^{(i-1)}) \\
&= \sum_{k=0}^{1}(\mathcal{T}^{-1}\mathcal{S})^k\mathcal{T}^{-1}Q \,+\, (\mathcal{T}^{-1}\mathcal{S})^2\Psi^{(i-1)} \\
&\vdots \\
&= \sum_{k=0}^{i}(\mathcal{T}^{-1}\mathcal{S})^k\mathcal{T}^{-1}Q \,+\, (\mathcal{T}^{-1}\mathcal{S})^{(i+1)}\Psi^{(0)}\ .
\end{aligned} \tag{4.29}$$

Hence, after applying $\mathcal{P}$, we get with $\mathcal{S} = \sigma_s\mathcal{P}$

$$\begin{aligned}
\phi^{(i+1)} &= \sum_{k=0}^{i}\mathcal{P}(\mathcal{T}^{-1}\mathcal{S})^k\mathcal{T}^{-1}Q \,+\, \mathcal{P}(\mathcal{T}^{-1}\mathcal{S})^{(i+1)}\Psi^{(0)} \\
&= \sum_{k=0}^{i}(\sigma_s\mathcal{P}\mathcal{T}^{-1})^k\mathcal{P}\mathcal{T}^{-1}Q \,+\, (\sigma_s\mathcal{P}\mathcal{T}^{-1})^{(i+1)}\mathcal{P}\Psi^{(0)} \\
&= \sum_{k=0}^{i}(\sigma_s\mathcal{K}_\sigma)^k\mathcal{P}\mathcal{T}^{-1}Q \,+\, (\sigma_s\mathcal{K}_\sigma)^{(i+1)}\phi^{(0)}\ .
\end{aligned}$$

Now using Lemma 2.9 and that $\sigma_s < \sigma$, we have $\|\sigma_s\mathcal{K}_\sigma\|_{L^2(V)} \le \sigma_s/\sigma < 1$, and therefore the last term tends for $i \to \infty$ to zero and we obtain in the limit

$$\phi^{(\infty)}(\mathbf{r}) \;=\; \sum_{k=0}^{\infty}(\sigma_s\mathcal{K}_\sigma)^k\mathcal{P}\mathcal{T}^{-1}Q(\mathbf{r},\mathbf{\Omega})\ .$$

This is equivalent to applying the Neumann series to solve (4.25) as we shall see now.

Lemma 4.8. *The Neumann series*

$$(\mathcal{I} - \sigma_s \mathcal{K}_\sigma)^{-1} \;=\; \sum_{k=0}^{\infty} (\sigma_s \mathcal{K}_\sigma)^k \tag{4.30}$$

converges in $L^2(V)$. Hence the solution of (4.25) *is given by*

$$\begin{aligned} \phi &= \sum_{k=0}^{\infty} (\sigma_s \mathcal{K}_\sigma)^k \mathcal{P}\mathcal{T}^{-1} Q \\ &= \sum_{k=0}^{\infty} \mathcal{P}(\mathcal{T}^{-1}\mathcal{S})^k \mathcal{T}^{-1} Q \\ &= \mathcal{P}\mathcal{T}^{-1}Q \;+\; \mathcal{P}\mathcal{T}^{-1}\mathcal{S}\mathcal{T}^{-1}Q \;+\; \mathcal{P}\mathcal{T}^{-1}\mathcal{S}\mathcal{T}^{-1}\mathcal{S}\mathcal{T}^{-1}Q \;+\; \ldots \;. \end{aligned}$$

Proof. From Lemma 2.9 we know that with $\sigma_s < \sigma$, we have

$$\|\sigma_s \mathcal{K}_\sigma\|_{\mathscr{L}(L^2(V))} \;\le\; \frac{\sigma_s}{\sigma} \;<\; 1 \;,$$

and therefore the series in (4.30) converges in $L^2(V)$. The remainder of this lemma follows from the application of the Neumann series to (4.25) and recalling that $\sigma_s \mathcal{K}_\sigma \mathcal{P} = \sigma_s \mathcal{P}\mathcal{T}^{-1}\mathcal{P} = \mathcal{P}\mathcal{T}^{-1}\mathcal{S}$. □

By Corollary 4.6 we know that the source iteration (4.24) converges for physically relevant source problems of the form (4.23). We also proved in Lemma 4.8 that for the corresponding scalar fluxes the source iteration is in the limit equivalent to using the Neumann series for the analytic solution of (4.25).

We shall now describe a Monte Carlo approach to solve (4.23) which performs the source iteration (4.24) with starting guess $\Psi^{(0)} = 0$ and efficiently implements the iteration process. Instead of rebuilding the source in every step, we use (4.29) which gives in the limit

$$\begin{aligned} \Psi^{(\infty)}(\mathbf{r},\mathbf{\Omega}) &= \sum_{k=0}^{\infty} (\mathcal{T}^{-1}\mathcal{S})^k \mathcal{T}^{-1} Q(\mathbf{r},\mathbf{\Omega}) \\ &= \sum_{k=0}^{\infty} \Psi_{[k]}(\mathbf{r},\mathbf{\Omega}) \;, \quad \text{where} \\ \Psi_{[k]}(\mathbf{r},\mathbf{\Omega}) &:= (\mathcal{T}^{-1}\mathcal{S})^k \mathcal{T}^{-1} Q(\mathbf{r},\mathbf{\Omega}) \;. \end{aligned} \tag{4.31}$$

Interpreting the terms in the sum (4.31) physically, we note that the first term $\Psi_{[0]} = \mathcal{T}^{-1}Q$ is the solution to the source problem (4.10), which we obtained by tracking particles from the initial source Q to their first collision. In the second term, this flux $\Psi_{[0]}$ undergoes a scatter and is then tracked further providing a flux contribution $\Psi_{[1]} = \mathcal{T}^{-1}\mathcal{S}\Psi_{[0]} = \mathcal{T}^{-1}\mathcal{S}\mathcal{T}^{-1}Q$. The next term is then a further scatter of $\Psi_{[1]}$ with subsequent tracking and so on. Hence, $\Psi_{[k-1]}$ denotes the flux contribution from the k-th collision and the Neumann series can be interpreted as a "collision expansion" (see also [29, p. 72]).

Let us now consider how to model the application of the scatter operator to a flux in the Monte Carlo setting. Recalling the relation between the flux Ψ and the collision density c in (4.19), as well as the definition of the scatter operator in (4.21), we can write the term $\mathcal{S}\Psi_{[0]}$ as

$$\mathcal{S}\Psi_{[0]} \;=\; \frac{1}{4\pi}\int_{\mathbb{S}^2} \frac{\sigma_s}{\sigma} c_{[0]}(\mathbf{r},\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}' \;, \tag{4.32}$$

where $c_{[0]}(\mathbf{r},\mathbf{\Omega}')$ is the number of particles that travelled in direction $\mathbf{\Omega}'$ and have a collision at $\mathbf{r}$. The integral $\int_{\mathbb{S}^2} c_{[0]}(\mathbf{r},\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}'$ contains therefore all first collisions occurring at $\mathbf{r}$.

In Monte Carlo terms the fraction σ_s/σ can now be interpreted as the probability of this collision being a scatter. As we are dealing with isotropic scatter, a realisation of the new travel direction $\mathbf{\Omega}$ is in this case determined by drawing two new random numbers and mapping them to μ and φ to obtain the particles new polar and azimuthal angles. The collection of all scattered neutrons then represents the scatter source $\mathcal{S}\Psi$.

The evaluation of (4.31) can now be done efficiently using Monte Carlo techniques by applying the following steps. We start a particle from the initial source Q, track it to the first collision and add $w/(\sigma \mathrm{d}V \mathrm{d}\mathbf{\Omega})$ to the current bin. This contributes to the flux $\Psi^{(\infty)}$ as part of the term $\Psi_{[0]} = \mathcal{T}^{-1}Q$ in (4.31). We now pick a random number $\xi \in U(0,1)$ to check if the particle is scattered as shown in Figure 4.3.

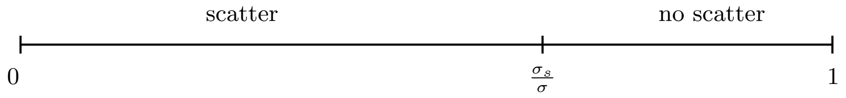


Figure 4.3: *If $\xi \le \sigma_s/\sigma$ the collision event is a scatter.*

If $\xi > \sigma_s/\sigma$, we finish tracking the particle and all its contribution to the flux $\Psi^{(\infty)}$ comes from the term $\Psi_{[0]}$ representing the first collisions in the Neumann series. However, if the random number indicates that a scatter event happened, we draw two further random numbers to obtain a new travel direction and track the particle to the next collision where $w/(\sigma \mathrm{d}V \mathrm{d}\mathbf{\Omega})$ is added to the respective bin. This is part of the contribution of $\Psi_{[1]} = \mathcal{T}^{-1}\mathcal{S}\mathcal{T}^{-1}Q$ to the flux $\Psi^{(\infty)}$ in (4.31). Then the same scatter or no scatter check is performed and the above steps are repeated until the neutron eventually stops scattering (which will happen since $\sigma_s < \sigma$) or leaves the system. Hence, we approximate the infinite sum in the collision expansion (4.31) in practice by a finite number of collision terms.

We note that this is also analogous to the situation in the real world, where particles start from a given source, travel some distance until they either leave the system or have a collision. If this collision results in a scatter, the neutron changes its direction and travels further. By following all neutrons from the source and "counting" where the neutrons had a collision, we build up a flux profile in the system. The same happens in a nuclear reactor when detectors are used to obtain a flux estimate. This similarity is the reason why the method is called "analog" or "simulation" approach. The only difference of our Monte Carlo approach is that we do not track as many neutrons as there are in the real world. However, we were able to link this simulation approach via the source iteration and the relation to the Neumann series to the mathematical solution of the integral equation.

4.3.3 Absorber, scatter and fission case

The above results can be extended in a straightforward way to problems of the form

$$(\mathcal{T} - \mathcal{S} - \alpha\mathcal{F})\Psi \;=\; Q \;, \tag{4.33}$$

as long as the shift α is chosen such that

$$\|\mathcal{T}^{-1}(\mathcal{S} + \alpha\mathcal{F})\| \;<\; 1 \;. \tag{4.34}$$

Corollary 4.9. *Provided the operators and α in the source problem* (4.33) *satisfy the criterion* (4.34)*, then the source iteration*

$$\mathcal{T}\Psi^{(i+1)} \;=\; Q + (\mathcal{S} + \alpha\mathcal{F})\Psi^{(i)} \;, \quad i = 0, 1, 2, \ldots \;.$$

converges to a solution Ψ in $L^2(V, L^1(\mathbb{S}^2))$ of (4.33)*. A sufficient condition to obtain convergence in the case of homogeneous problems with isotropic scattering is given by*

$$\frac{\sigma_s + \alpha\nu\sigma_f}{\sigma} \;<\; 1 \;.$$

Proof. The result follows analogous to the proofs of Lemma 4.4 and Corollary 4.6. □

Therefore, using the Neumann series expansion and similar steps as above, we can rearrange (4.33) to

$$\begin{aligned}\Psi &= \sum_{k=0}^{\infty}[\mathcal{T}^{-1}(\mathcal{S}+\alpha\mathcal{F})]^k\mathcal{T}^{-1}Q \\ &= \mathcal{T}^{-1}Q \ + \ \mathcal{T}^{-1}(\mathcal{S}+\alpha\mathcal{F})\mathcal{T}^{-1}Q \ + \ \ldots \\ &= \mathcal{T}^{-1}Q \ + \ \mathcal{T}^{-1}\mathcal{S}\mathcal{T}^{-1}Q \ + \ \alpha\mathcal{T}^{-1}\mathcal{F}\mathcal{T}^{-1}Q \ + \ \ldots \ .\end{aligned}$$

The term $\alpha\mathcal{T}^{-1}\mathcal{F}\mathcal{T}^{-1}Q$ is now evaluated similarly to $\mathcal{T}^{-1}\mathcal{S}\mathcal{T}^{-1}Q$ in the previous section, writing

$$\alpha\mathcal{F}\Psi_{[0]} \ = \ \alpha\nu\frac{1}{4\pi}\int_{\mathbb{S}^2}\frac{\sigma_f}{\sigma}c_{[0]}(\mathbf{r},\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}' \ , \tag{4.35}$$

where $c_{[0]}(\mathbf{r},\mathbf{\Omega}')$ is again the number of particles that travelled in direction $\mathbf{\Omega}'$ and have a first collision at $\mathbf{r}$. The fraction σ_f/σ is the probability that the collision results in a fission event. If this happens, $\alpha\nu$ new neutrons are sent off with an equal probability for every direction and are tracked to their next collision.

4.4 Monte Carlo as a solution method for the criticality problem

While Monte Carlo methods had initially been used to solve source problems as discussed in the previous section, during the 1960s the neutron transport community started to apply them to compute the criticality of nuclear reactors (see, for example, [81, 84]). One way to solve the eigenvalue problem (1.3) is to use again a simulation approach and track particles through the system. The estimate for λ is then obtained by counting how many neutrons are lost and how many are gained during the process.

Although motivated by simply simulating the physical interactions of the particles, the approach can be considered as an implementation of the power method using Monte Carlo techniques to solve

$$(\mathcal{T}-\mathcal{S})\widetilde{\Psi}^{(i+1)} \ = \ \mathcal{F}\Psi^{(i)} \ .$$

The right-hand side $\mathcal{F}\Psi^{(i+1)}$ of the next iteration is obtained by normalising $\mathcal{F}\widetilde{\Psi}^{(i+1)}$. An eigenvalue estimate is computed in every iteration by evaluating a certain Rayleigh quotient. We first describe the method by following the idea of simulating the individual neutrons but we will show below that this is equivalent to the above mentioned implementation of the power method.

Let us start by motivating how to get an estimate for the criticality value λ from a Monte Carlo simulation. Consider the original eigenvalue problem

$$\begin{aligned} & \boldsymbol{\Omega} \cdot \nabla \Psi(\mathbf{r}, E, \boldsymbol{\Omega}) \;+\; \sigma(\mathbf{r}, E) \Psi(\mathbf{r}, E, \boldsymbol{\Omega}) \\ = \; & \frac{1}{4\pi} \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \sigma_s(\mathbf{r}, E', E, \boldsymbol{\Omega}', \boldsymbol{\Omega}) \Psi(\mathbf{r}, E', \boldsymbol{\Omega}') \,\mathrm{d}\boldsymbol{\Omega}' \,\mathrm{d}E' \\ & + \lambda \, \frac{\chi(E)}{4\pi} \int_{\mathbb{R}^+} \nu(\mathbf{r}, E') \sigma_f(\mathbf{r}, E') \int_{\mathbb{S}^2} \Psi(\mathbf{r}, E', \boldsymbol{\Omega}') \,\mathrm{d}\boldsymbol{\Omega}' \,\mathrm{d}E' \;, \end{aligned} \tag{4.36}$$

as given in (1.3), and recall that the total cross-section is defined by

$$\sigma(\mathbf{r}, E) \;=\; \sigma_c(\mathbf{r}, E) \;+\; \frac{1}{4\pi} \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \sigma_s(\mathbf{r}, E, E', \boldsymbol{\Omega}, \boldsymbol{\Omega}') \,\mathrm{d}\boldsymbol{\Omega}' \,\mathrm{d}E' \;+\; \sigma_f(\mathbf{r}, E) \;.$$

We insert this expression of σ in the left-hand side of (4.36) and integrate both sides of the equation over space, energy and direction to get

$$\begin{aligned} & \int_V \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \boldsymbol{\Omega} \cdot \nabla \Psi(\mathbf{r}, E, \boldsymbol{\Omega}) \,\mathrm{d}\boldsymbol{\Omega} \,\mathrm{d}E \,\mathrm{d}\mathbf{r} \\ & + \int_V \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} [\sigma_c(\mathbf{r}, E) + \sigma_f(\mathbf{r}, E)] \Psi(\mathbf{r}, E, \boldsymbol{\Omega}) \,\mathrm{d}\boldsymbol{\Omega} \,\mathrm{d}E \,\mathrm{d}\mathbf{r} \\ & + \frac{1}{4\pi} \int_V \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \sigma_s(\mathbf{r}, E, E', \boldsymbol{\Omega}, \boldsymbol{\Omega}') \Psi(\mathbf{r}, E, \boldsymbol{\Omega}) \,\mathrm{d}\boldsymbol{\Omega}' \,\mathrm{d}E' \,\mathrm{d}\boldsymbol{\Omega} \,\mathrm{d}E \,\mathrm{d}\mathbf{r} \\ = \; & \frac{1}{4\pi} \int_V \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \sigma_s(\mathbf{r}, E', E, \boldsymbol{\Omega}', \boldsymbol{\Omega}) \Psi(\mathbf{r}, E', \boldsymbol{\Omega}') \,\mathrm{d}\boldsymbol{\Omega}' \,\mathrm{d}E' \,\mathrm{d}\boldsymbol{\Omega} \,\mathrm{d}E \,\mathrm{d}\mathbf{r} \\ & + \lambda \int_V \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \chi(\mathbf{r}, E) \int_{\mathbb{R}^+} \frac{1}{4\pi} \nu(\mathbf{r}, E') \sigma_f(\mathbf{r}, E') \int_{\mathbb{S}^2} \Psi(\mathbf{r}, E', \boldsymbol{\Omega}') \,\mathrm{d}\boldsymbol{\Omega}' \,\mathrm{d}E' \,\mathrm{d}\boldsymbol{\Omega} \,\mathrm{d}E \,\mathrm{d}\mathbf{r} \,. \end{aligned}$$

By changing the order of integration, we see that the two scattering terms are equal and can be eliminated. The integrand of the fission term is independent of the outgoing travel direction $\boldsymbol{\Omega}$, allowing us to simplify the integral. Now, assuming the fission integral is non-zero (this is true for problems of physical interest since the angular flux is strictly positive in the interior of the reactor and χ, ν and σ_f are non-negative and non-zero), we can divide by $\langle F\Psi \rangle$ to obtain

$$\lambda \;=\; \frac{\langle L\Psi \rangle + \langle A\Psi \rangle}{\langle F\Psi \rangle} \;, \tag{4.37}$$

where

$$\begin{aligned}
\langle L\Psi\rangle &:= \int_V \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} \mathbf{\Omega}\cdot\nabla\Psi(\mathbf{r},E,\mathbf{\Omega})\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\,\mathrm{d}\mathbf{r}\ ,\\
\langle A\Psi\rangle &:= \int_V \int_{\mathbb{R}^+} \int_{\mathbb{S}^2} [\sigma_c(\mathbf{r},E)+\sigma_f(\mathbf{r},E)]\Psi(\mathbf{r},E,\mathbf{\Omega})\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\,\mathrm{d}\mathbf{r}\ , \quad \text{and}\\
\langle F\Psi\rangle &:= \int_V \int_{\mathbb{R}^+} \chi(\mathbf{r},E)\int_{\mathbb{R}^+}\nu(\mathbf{r},E')\sigma_f(\mathbf{r},E')\int_{\mathbb{S}^2}\Psi(\mathbf{r},E',\mathbf{\Omega}')\,\mathrm{d}\mathbf{\Omega}'\,\mathrm{d}E'\,\mathrm{d}E\,\mathrm{d}\mathbf{r}\ .
\end{aligned}$$

An alternative way to look at this estimate for λ is to consider the Rayleigh quotient

$$\rho \;=\; \frac{\langle f,(\mathcal{T}-\mathcal{S})\Psi\rangle}{\langle f,\mathcal{F}\Psi\rangle}\ , \tag{4.38}$$

where f is the constant function of unity on $V\times\mathbb{R}^+\times\mathbb{S}^2$ and the inner product is given by

$$\langle f,g\rangle \;=\; \int_V\int_{\mathbb{R}^+}\int_{\mathbb{S}^2} f(\mathbf{r},E,\mathbf{\Omega})g(\mathbf{r},E,\mathbf{\Omega})\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\,\mathrm{d}\mathbf{r}\ .$$

A different eigenvalue estimate can be obtained by applying $f=\Psi^*$, where Ψ^* is an approximation to the solution of the adjoint problem. This choice is used in variational variance reduction (VVR) methods that have been studied, for example, in [11, 27]. We will focus in this chapter on criticality estimates obtained from (4.37) which, as we show now, can be evaluated easily when using Monte Carlo methods.

To illustrate this we rewrite the term $\langle L\Psi\rangle$ in the numerator by using that $\nabla\cdot\mathbf{\Omega}=0$ and applying the divergence theorem (with $\mathbf{n}(\mathbf{r})$ denoting the outward normal at $\mathbf{r}$), such that

$$\begin{aligned}
\langle L\Psi\rangle &= \int_V\int_{\mathbb{R}^+}\int_{\mathbb{S}^2}\big(\mathbf{\Omega}\cdot\nabla\Psi(\mathbf{r},E,\mathbf{\Omega})+\Psi(\mathbf{r},E,\mathbf{\Omega})(\nabla\cdot\mathbf{\Omega})\big)\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\,\mathrm{d}\mathbf{r}\\
&= \int_{\mathbb{R}^+}\int_{\mathbb{S}^2}\int_V \nabla\cdot\big(\Psi(\mathbf{r},E,\mathbf{\Omega})\mathbf{\Omega}\big)\,\mathrm{d}\mathbf{r}\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\\
&= \int_{\mathbb{R}^+}\int_{\mathbb{S}^2}\int_{\partial V}\big(\Psi(\mathbf{r},E,\mathbf{\Omega})\mathbf{\Omega}\big)\cdot\mathbf{n}(\mathbf{r})\,\mathrm{d}\mathbf{r}\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\\
&= \int_{\partial V}\int_{\mathbb{R}^+}\int_{\mathbb{S}^2}\Psi(\mathbf{r},E,\mathbf{\Omega})\big(\mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r})\big)\,\mathrm{d}\mathbf{\Omega}\,\mathrm{d}E\,\mathrm{d}\mathbf{r}\ .
\end{aligned}$$

This is exactly the amount of flux that crosses the outer boundary. In the case of vacuum boundary conditions (1.6), where

$$\Psi(\mathbf{r},E,\mathbf{\Omega}) \;=\; 0 \quad \text{when} \quad \mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r}) \;<\; 0\ , \quad \mathbf{r}\in\partial V\ ,$$

$\langle L\Psi\rangle$ consists purely of neutrons that leave the system and is therefore called leakage.

In the case of reflective boundary conditions (1.7), where

$$\Psi(\mathbf{r},E,\mathbf{\Omega}) \;=\; \Psi(\mathbf{r},E,\mathbf{\Omega}') \quad \text{when} \quad \mathbf{\Omega}\cdot\mathbf{n}(\mathbf{r}) \;<\; 0\;, \quad \mathbf{r}\in\partial V\;.$$

the term $\langle L\Psi\rangle$ becomes zero.

The integral $\langle A\Psi\rangle$ contains the neutrons that are absorbed due to capture and fission events. Note that a fission is considered physically as the absorption of a neutron and the subsequent release of two to four new neutrons.

The denominator $\langle F\Psi\rangle$ of (4.37) contains the amount of newly produced neutrons from fission events. Therefore, we can describe λ in words as

$$\lambda \;=\; \frac{\text{neutrons lost due to leakage} + \text{neutrons lost by absorptions}}{\text{neutrons gained from fission events}}\;. \tag{4.39}$$

Hence, when using Monte Carlo methods, we are able to obtain an estimate for λ by simply counting how many neutrons

1. leave the system through the outer boundary (leakage),
2. are absorbed (either due to capture or when causing a fission), and
3. how many neutrons are produced in fission events.

This idea is rather simple but we still need to describe some details as to where to start neutrons from and when to stop the counting. If we were to follow every starting neutron and its fission products until they are absorbed or leave the system, we would expect for (super)critical problems to never finish tracking since the number of neutrons in the system does not decrease. To avoid this, we introduce a concept of neutron generations that correspond to different sets of neutrons which we will call *batches*.

The process can be interpreted as a Monte Carlo way of performing the power method from Algorithm 1. Algorithm 6 gives an overview of the different steps. We describe these now in more detail.

We start with an initial source $Q^{(0)}$ which can be obtained from an initial flux distribution $\Psi^{(0)}$. Finding a good initial guess greatly helps to speed the convergence of the power method up (and hence to reduce the runtime of the Monte Carlo method), but can sometimes be very difficult. A simple approach is to assume a uniform flux in all the regions with fissile material. As we are solving an eigenvalue problem, we can scale the flux arbitrarily. This initial flux then gives rise to the initial source $Q^{(0)} = \mathcal{F}\Psi^{(0)}$.

Algorithm 6 Power method – Monte Carlo approach

```
Require: Starting guess $Q^{(0)} = \mathcal{F}\Psi^{(0)}$.
  for i=0,1,2,... do
    Solve $(\mathcal{T}-\mathcal{S})\widetilde{\Psi}^{(i+1)} = Q^{(i)}$ by computing $\widetilde{\Psi}^{(i+1)} = \sum_{k=0}^{\infty} \Psi_{[k]}^{(i+1)}$,
       where $\Psi_{[k]}^{(i+1)} := (\mathcal{T}^{-1}\mathcal{S})^k\mathcal{T}^{-1}Q^{(i)}$.
    Compute $\widetilde{Q}^{(i+1)} = \mathcal{F}\widetilde{\Psi}^{(i+1)}$.
    Estimate $\lambda^{(i+1)}$ via (4.39).
    Obtain $Q^{(i+1)}$ by normalisation of $\widetilde{Q}^{(i+1)}$.
  end for
```

We now pick n neutrons which determine our first batch and distribute them according to the initial source $Q^{(0)}$. If we are only interested in the criticality value λ and not in the flux shape Ψ, we can efficiently combine the solution of the source problem

$$(\mathcal{T}-\mathcal{S})\widetilde{\Psi}^{(i+1)} = Q^{(i)} \tag{4.40}$$

by the source iteration (4.24), i.e. computing an approximation of

$$\widetilde{\Psi}^{(i+1)} = \sum_{k=0}^{\infty} \Psi_{[k]}^{(i+1)}, \quad \text{where} \quad \Psi_{[k]}^{(i+1)} = (\mathcal{T}^{-1}\mathcal{S})^k\mathcal{T}^{-1}Q^{(i)}, \tag{4.41}$$

and the computation of the new source

$$\widetilde{Q}^{(i+1)} = \mathcal{F}\widetilde{\Psi}^{(i+1)}. \tag{4.42}$$

This is done by using a Monte Carlo approach that models what happens in the real world. Particles emerge from the source $Q^{(i)}$ according to the source distribution and are tracked to their first collision. Assuming this collision point lies within the reactor, we add $w/(\sigma \mathrm{d}V\mathrm{d}\mathbf{\Omega})$ to the flux $\widetilde{\Psi}^{(i+1)}$. This contribution corresponds to $\widetilde{\Psi}_{[0]}^{(i+1)}$ in (4.41). Now three different events can happen: (i) the neutron is scattered; (ii) the neutron causes a fission; and (iii) the neutron is captured.

We saw in (4.32) that the neutrons arriving at $\mathbf{r}$ will scatter with probability σ_s/σ. We also know from (4.42) that we have to apply the fission operator to the flux $\widetilde{\Psi}^{(i+1)}$ in order to obtain the source $\widetilde{Q}^{(i+1)} = \mathcal{F}\widetilde{\Psi}^{(i+1)}$ for the next iteration. We can immediately evaluate the current particle's contribution to $\mathcal{F}\widetilde{\Psi}^{(i+1)}$ in the way described in (4.35) by considering the probability σ_f/σ that the neutron causes a fission event.

We now combine these two steps by picking a random number $\xi \in U(0,1)$ and checking in which part of the unit interval ξ falls, as shown in Figure 4.4.

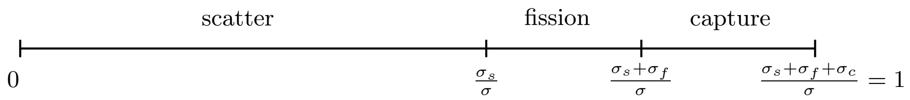

scatter
fission
capture
0
$\frac{\sigma_s}{\sigma}$
$\frac{\sigma_s+\sigma_f}{\sigma}$
$\frac{\sigma_s+\sigma_f+\sigma_c}{\sigma} = 1$

Figure 4.4: *The location of the random number determines which kind of collision happens.*

If a capture occurs, we stop tracking this particle which then only contributed to $\widetilde{\Psi}^{(i+1)}$ via the term $\widetilde{\Psi}^{(i+1)}_{[0]} = \mathcal{T}^{-1}Q^{(i)}$ in (4.41). If the random number predicted a scatter, we pick a new travel direction and follow the particle to the next collision where we repeat the above process. In the case of a fission event we stop tracking this particle but create ν new neutrons whose exact positions are stored in the new source $\widetilde{Q}^{(i+1)}$. This process is repeated until all n particles are either captured, left the system or caused a fission event. By obtaining the new source $\widetilde{Q}^{(i+1)}$ we have thus solved (4.40) and (4.42). Therefore, tracking all particles of one batch can be interpreted as performing one outer iteration of the power method.

Note also that by storing the exact location of the fission events and restarting the neutrons from these positions, we remove the dependence on a spatial and angular mesh for the criticality calculation. The discretisation, i.e. the spatial and angular bins, are only needed if, in addition to the eigenvalue, an estimate of the angular flux is desired. If this is not the case, we do not need to compute the flux $\widetilde{\Psi}^{(i+1)}$ and only store the exact positions of the new fission neutrons in $\widetilde{Q}^{(i+1)}$.

We now know that we "lost" n particles and by counting the number of particles in the source $\widetilde{Q}^{(i+1)}$, we can obtain an estimate of the criticality via (4.39). In order to easily compute an estimate for the standard deviation of our results via (4.8), we consider instead of λ its reciprocal

$$k \;=\; \frac{1}{\lambda} \;=\; \frac{\text{neutrons gained from fission events}}{\text{neutrons lost due to leakage} + \text{neutrons lost by absorptions}}\;. \qquad (4.43)$$

We can even obtain for every single simulated particle a (very bad) estimate for k. Let X denote the discrete random variable that gives via (4.43) for every neutron the number of secondary particles. If the source is derived from the time-independent flux distribution, the expectation of this random variable is $E[X] = k$.

Let $x_j^{(i)}$ denote now the estimate of k for the simulated neutron j in batch (or iteration) i. The values that the discrete random variable $X_j^{(i)}$ can attain are either zero (if the neutron leaves the system or is captured), or two, three or four, depending on the number of newly produced neutrons in the case of a fission.

We now define the average of the i-th batch as

$$k_i \;=\; \frac{1}{n}\sum_{j=1}^{n} X_j^{(i)} \;.$$

This sample mean k_i is an estimator for k and realisations can be obtained by using the data, i.e. the number of produced fission neutrons, in the i-th batch. Assuming that the $X_j^{(i)}$ are independent and identically distributed with $E[X_j^{(i)}] = E[X]$ the probability theory from Section 4.2 tells us that k_i, as a random variable, is for sufficiently large n approximately Gaussian with variance $\sigma^2[k_i] = \sigma^2[X]/n$ and expectation $E[k_i] = E[X]$. Therefore, if n is large, we expect a realisation of k_i, i.e. a realisation of the sample mean computed from n simulated particles originating from the steady-state source $\mathcal{F}\Psi$, to be a good estimate of $E[X] = k$.

However, recall that we perform a form of the power method and that the eigenvalue approximations that we obtain depend on the accuracy of the source $Q^{(i)} = \mathcal{F}\Psi^{(i)}$. As we started the neutrons from an initial distribution that is unlikely to equal the correct neutron flux, we are not simulating the true behaviour of the physical system for small i. Therefore our estimate for the criticality will not be correct in those cases.

The physical motivation to overcome this problem is to follow enough batches so that the particles can move around and cause fissions that represent the time-independent source "accurately enough". This corresponds mathematically to the convergence of the power method to the correct flux shape. In a deterministic solver the iteration would now be finished and the obtained flux would be used to estimate the eigenvalue via a Rayleigh quotient.

However, in the Monte Carlo approach these *settling stages* to "converge the source" are only the first part of the criticality computation. Knowing how many settling stages are needed is difficult and poses a major challenge for many practical problems. Several acceleration techniques have been developed over the years, such as the Wielandt acceleration discussed in [125] (which is related to shifted inverse iteration in Algorithm 2), or schemes that combine the Monte Carlo approach with deterministic methods (e.g. the p-CMFD method, see [126]). For further information on acceleration methods for neutron transport criticality problems we refer the reader to the thesis [117].

Assuming that the source is converged to its steady-state form (confirming which is another major challenge when using Monte Carlo methods), we now reduce the *statistical uncertainty* in our criticality estimate by averaging the k_i from all the following batches. This part of the calculation is denoted as the *scoring stages* and assuming we use s batches, the final estimate for $\lambda = 1/k$ is then obtained from the estimator

$$k_s = \frac{1}{s}\sum_{i=1}^{s} k_i \ .$$

Note that using the fission birth neutrons from the previous iteration as the new source in the next batch and normalising the source introduces a bias and correlations between the batches which leads to the underestimation of the confidence intervals. This problem has been discussed, for example, in [17], where it is shown that the bias in k is of the order n^{-1} and the bias in the variance is of the order $(ns)^{-1}$. The paper [17] also presents strategies such as *superhistory powering* to overcome these difficulties.

As we saw in Section 4.2, Monte Carlo methods allow to easily obtain an estimate for the standard deviation which then gives an indication of the accuracy of the computed solution. By simply counting the squares of the realisations $x_j^{(i)}$ of $X_j^{(i)}$, we can use (4.8) to obtain an estimate of $\sigma[X]$. This estimate can then be used to get a confidence interval for the criticality estimate via (4.9) as we will show on page 115 for our numerical results.

In order to normalise the source to start exactly n neutrons in every iteration, we randomly remove neutrons from $\widetilde{Q}^{(i+1)}$ (in the case of too many produced neutrons) or multiply them (if there are too few neutrons). This ensures that we pick physically sensible positions but also introduces a bias. As we are choosing the direction of travel randomly in the next step, the new neutrons should travel in different directions so that we obtain different information from different neutrons.

We claimed above that the biggest strength of the Monte Carlo method is being able to deal with complicated geometries but so far we have only considered the homogeneous case. In the following we will describe how heterogeneous problems are dealt with.

Obtaining the travel distance d via (4.17) only holds as long as σ does not change, i.e. as long as the neutron remains in the same material. We therefore check in the heterogeneous case if d is larger than the distance from the current position of the particle to the boundary of the next material. If this happens, we set the neutron to the intersection point on the material boundary and compute the new distance by generating a new random number and repeating the above process. The reason that we can use a new random number is the “memoryless property” of the exponential.

Let again $P[X \geq d] = \mathrm{e}^{-\sigma d}$ be the probability that the neutron travels a distance $X \geq d$ without having a collision. We now ask for the probability of travelling a distance $d_B + x$ provided that it already travelled a distance d_B. This conditional probability is denoted by $P[X \geq d_B + x \mid X \geq d_B]$ and can be computed as

$$\begin{aligned} P[X \geq d_B + x \mid X \geq d_B] &= \frac{P[\{X \geq d_B + x\} \cap \{X \geq d_B\}]}{P[X \geq d_B]} \\ &= \frac{P[X \geq d_B + x]}{P[X \geq d_B]} \\ &= \frac{e^{-\sigma(d_B+x)}}{e^{-\sigma d_B}} = e^{-\sigma x} = P[X \geq x] \ . \end{aligned}$$

So if we know that the particle reached the next boundary (i.e. that it travelled a distance d_B), the probability that it then travels a further distance x is the same as if we restarted the particle at the new boundary. For the next material we now use the new σ and repeat the tracking process from the new point on the boundary. This approach can be very slow for complicated geometries with rapidly varying material zones. To avoid this problem commercial codes use an acceleration scheme called *Woodcock tracking* (see [123] for the original reference).

Other extensions of the above Monte Carlo approach, such as including the energy dependence, are also reasonably straightforward but will not be discussed here.

4.5 Numerical results

The numerical results for the Monte Carlo method that we present here use the same data as the discrete ordinates method. This allows us to compare the Monte Carlo results to the discrete ordinates results since we expect both methods to converge to the same results when the number of spatial and angular mesh points, as well as the batch size, tend to infinity.

We choose the same spatial and angular mesh for the statistical sampling of the flux that we use for the deterministic method. Despite this, we cannot expect the two approaches to give the same results for a given number of spatial and angular intervals, as they are very different methods.

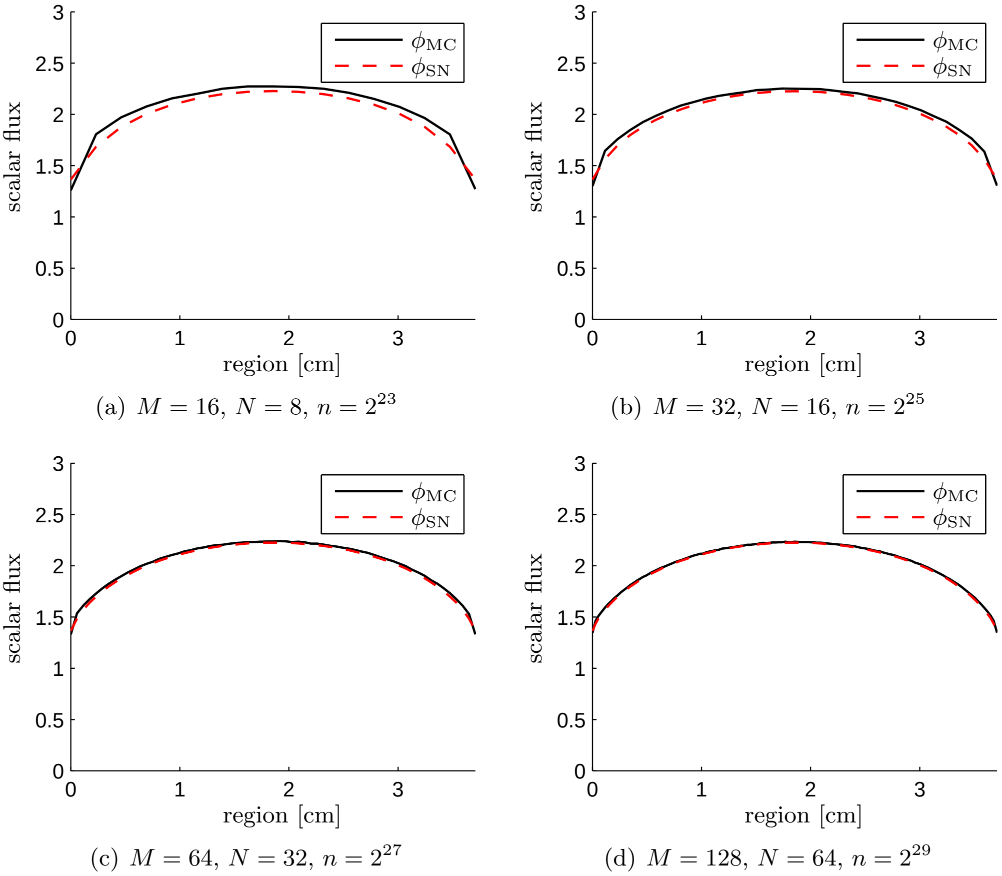
4.5.1 Numerical results for source problems

We start with comparing the numerical solutions for a 1D source problem $\mathcal{T}\Psi = Q$ when applying the discrete ordinates method from Section 3.3.3 and the analog Monte Carlo approach discussed in Section 4.3.1. We test different numbers of spatial and angular bins against varying numbers of particles that we sent off. We also change the source Q to allow for a partly negative source to simulate problems that will occur in an iterative method discussed in Chapter 5.

Let us begin with considering how the accuracy of the Monte Carlo method for a source problem depends on the number of mesh elements chosen. The graphs in Figure 4.5 show the scalar fluxes that we obtain when using Monte Carlo (ϕ_{MC}) and a discrete ordinates approach (ϕ_{SN}) for different numbers of spatial intervals (M) and angular directions ($2N$). The problem that is solved is

$$\mathcal{T}\Psi(z,\mu) \;=\; Q(z,\mu)$$

with a uniform and isotropic source $Q(z,\mu) = 1/2$ for all $z \in [0,L]$ and $\mu \in [-1,1]$, where the cross-section data is again taken from the Los Alamos benchmark test set problem number 2 which we introduced in Section 3.3.1.



(a) $M = 16$, $N = 8$, $n = 2^{23}$

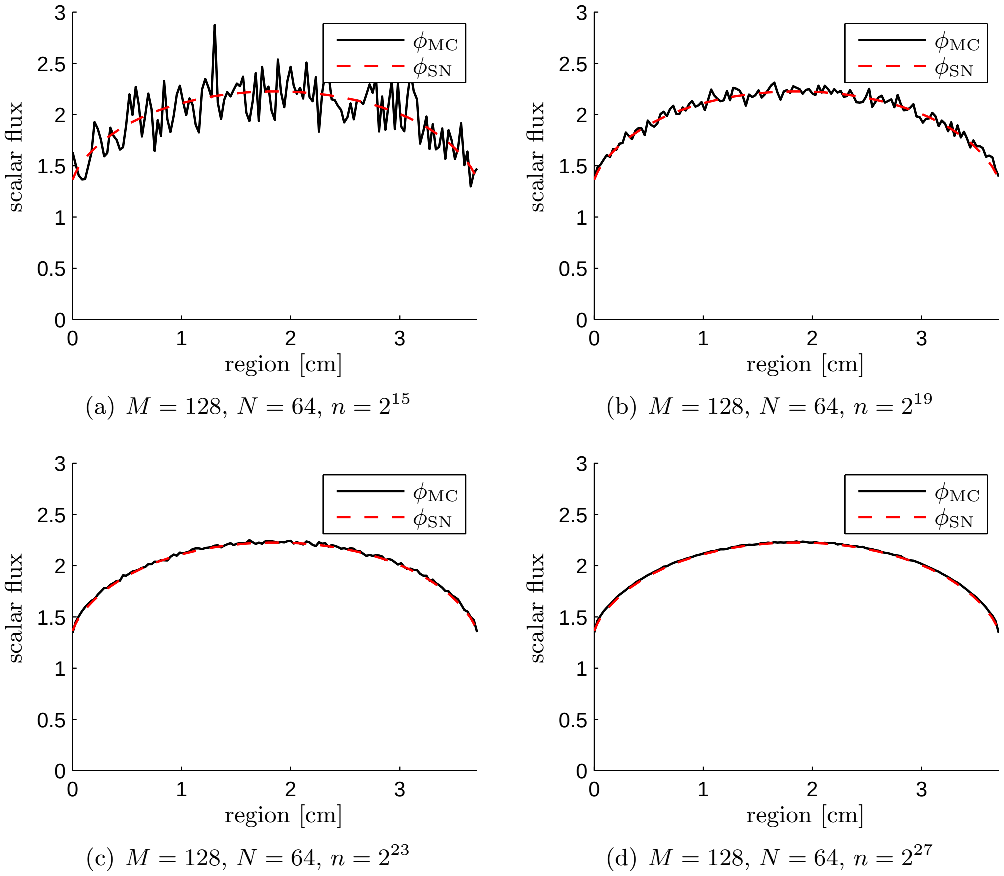
(b) $M = 32$, $N = 16$, $n = 2^{25}$

(c) $M = 64$, $N = 32$, $n = 2^{27}$

(d) $M = 128$, $N = 64$, $n = 2^{29}$

Figure 4.5: *Shape of the scalar flux when using a Monte Carlo (ϕ_{MC}) and a discrete ordinates (ϕ_{SN}) approach for different discretisations.*

The plots suggest that for small M and N, the two methods give noticeably different results, but that the two curves get closer together when the mesh is refined. Note that we used for the Monte Carlo method $n/(2MN) = 32768$ particles per mesh element to produce the graphs. We now show that the accuracy of the Monte Carlo solution depends strongly on the number of particles that are used. The plots in Figure 4.6 indicate that for small numbers of particles per mesh element the results contain a large amount of uncertainty.



(a) $M = 128$, $N = 64$, $n = 2^{15}$

(b) $M = 128$, $N = 64$, $n = 2^{19}$

(c) $M = 128$, $N = 64$, $n = 2^{23}$

(d) $M = 128$, $N = 64$, $n = 2^{27}$

Figure 4.6: *Shape of the scalar flux when using a Monte Carlo (ϕ_{MC}) and a discrete ordinates (ϕ_{SN}) approach for different numbers of particles.*

Table 4.7 emphasises that the results improve for growing M, N and n. The entries in the table represent

$$\int_0^L |\phi_{\mathrm{MC}}(z) - \phi_{\mathrm{SN}}(z)| \, \mathrm{d}z$$

for different mesh refinement numbers M and N and varying numbers of particles n.

Comparing the values for a fixed M and N, say $M = 512$ and $N = 256$ with increasing

M	N	$n = 2^{13}$	2^{15}	2^{17}	2^{19}	2^{21}	2^{23}	2^{25}	2^{27}
8	4	0.4537	0.4878	0.4538	0.4619	0.4669	0.4683	0.4643	0.4650
16	8	0.3325	0.2801	0.2577	0.2536	0.2607	0.2626	0.2624	0.2615
32	16	0.5989	0.2247	0.1989	0.1596	0.1402	0.1415	0.1429	0.1416
64	32	0.7371	0.3354	0.1656	0.1061	0.0817	0.0722	0.0752	0.0736
128	64	1.1173	0.5309	0.2654	0.1469	0.0753	0.0455	0.0388	0.0375
256	128	1.5672	0.7232	0.3966	0.1786	0.0915	0.0482	0.0262	0.0221
512	256	2.1006	1.0703	0.5031	0.2653	0.1252	0.0652	0.0334	0.0184
1024	512	2.8476	1.4301	0.7621	0.3627	0.1858	0.0938	0.0478	0.0237

Table 4.7: *Size of the integral between the scalar fluxes obtained using a discrete ordinates and a Monte Carlo approach for different mesh refinements and different numbers of particles.*

n, we see that the difference between the two results roughly halves when we quadruple the number of particles. This agrees with the result (4.9) for the "convergence" of Monte Carlo, which says that the expected error should decrease with order $n^{-\frac{1}{2}}$.

On the other hand, for smaller mesh numbers such as $M = 32$ and $N = 16$, the terms decrease initially but then stagnate at about 0.14, even if we keep increasing the batch size. This is due to the discretisation error in ϕ_{SN}.

Looking at the final column in the table we note that the difference between the two approaches roughly halves when doubling the number of spatial and angular points. The increase from the penultimate to the final row is most likely due to the larger uncertainty in the Monte Carlo solution resulting from not starting enough particles to reduce the variance sufficiently.

Table 4.8 shows that the computing time of the Monte Carlo method grows linearly with the number of particles. If we want to double the number of spatial and angular intervals, we need to take four times as many neutrons in our Monte Carlo implementation to keep the variance of the solution at the same level. This leads to a growth of the computing time of roughly four.

On the other hand, if we apply the same mesh refinement in our discrete ordinates approach, the growth in computing time for the solution of the linear system is bigger than a factor of four. In our example we used GMRES with an LU preconditioner ($t^{(i)}_{\mathrm{SN1}}$) which gave a growth factor of about 16 for larger problems. Using GMRES without a preconditioner ($t^{(i)}_{\mathrm{SN2}}$) but restarts after 1000 iterations, lead for some i to a smaller growth factor but overall to longer run times. These results suggest that for very high levels of detail the Monte Carlo solution will be computationally more efficient than a discrete ordinates approach.

i	M	N	n	$t^{(i)}_{\texttt{MC}}$	$t^{(i)}_{\texttt{SN1}}$	$t^{(i)}_{\texttt{SN2}}$	$\frac{t^{(i)}_{\texttt{MC}}}{t^{(i-1)}_{\texttt{MC}}}$	$\frac{t^{(i)}_{\texttt{SN1}}}{t^{(i-1)}_{\texttt{SN1}}}$	$\frac{t^{(i)}_{\texttt{SN2}}}{t^{(i-1)}_{\texttt{SN2}}}$
1	8	4	2^{13}	0.7	0.1	0.1			
2	16	8	2^{15}	2.3	0.0	0.3	3.36	0.09	2.76
3	32	16	2^{17}	9.1	0.0	3.1	3.93	2.70	12.06
4	64	32	2^{19}	36.8	0.2	99.8	4.06	15.43	31.93
5	128	64	2^{21}	148.5	3.6	1915.9	4.03	15.66	19.20
6	256	128	2^{23}	604.8	55.1	20822.8	4.07	15.20	10.87
7	512	256	2^{25}	2512.4	875.9	231638.8	4.15	15.90	11.12
8	1024	512	2^{27}	10691.4	13879.9		4.26	15.85	

Table 4.8: *Comparison of the computing times needed (in seconds) to solve the source problem $\mathcal{T}\Psi = Q$ for increasing levels of detail.*

The next results in Figure 4.9 are for different isotropic sources. The first source Q_1 is the Heaviside function

$$Q_1(z,\mu) \;=\; \begin{cases} 1/2 & \text{if } z \,\le\, L/2 \\ 0 & \text{if } z \,>\, L/2 \quad . \end{cases}$$

The second test case considers an approximation of the δ-function to represent a point source at the centre of the reactor with $\int_0^L \int_{-1}^1 Q_2(z,\mu)\,\mathrm{d}\mu\,\mathrm{d}z = 1$, given by

$$Q_2(z,\mu) \;=\; \begin{cases} M/2 & \text{if } L/2 - 1/(2M) \,\le\, z \,\le\, L/2 + 1/(2M) \\ 0 & \text{otherwise .} \end{cases}$$

We also test a source that contains negative components denoted by

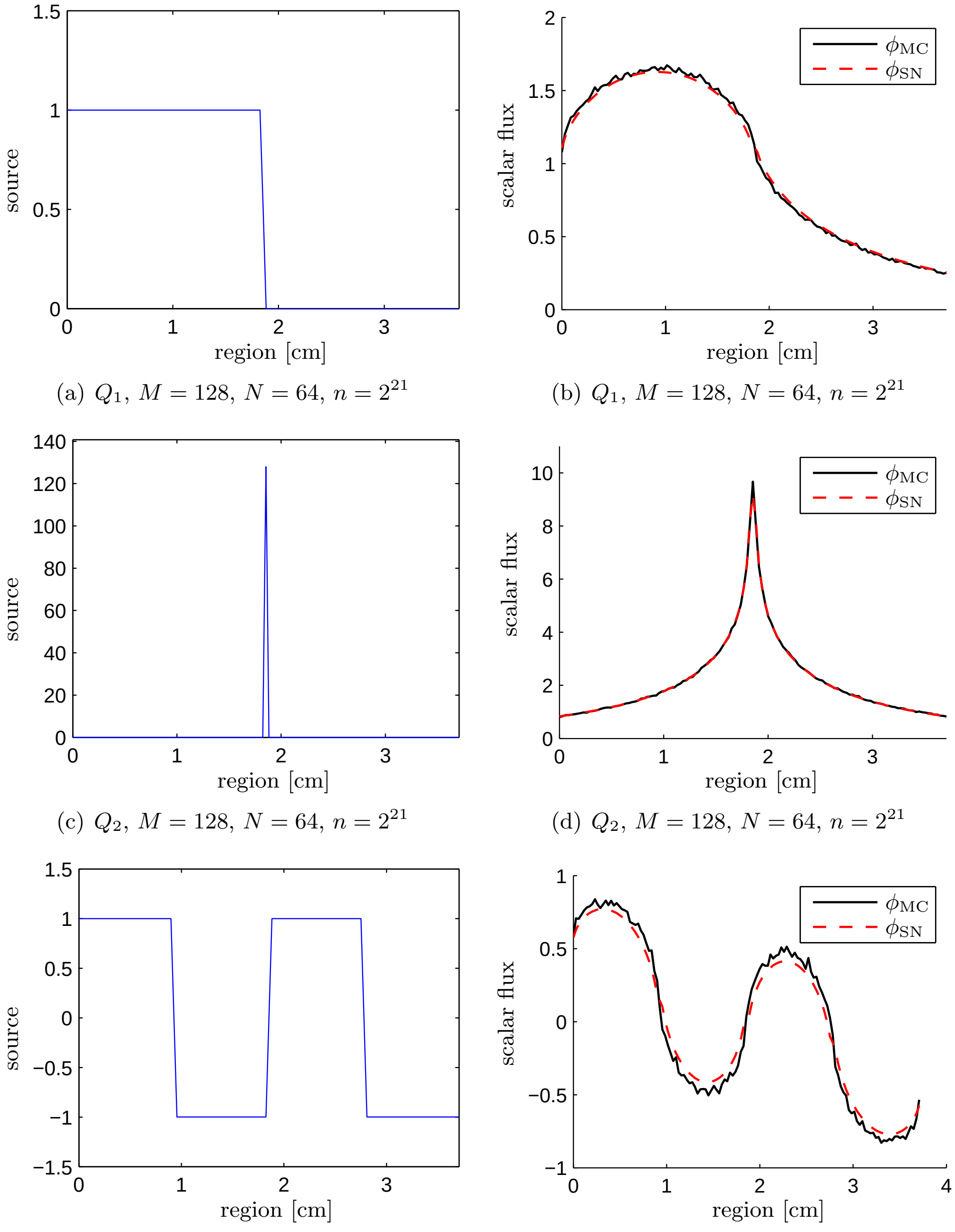
$$Q_3(z,\mu) \;=\; \begin{cases} 1/2 & \text{if } 0 \,\le\, z \,\le\, L/4 \\ -1/2 & \text{if } L/4 \,<\, z \,\le\, L/2 \\ 1/2 & \text{if } L/2 \,<\, z \,\le\, 3L/4 \\ -1/2 & \text{if } 3L/4 \,<\, z \,\le\, L \quad . \end{cases}$$

The very good results for the point source Q_2 are due to all neutrons starting from the interval around $L/2$. Similarly, the results for Q_1 are better than for Q_3 since in the first case particles are only started from the left half of the slab while the same number of neutrons is distributed over the full slab in the final example.

4.5.2 Convergence of source iteration

We will now consider the convergence properties of the source iteration defined in (4.24) for problems of the form

$$(\mathcal{T} - \mathcal{S})\Psi \;=\; Q\ .$$



(a) Q_1, $M = 128$, $N = 64$, $n = 2^{21}$

(b) Q_1, $M = 128$, $N = 64$, $n = 2^{21}$

(c) Q_2, $M = 128$, $N = 64$, $n = 2^{21}$

(d) Q_2, $M = 128$, $N = 64$, $n = 2^{21}$

(e) Q_3, $M = 128$, $N = 64$, $n = 2^{21}$

(f) Q_3, $M = 128$, $N = 64$, $n = 2^{21}$

Figure 4.9: *Spatial shapes* $\int_{-1}^{1} Q_i(z,\mu)\,\mathrm{d}\mu$ *for the sources* Q_i *(left) and the scalar fluxes (right) obtained from the solution to the source problems* $T\Psi = Q_i$ *when using a Monte Carlo* (ϕ_{MC}) *and a discrete ordinates* (ϕ_{SN}) *approach.*

In Lemma 4.4 we showed that

$$\|\phi - \phi^{(i+1)}\|_{L^2(V)} \;\leq\; \frac{\sigma_s}{\sigma}\, \|\phi - \phi^{(i)}\|_{L^2(V)} \;.$$

This suggests that the scattering ratio σ_s/σ might influence the convergence of the iterative scheme. We now provide numerical results for different scattering ratios when using the discrete ordinates scheme from Section 3.3.3 to arrive at a matrix form of the source iteration, i.e.

$$T\mathbf{\Psi}^{(i+1)} \;=\; \mathbf{Q} + S\mathbf{\Psi}^{(i)} \;. \tag{4.44}$$

As solver for the linear problems we use again the GMRES function in Matlab with an LU factorisation as preconditioner and demand an inner tolerance of 10^{-14}. We stop the source iteration when the norm of the residual $\mathbf{res}^{(i)} = (T - S)\mathbf{\Psi}^{(i)} - \mathbf{Q}$ is less than 10^{-10}. As starting guess we use $\mathbf{\Psi}^{(0)} = \mathbf{0}$.

Table 4.10 shows results for different ratios σ_s/σ. The column i denotes the iteration number in which the norm $\|\mathbf{res}^{(i)}\|$, stated in the last column, was obtained. The entries r_∞ and r_2 are the mean values of the ratios

$$\frac{\|P\mathbf{\Psi} - P\mathbf{\Psi}^{(i+1)}\|_\infty}{\|P\mathbf{\Psi} - P\mathbf{\Psi}^{(i)}\|_\infty} \quad \text{and} \quad \frac{\|P\mathbf{\Psi} - P\mathbf{\Psi}^{(i+1)}\|_2}{\|P\mathbf{\Psi} - P\mathbf{\Psi}^{(i)}\|_2}\;,$$

where P is the discrete version of the projection operator and $\mathbf{\Psi}$ is the solution of the source problem.

M	N	σ_s/σ	i	r_∞	r_2	$\|\mathtt{res}^{(i)}\|$
128	64	0.0100	7	0.0067	0.0067	1.65E-11
128	64	0.1000	12	0.0666	0.0665	2.07E-11
128	64	0.5000	27	0.3337	0.3336	7.07E-11
128	64	0.9000	57	0.6002	0.6001	6.74E-11
128	64	0.9500	63	0.6335	0.6334	9.09E-11
128	64	0.9900	69	0.6601	0.6601	9.58E-11
128	64	0.9990	71	0.6662	0.6661	7.93E-11
128	64	1.0000	71	0.6669	0.6668	8.51E-11
128	64	1.1000	92	0.7334	0.7334	9.95E-11
128	64	2.0000	1000	1.3333	1.3333	1.53E+127

Table 4.10: *Convergence results of the source iteration* (4.44) *for different ratios* σ_s/σ.

The results in Table 4.10 show that the rate of decrease in the norm of the error from one iteration to the next, indicated by r_∞ and r_2, does depend on the ratio σ_s/σ. This leads for the (nonphysical) case $\sigma_s = 2\sigma$ to a failure of the source iteration to converge (see the final row of the table).

We now briefly indicate the convergence difficulties when $\sigma \to \infty$ and $\sigma_s/\sigma \to 1$.

This situation is known as the diffusion limit as (4.20) becomes singular and its solution tends to a solution of a diffusion equation [73, 82]. The physical equivalent is a problem with a strong scatterer and small capture probability. We simulated the problem by multiplying σ and σ_s each with the scaling factor β, keeping the ratios σ_s/σ as in Table 4.10. Table 4.11 shows the results for a fixed value of $\beta = 100$.

M	N	β	σ_s/σ	i	r_∞	r_2	$\|\mathtt{res}^{(i)}\|$
128	64	100	0.0100	8	0.1469	0.0268	1.95E-12
128	64	100	0.1000	14	0.1663	0.0971	1.72E-11
128	64	100	0.5000	42	0.5045	0.4972	7.86E-11
128	64	100	0.9000	268	0.8997	0.8992	9.82E-11
128	64	100	0.9500	548	0.9499	0.9495	9.78E-11
128	64	100	0.9900	1000	0.9900	0.9897	5.69E-03
128	64	100	0.9990	1000	0.9989	0.9988	4.85E+01
128	64	100	1.0000	1000	0.9998	0.9998	1.32E+02
128	64	100	1.1000	1000	1.0993	1.0995	3.26E+43
128	64	100	2.0000	1000	1.9989	1.9991	1.41E+303

Table 4.11: *Convergence results of the source iteration* (4.44) *for different ratios* σ_s/σ *and a scaling factor* $\beta = 100$ *to model the difficulties for problems with strong scatterers.*

When $\sigma_s/\sigma \to 1$ the values for r_∞ and r_2 in the table get close to the scattering ratio σ_s/σ and lead to very slow convergence as indicated by the theory. A popular acceleration method to overcome this problem is diffusion synthetic acceleration (DSA) which uses a diffusion equation to compute a correction term to the source iteration solution (see [3, 7, 19] and references therein). In [34] it is shown that diffusion synthetic acceleration for the scalar flux problem is equivalent to applying a preconditioner based on the Green's function of a diffusion operator to the Neumann series solution of (4.25).

4.5.3 Numerical results for criticality computations

We now consider numerical results when the Monte Carlo method is used to estimate the criticality as described in Section 4.4. We use again the Los Alamos model problem number 2.

The first example considers a batch size of $n = 1000$ while the number of settling and scoring stages are set to $s_s = 20$ and $s_c = 200$ respectively. The left plot of Figure 4.12 shows the estimated $\lambda_i = 1/k_i$ for each of the 220 batches. The dashed vertical line denotes the end of the 20 settling and the begin of the 200 scoring stages while the horizontal dashed line denotes the true eigenvalue $\lambda^* = 1$ of the problem.

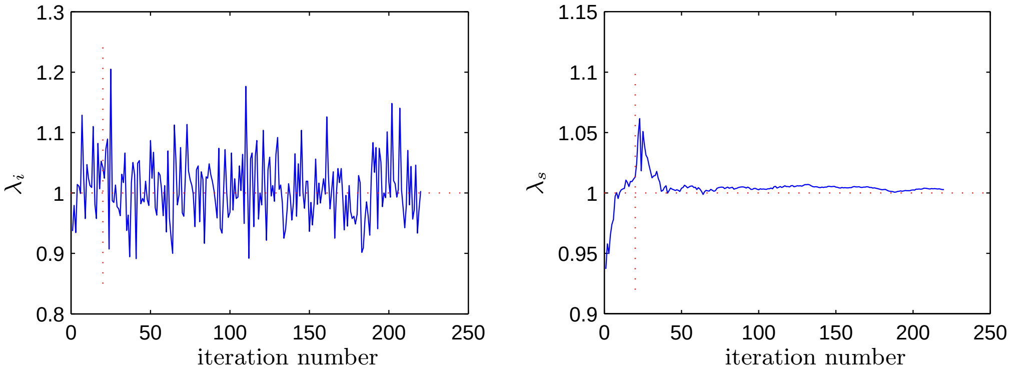


Figure 4.12: *Results for the individual λ_i and cumulative λ_s criticality estimates for $n = 1000$, $s_s = 20$ settling and $s_c = 200$ scoring stages.*

The right plot contains the cumulative criticality value $\lambda_s = 1/k_s$ during the settling and scoring stages, computed from

$$k_s \;=\; \frac{1}{s}\sum_{i=1}^{s} k_i \; .$$

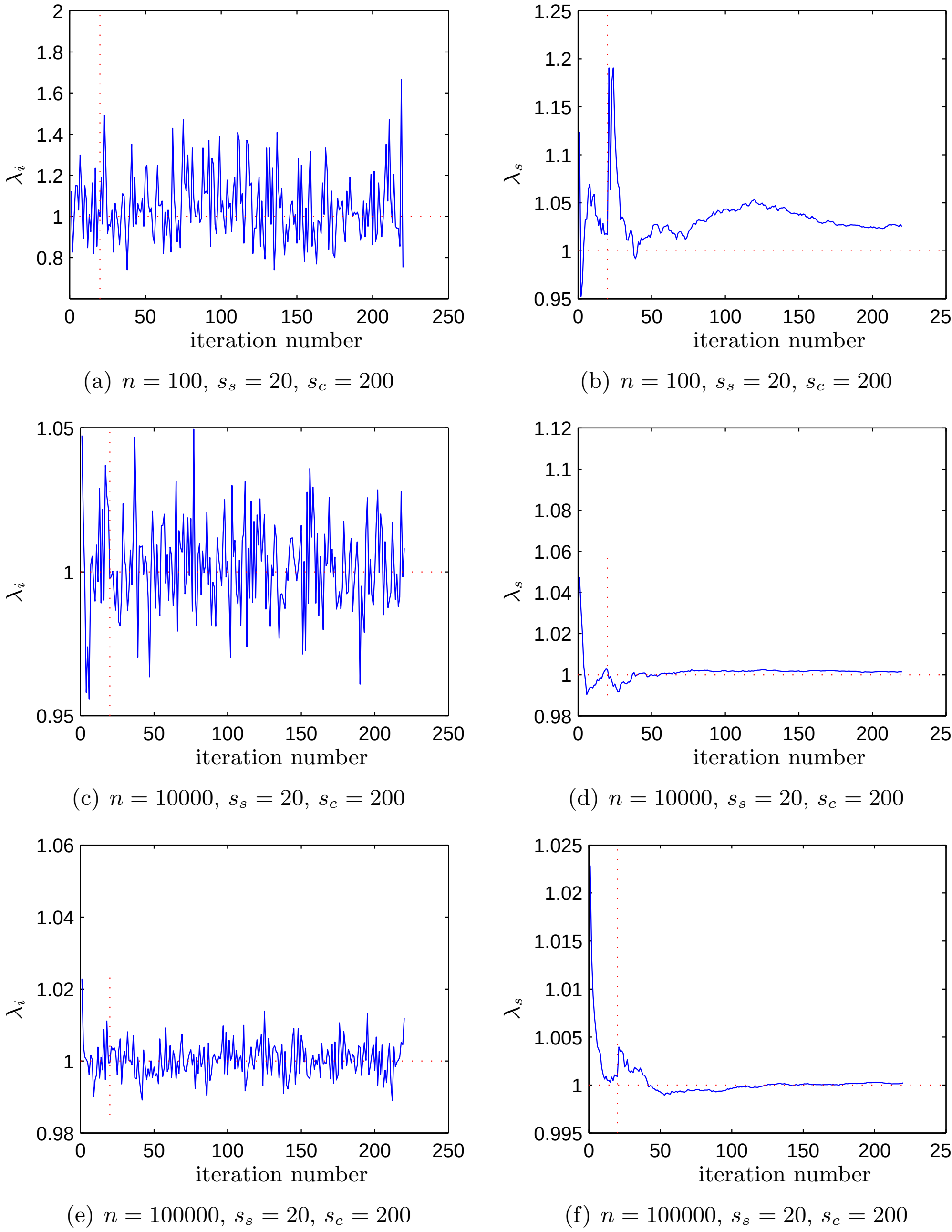
The individual λ_i vary around the true λ^* and the cumulative λ_s seems to settle. The final estimate is $\lambda = 1.0029$ and the estimate for the standard deviation from (4.8) gives $S \approx 1.79$. Analogously to (4.9) (and ignoring any bias), we can use these values to construct a 99.7% confidence interval of $[1.0029 - 0.0121, 1.0029 + 0.0121]$. This means that, in the long run, 99.7% of intervals constructed in this way will contain the true value λ^*. We shall denote this in the future as $\lambda_{\mathrm{MC}} = 1.0029 \pm 0.0121$.

We now vary the number of particles n in each batch to 100, 10000 and 100000. Figure 4.13 contains the plots of the results.

As expected the estimates improve for larger batch sizes (note the scaling of the y-axis). The final approximations λ_{MC} for different values of n, together with the dimensions of the corresponding confidence intervals, are given in Table 4.14.

n	s_s	s_c	λ_{MC}	$\lvert\lambda^* - \lambda_{\mathrm{MC}}\rvert$	S	$\frac{3S}{\sqrt{n\cdot s_c}}$
100	20	200	1.025326	0.025326	1.80	0.038119
1000	20	200	0.999196	0.000804	1.79	0.012026
10000	20	200	1.000155	0.000155	1.79	0.003793
100000	20	200	1.000201	0.000201	1.78	0.001194
1000000	20	200	0.999983	0.000017	1.79	0.000380

Table 4.14: *Criticality estimates from Monte Carlo calculations for different batch sizes n.*



(a) $n = 100$, $s_s = 20$, $s_c = 200$

(b) $n = 100$, $s_s = 20$, $s_c = 200$

(c) $n = 10000$, $s_s = 20$, $s_c = 200$

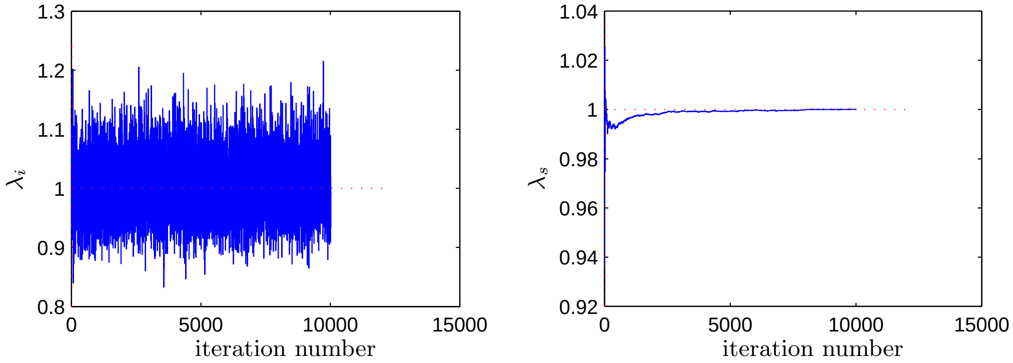
(d) $n = 10000$, $s_s = 20$, $s_c = 200$

(e) $n = 100000$, $s_s = 20$, $s_c = 200$

(f) $n = 100000$, $s_s = 20$, $s_c = 200$

Figure 4.13: *Results for the individual λ_i and cumulative λ_s criticality estimates using different batch sizes.*

Instead of increasing the batch size, we can also increase the number of scoring stages to reduce the uncertainty on the result. Figure 4.15 considers the case $n = 1000$, $s_s = 20$, $s_c = 10000$. The final result is $\lambda_{\mathrm{MC}} = 1.000049 \pm 0.0017$.



(a) $n = 1000$, $s_s = 20$, $s_c = 10000$

(b) $n = 1000$, $s_s = 20$, $s_c = 10000$

Figure 4.15: *With growing number of scoring stages the eigenvalue estimate λ_s becomes more accurate.*

Table 4.16 contains criticality estimates from Monte Carlo calculations and includes the computing times needed. The table focusses on the case when the number of particles in each batch, as well as the number of scoring stages, is doubled from one row to the next.

n	s_s	s_c	$\lambda_{\mathtt{MC}}$	$\lvert\lambda^* - \lambda_{\mathtt{MC}}\rvert$	S	$\frac{3S}{\sqrt{n \cdot s_c}}$	$t_{\mathtt{MC}}^{(i)}$	$\frac{t_{\mathtt{MC}}^{(i)}}{t_{\mathtt{MC}}^{(i-1)}}$
20	100	20	0.930233	0.069767	1.78	0.266722	0.5	
40	100	40	1.040312	0.040312	1.81	0.136114	0.6	1.3
80	100	80	1.005657	0.005657	1.85	0.069466	1.5	2.4
160	100	160	0.986095	0.013905	1.78	0.033292	4.0	2.7
320	100	320	1.006279	0.006279	1.78	0.016706	12.7	3.1
640	100	640	1.000836	0.000836	1.78	0.008364	42.9	3.4
1280	100	1280	0.999427	0.000573	1.78	0.004178	160.6	3.7
2560	100	2560	0.999114	0.000886	1.78	0.002091	626.9	3.9
5120	100	5120	0.999787	0.000213	1.79	0.001046	2446.2	3.9
10240	100	10240	1.000167	0.000167	1.79	0.000523	10232.0	4.2
20480	100	20480	1.000076	0.000076	1.79	0.000262	42820.4	4.2

Table 4.16: *Criticality estimates from Monte Carlo calculations for different batch sizes n and increasing numbers of scoring stages s_c.*

The size of the estimate $S/\sqrt{n \cdot s_c}$, which determines the width of the confidence interval, halves, as expected, with every test, but the actual error $|\lambda^* - \lambda_{\mathrm{MC}}|$ does not follow such a strict reduction. It even increases twice when doubling n and s_c compared to the previous test. This is typical for Monte Carlo calculations and is a major difference to working with deterministic methods. If we wanted to observe a more reliable decrease in the actual error, we would need to repeat the tests several times and take the average of the results.

We will now finish this chapter by comparing the computing time and obtained accuracy of the Monte Carlo method to the discrete ordinates schemes that we introduced in Section 2.4.1 and 2.4.2. For the deterministic methods the resulting matrix eigenvalue problem is solved by inexact inverse iteration using the special Rayleigh quotient $\widetilde{\rho}$. The inner and outer tolerances were set to 10^{-10} and 10^{-14} respectively. To solve the arising linear systems we used again GMRES with an LU factorisation of T as preconditioner.

Table 4.17 contains the observed error in the results of the discrete ordinates methods with respect to the number of spatial and angular intervals for the Los Alamos test problem, as well as the computing time that was needed to solve the eigenvalue problem.

M	N	$\lvert\lambda^* - \lambda_{\mathtt{SN}}^{\mathtt{SE}}\rvert$	$t_{\mathtt{SE}}^{(i)}$	$\frac{t_{\mathtt{SE}}^{(i)}}{t_{\mathtt{SE}}^{(i-1)}}$	$\lvert\lambda^* - \lambda_{\mathtt{SN}}^{\mathtt{CN}}\rvert$	$t_{\mathtt{CN}}^{(i)}$	$\frac{t_{\mathtt{CN}}^{(i)}}{t_{\mathtt{CN}}^{(i-1)}}$
4	2	0.124067	0.1		0.070584	0.3	
8	4	0.068751	0.1	1.0	0.011292	0.1	0.2
16	8	0.034248	0.1	1.4	0.002248	0.1	1.4
32	16	0.017137	0.2	1.9	0.000537	0.2	2.0
64	32	0.008588	0.7	3.2	0.000133	0.7	3.9
128	64	0.004301	4.7	6.8	0.000034	4.6	7.0
256	128	0.002152	56.3	12.0	0.000010	55.5	12.0
512	256	0.001076	819.5	14.5	0.000004	858.4	15.5
1024	512	0.000537	12291.0	15.0	0.000002	12376.7	14.4

Table 4.17: *Comparison of the discretisation error of the symmetry preserving Euler scheme λ_{SN}^{SE} and the Crank-Nicolson scheme λ_{SN}^{CN} for the Los Alamos problem.*

The table shows that the discretisation error halves for the symmetry preserving Euler scheme when the number of spatial and angular mesh points are doubled, i.e. that we obtain linear convergence for this scheme. For the Crank-Nicolson approach the convergence of the discretisation error is quadratic which agrees with the discretisation error estimate in Theorem 2.38.

Let us now compare the error of the deterministic methods for the finest mesh $M = 1024$, $N = 512$ with the Monte Carlo result corresponding to a similar amount of computing time ($n = s_s = 10240$). We observe that the Crank-Nicolson scheme yields the most accurate result, while the actual errors in the Monte Carlo and symmetry preserving Euler methods are of similar sizes. Comparing the growth in time in Tables 4.16 and 4.17, we note that the times for the Monte Carlo computations are multiplied by four from one row to the next. The growth for the deterministic approaches lies between 12 and 16 for large systems. We observed the same situation in Table 4.8 for the solution of a source problem, where the growth factor appeared to approximate 16.

Therefore, we expect that in order to obtain very accurate results the Monte Carlo method will be more efficient than using the symmetry preserving Euler scheme. In addition, the Monte Carlo computations can be performed easily and efficiently in parallel while this is a more challenging task for the deterministic methods.

Note that, while we get from a deterministic calculation the same result every time we run the problem, the Monte Carlo method gives us an expected value and a confidence interval which will include the true value with a certain probability (99.7% in our case). Even if we reduced the width of the interval to an acceptable size for our problem, there is a 0.3% chance that the true value will lie outside this range. Hence to be sure that the obtained estimate for the criticality value is accurate, we would need to repeat the experiment several times. Therefore, getting accurate results from Monte Carlo computations is computationally very expensive. As we will see in the next Chapter, this property of Monte Carlo methods can lead to difficulties if we are interested in the difference in the criticality between two similar problems.

Chapter 5

Method of perturbation

In this chapter we discuss a new iterative method to compute the criticality of a nuclear reactor. The idea is to compute the *change* in the eigenvalue and corresponding eigenfunction as a perturbation from a known eigenpair of a previously solved problem, rather than solving directly for the eigenpair of the new problem.

First and second order perturbation theory has been applied to neutron transport problems for many years (e.g. [54]) and an extensive review and summary on this topic is given in [49]. The perturbation approach considered here, the so-called "*method of perturbation*", differs from standard perturbation techniques in the sense that we work with exact rearrangements of the governing equations and do not drop higher order terms. For this reason there is no *a priori* requirement that the perturbation should be "small". The resulting new iterative method, which was developed at Serco Technical and Assurance Services, has been presented briefly in [59].

We motivate and describe the method of perturbation in the first section before relating it in the following section to standard eigenvalue methods from Section 3.1. By applying a similar convergence analysis as in Section 3.2 to a variation of shifted inverse iteration, and using its relation to the method of perturbation, convergence results for the new method are obtained. These show how guaranteed convergence of the method of perturbation does depend (despite using exact equations) on the size of the perturbation as well as the accuracy of the solutions for the inner problems. We will show that even for exact solves convergence is only guaranteed if the perturbations are not too large.

We then describe a variation of the method that is used in practice, before giving numerical results to support the theory. We suggest, motivated by the analysis, an adjustment to the method of perturbation, which resulted in practice in less restrictive demands on the accuracy of the inner solves for the numerical examples considered. We finish the chapter with some remarks on efficient implementations of the inner solver.

5.1 Motivation of the method of perturbation

The main motivation for the following approach results from the high computational costs when Monte Carlo methods are used to solve two similar criticality problems. Nuclear engineers are frequently interested in the change of criticality (i.e. the change in the principal eigenvalue λ) due to modifications of a certain reactor design. This usually requires computing solutions to many perturbed problems.

One popular example for such a situation is the testing of different material compositions in a part of the reactor. The control rod problem from Section 3.3.2 may be considered as such a case: When the absorber rods are lowered into the water channels, the resulting variation in the material composition leads to a change in the criticality and the shape of the flux distribution. The nuclear engineer wants to be able to predict these changes for different insertion depths.

As discussed in Section 4.4, when using the Monte Carlo method to compute the eigenvalue, the engineer obtains a confidence interval for the true λ. To solve the (unperturbed) base problem and the perturbed problem independently, and to then compute a meaningful difference between the confidence intervals for the eigenvalues, demands large sample sizes and therefore long computing times. The idea of the method of perturbation is to solve "explicitly" for the change in the eigenvalue, i.e. without computing the eigenpair of the base problem and the perturbed problems individually and then taking the difference. We now describe the method in more detail.

Let us assume that we know the principal eigenpair (λ_0, Ψ_0), with real and strictly positive λ_0 and Ψ_0, for the base problem

$$(\mathcal{T}_0 - \mathcal{S}_0)\Psi_0 \;=\; \lambda_0\,\mathcal{F}_0\Psi_0 \;,$$

subject to suitable boundary conditions. Then equivalently,

$$(\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)\Psi_0 \;=\; 0 \;. \tag{5.1}$$

We proved in Corollary 2.35 that for the model problems in Section 2.1 such an eigenpair exists and that this eigenpair is also simple. The more general case is covered in [86].

We can now consider any other eigenvalue problem

$$(\mathcal{T} - \mathcal{S} - \lambda\,\mathcal{F})\Psi \;=\; 0 \tag{5.2}$$

as a perturbation of the base problem. Denoting the (known) changes in the operators by $\Delta\mathcal{T}$, $\Delta\mathcal{S}$ and $\Delta\mathcal{F}$, the operators of the perturbed problem (5.2) can be expressed as

$$\begin{aligned} \mathcal{T} &= \mathcal{T}_0 + \Delta\mathcal{T}\ , && (5.3)\\ \mathcal{S} &= \mathcal{S}_0 + \Delta\mathcal{S}\ , \quad \text{and} && (5.4)\\ \mathcal{F} &= \mathcal{F}_0 + \Delta\mathcal{F}\ . && (5.5) \end{aligned}$$

Let us write the principal eigenpair (λ, Ψ) of the perturbed problem as

$$\begin{aligned} \lambda &= \lambda_0 + \Delta\lambda\ , && (5.6)\\ \Psi &= \Psi_0 + \Delta\Psi\ , && (5.7) \end{aligned}$$

where $\Delta\lambda$ and $\Delta\Psi$ are to be found.

Furthermore, we assume that we also know the principal eigenpair (λ_0^*, Ψ_0^*), with real and positive λ_0^* and Ψ_0^*, of the adjoint problem to (5.1), i.e. that we have

$$(\mathcal{T}_0^* - \mathcal{S}_0^* - \lambda_0^*\,\mathcal{F}_0^*)\Psi_0^* \;=\; 0\ . \tag{5.8}$$

First note that $\lambda_0 = \lambda_0^*$ (see also [80, p. 53]). To obtain this result, we multiply (5.1) with Ψ_0^*, and then integrate over all independent variables, which we denote by the inner product $\langle\,\cdot\,,\,\cdot\,\rangle$. In addition, we take the inner product of (5.8) with Ψ_0, and then subtract this from the former equation to obtain

$$\begin{aligned} 0 &= \langle\Psi_0^*, (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)\Psi_0\rangle - \langle\Psi_0, (\mathcal{T}_0^* - \mathcal{S}_0^* - \lambda_0^*\,\mathcal{F}_0^*)\Psi_0^*\rangle\\ &= \langle\Psi_0^*, (\mathcal{T}_0 - \mathcal{S}_0)\Psi_0\rangle - \langle\Psi_0, (\mathcal{T}_0^* - \mathcal{S}_0^*)\Psi_0^*\rangle - \lambda_0\,\langle\Psi_0^*, \mathcal{F}_0\Psi_0\rangle + \lambda_0^*\,\langle\Psi_0, \mathcal{F}_0^*\Psi_0^*\rangle\\ &= (\lambda_0^* - \lambda_0)\langle\Psi_0^*, \mathcal{F}_0\Psi_0\rangle\ . \end{aligned}$$

However, as Ψ_0 and Ψ_0^* are real and strictly positive, the inner product on the right is positive for reactors containing fissile material, so that the eigenvalues of the two problems must agree.

We now use (5.1) to (5.8) to generate two equations for the unknowns $\Delta\lambda$ and $\Delta\Psi$.

Proposition 5.1. *Assuming* $\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi)\rangle \neq 0$ *, then*

$$\begin{aligned}\big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda)\mathcal{F}\big)\Delta\Psi &= -(\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 + \Delta\lambda\mathcal{F}\Psi_0 \ , \ \textit{and} \qquad (5.9)\\ \Delta\lambda &= \frac{\langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})(\Psi_0 + \Delta\Psi)\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi)\rangle} \ . \qquad (5.10)\end{aligned}$$

Proof. From (5.2), and using (5.3) to (5.7), we obtain

$$\begin{aligned}0 &= (\mathcal{T} - \mathcal{S} - \lambda\,\mathcal{F})(\Psi_0 + \Delta\Psi)\\ &= (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\mathcal{F}_0)\Psi_0 + (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 - \Delta\lambda\mathcal{F}\Psi_0 + (\mathcal{T} - \mathcal{S} - \lambda\mathcal{F})\Delta\Psi \ .\end{aligned}$$

Now by applying (5.1) and rearranging the resulting equation, we obtain (5.9). Equation (5.10) for $\Delta\lambda$ is obtained as follows. From (5.2), and using again (5.1), we get

$$\begin{aligned}0 &= \big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda)\mathcal{F}\big)\Psi\\ &= (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)(\Psi_0 + \Delta\Psi) + (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})\Psi - \Delta\lambda\mathcal{F}\Psi\\ &= (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)\Delta\Psi + (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})\Psi - \Delta\lambda\mathcal{F}\Psi \ .\end{aligned}$$

Moving the last term to the left-hand side and taking the inner product with the adjoint solution Ψ_0^* then gives

$$\begin{aligned}\Delta\lambda\langle \Psi_0^*, \mathcal{F}\Psi\rangle &= \langle \Psi_0^*, (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)\Delta\Psi\rangle + \langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})\Psi\rangle\\ &= \langle (\mathcal{T}_0^* - \mathcal{S}_0^* - \lambda_0^*\,\mathcal{F}_0^*)\Psi_0^*, \Delta\Psi\rangle + \langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})\Psi\rangle \ .\end{aligned}$$

Hence, by using (5.8) and dividing by $\langle \Psi_0^*, \mathcal{F}\Psi\rangle \neq 0$, we obtain (5.10). □

Note that no higher order terms were dropped in the derivation of Proposition 5.1, and that (5.9) and (5.10) are exact.

We can now use (5.9) and (5.10) to construct iterative schemes. Starting with a guess for $\Delta\Psi^{(0)}$ a new $\Delta\lambda^{(1)}$ can be computed from (5.10). This can then be inserted for $\Delta\lambda$ on the left-hand side of (5.9), while we use for the $\Delta\lambda$ on the right an initial guess $\Delta\lambda^{(0)}$. Solving the resulting equation for $\Delta\Psi$ gives an update $\Delta\Psi^{(1)}$ which can then be used to obtain a new $\Delta\lambda^{(2)}$ from (5.10), and the iteration can be continued. The resulting method is given in Algorithm 7.

The iteration in this, and all following algorithms in this chapter, is again stopped when the eigenvalue residual

$$\mathrm{res}^{(i)} = (\mathcal{T} - \mathcal{S} - \rho^{(i)}\mathcal{F})\Psi^{(i)}$$

Algorithm 7 Method of perturbation – first version

```
Require: Base problem solutions $\lambda_0$, $\Psi_0$, $\Psi_0^*$, starting guesses $\Delta\Psi^{(0)}$, $\Delta\lambda^{(0)}$.
  for i=0,1,2,... do
    Compute $\Delta\lambda^{(i+1)} = \frac{\langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0 \Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle}$ .
    Obtain $\Delta\Psi^{(i+1)}$ by solving
    $\big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda^{(i+1)})\mathcal{F}\big)\Delta\Psi^{(i+1)} = -(\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 + \Delta\lambda^{(i)}\mathcal{F}\Psi_0$ .
    Define $\widetilde{\Psi}^{(i+1)} = \Psi_0 + \Delta\Psi^{(i+1)}$ .
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$ .
  end for
```

is sufficiently small in some suitable norm, where $\rho^{(i)}$ is the Rayleigh quotient defined in (3.4).

We observed in the proof of Proposition 5.1 that the right-hand side of (5.9) is just a rearrangement of $-\big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda)\mathcal{F}\big)\Psi_0$. It is therefore important that we do not use the same estimate for $\Delta\lambda$ on the left and on the right of equation (5.10), since this would result, for example for $\Delta\lambda^{(i)}$, in a linear system which is equivalent to

$$\big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda^{(i)})\mathcal{F}\big)\Delta\Psi^{(i+1)} = -\big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda^{(i)})\mathcal{F}\big)\Psi_0 \ .$$

The unique solution to this problem is $\Delta\Psi^{(i+1)} = -\Psi_0$ and the next iterate becomes then, in the case of an exact solve, $\Psi^{(i+1)} = 0$. This constraint is similar to the situation for inverse correction in Section 3.1.3, where we avoided using the exact Rayleigh quotient as the shift. We establish in the next section how Algorithm 7 relates to a variation of inverse correction and inverse iteration. Before we do this, let us consider Figure 5.1 which contains a visualisation of the different methods described so far.

The major difference between the method of perturbation and the other iterative schemes lies in the fact that all standard schemes use the *current* iterate as a basis to compute the next estimate, while Algorithm 7 always computes corrections to the *initial* guess Ψ_0 (the base problem solution). The perturbation method solves iteratively for $\Delta\Psi^{(i)}$ and the hope is that $\Delta\Psi^{(i)} \to \Delta\Psi$ such that $\Psi_0 + \Delta\Psi = \Psi$. We now construct variations of the inverse correction and inverse iteration approach which produce, under certain conditions, the same iterates as the method of perturbation. This then allows us to obtain convergence results via a similar analysis as in Section 3.2.

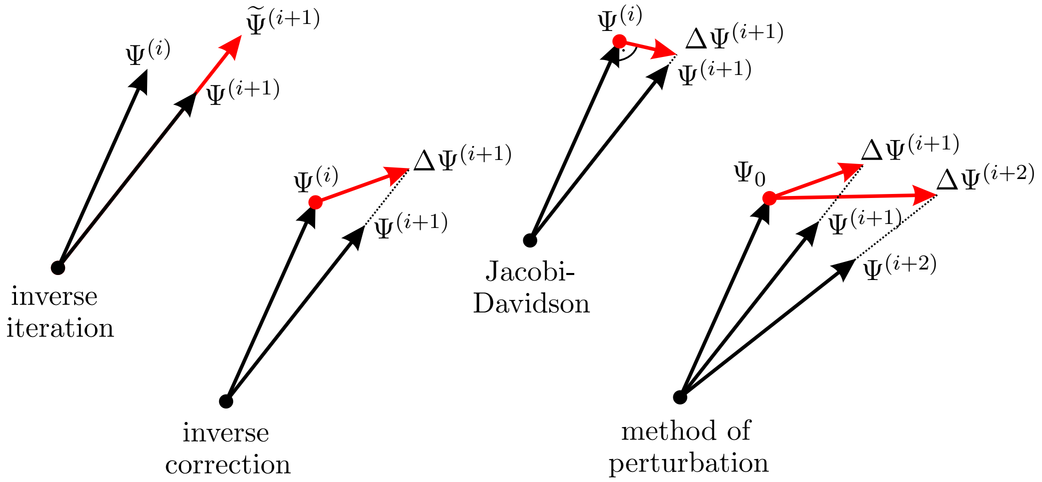


Figure 5.1: *Visualisation of the different iterative methods discussed in Section 3.1 and the method of perturbation from Algorithm 7.*

5.2 Relation to shifted inverse iteration with a fixed right-hand side

In order to explain the relation between the method of perturbation (Algorithm 7) and a form of inverse iteration (Algorithm 2, page 53), we use the following variation of inverse correction (Algorithm 3) as an intermediate step.

Algorithm 8 Inverse correction with a fixed right-hand side

```
Require: Base problem solution $\Psi_0$, shift $\alpha^{(0)}$.
  for i=0,1,2,... do
    Choose a shift $\alpha^{(i+1)}$.
    Obtain $\Delta\Psi^{(i+1)}$ by solving
    $(\mathcal{T}-\mathcal{S}-\alpha^{(i+1)}\mathcal{F})\Delta\Psi^{(i+1)} = -(\mathcal{T}-\mathcal{S}-\alpha^{(i)}\mathcal{F})\Psi_0$.
    Define $\widetilde{\Psi}^{(i+1)} = \Psi_0+\Delta\Psi^{(i+1)}$.
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$.
  end for
```

Lemma 5.2. *Assume $\Delta\Psi^{(0)}$ and $\Delta\lambda^{(0)}$ are as in Algorithm 7. If Algorithm 8 uses the shifts $\alpha^{(0)} := \lambda_0 + \Delta\lambda^{(0)}$ and*

$$\alpha^{(i+1)} := \frac{\langle \Psi_0^*, (\mathcal{T}-\mathcal{S})(\Psi_0+\Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0+\Delta\Psi^{(i)})\rangle}\ , \quad i=0,1,2,\dots\ , \tag{5.11}$$

then Algorithms 7 and 8 produce the same iterates.

Proof. We prove this result by showing that the linear systems in both algorithms agree. Let us first compare the shifts that are used in the two methods. For the claim to hold we need to show that $\alpha^{(i+1)} = \lambda_0 + \Delta\lambda^{(i+1)}$. Starting with the definition of $\alpha^{(i+1)}$ in (5.11), and subtracting and adding λ_0, we get

$$\begin{aligned}\alpha^{(i+1)} &= \frac{\langle \Psi_0^*, (\mathcal{T}-\mathcal{S})(\Psi_0 + \Delta\Psi^{(i)})\rangle \;-\; \lambda_0 \langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle} \;+\; \lambda_0 \\ &= \frac{\langle \Psi_0^*, (\mathcal{T}-\mathcal{S}-\lambda_0\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle} \;+\; \lambda_0 \;.\end{aligned}$$

Hence, we have to show that the first term on the right-hand side is equal to

$$\Delta\lambda^{(i+1)} \;=\; \frac{\langle \Psi_0^*, (\Delta\mathcal{T}-\Delta\mathcal{S}-\lambda_0\Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle} \;.$$

Using (5.8) and (5.1) we get

$$\begin{aligned}0 &= \langle (\mathcal{T}_0^* - \mathcal{S}_0^* - \lambda_0^* \mathcal{F}_0^*)\Psi_0^*, \Delta\Psi^{(i)}\rangle \\ &= \langle \Psi_0^*, (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)\Delta\Psi^{(i)}\rangle + \langle \Psi_0^*, (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)\Psi_0\rangle \\ &= \langle \Psi_0^*, (\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\,\mathcal{F}_0)(\Psi_0 + \Delta\Psi^{(i)})\rangle + \langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle \\ &\qquad - \langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle \\ &= \langle \Psi_0^*, (\mathcal{T} - \mathcal{S} - \lambda_0\,\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle - \langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle \;,\end{aligned}$$

and, under the assumption $\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle \neq 0$, we then obtain

$$\begin{aligned}\frac{\langle \Psi_0^*, (\mathcal{T} - \mathcal{S} - \lambda_0\,\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle} &= \frac{\langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\,\Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle} \\ &= \Delta\lambda^{(i+1)} \;.\end{aligned}$$

Therefore,

$$\alpha^{(i+1)} \;=\; \lambda_0 + \Delta\lambda^{(i+1)} \;,$$

and the shifts agree. Hence, the operators on the left-hand sides of the linear systems are the same. We now compare the right-hand sides. Starting with the right-hand side of the linear system in Algorithm 8, and using again (5.1), we have

$$\begin{aligned}-(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\Psi_0 &= -\big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda^{(i)})\mathcal{F}\big)\Psi_0 \\ &= -(\mathcal{T}_0 - \mathcal{S}_0 - \lambda_0\mathcal{F}_0)\Psi_0 - (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 + \Delta\lambda^{(i)}\mathcal{F}\Psi_0 \\ &= -(\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 + \Delta\lambda^{(i)}\mathcal{F}\Psi_0 \;.\end{aligned}$$

Hence, both algorithms solve the same linear systems and therefore produce the same iterates. □

Next we show that Algorithm 8 is equivalent to inverse iteration in Algorithm 2 with a fixed right-hand side defined as follows.

Algorithm 9 Shifted inverse iteration with a fixed right-hand side

```
Require: Base problem solution $\Psi_0$.
  for i=0,1,2,... do
    Choose a shift $\alpha^{(i+1)}$.
    Compute $\widetilde{\Psi}^{(i+1)}$ such that $(\mathcal{T}-\mathcal{S}-\alpha^{(i+1)}\mathcal{F})\widetilde{\Psi}^{(i+1)} = \mathcal{F}\Psi_0$.
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$.
  end for
```

To avoid confusion we denote the next iterate before normalisation of Algorithm 8 (inverse correction with a fixed right-hand side) by $\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)}$, and denote the result of the linear system in Algorithm 9 by $\widetilde{\Psi}_{\mathrm{II}}^{(i+1)}$. We obtain, similar to Lemma 3.1, the following result.

Lemma 5.3. *If we use in Algorithms 8 and 9 the same shifts $\alpha^{(i)}$, solve the linear systems exactly and apply the same normalisation, then Algorithms 8 and 9 produce, up to a different sign, the same iterates.*

Proof. By the definition of the next iterate of Algorithm 8, we have

$$\begin{aligned}
\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)} &= \Psi_0 + \Delta\Psi^{(i)} \\
&= \Psi_0 - (\mathcal{T}-\mathcal{S}-\alpha^{(i+1)}\mathcal{F})^{-1}(\mathcal{T}-\mathcal{S}-\alpha^{(i)}\mathcal{F})\Psi_0 \\
&= \Psi_0 - (\mathcal{T}-\mathcal{S}-\alpha^{(i+1)}\mathcal{F})^{-1}[(\mathcal{T}-\mathcal{S}-\alpha^{(i+1)}\mathcal{F})\Psi_0 + (\alpha^{(i+1)}-\alpha^{(i)})\mathcal{F}\Psi_0] \\
&= (\alpha^{(i)}-\alpha^{(i+1)})(\mathcal{T}-\mathcal{S}-\alpha^{(i+1)}\mathcal{F})^{-1}\mathcal{F}\Psi_0 \\
&= (\alpha^{(i)}-\alpha^{(i+1)})\widetilde{\Psi}_{\mathrm{II}}^{(i+1)} \,. \qquad (5.12)
\end{aligned}$$

If the same normalisation is used in both methods, we therefore get

$$\Psi_{\mathrm{IC}}^{(i+1)} = \frac{\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)}}{\|\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)}\|} = \frac{\alpha^{(i)}-\alpha^{(i+1)}}{|\alpha^{(i)}-\alpha^{(i+1)}|}\frac{\widetilde{\Psi}_{\mathrm{II}}^{(i+1)}}{\|\widetilde{\Psi}_{\mathrm{II}}^{(i+1)}\|} = \pm\Psi_{\mathrm{II}}^{(i+1)} \,.$$

Hence, Algorithms 8 and 9 with equal shifts produce, up to a different sign, the same iterates if the same normalisation is applied. □

Remark 5.4. *Note that usually the previous iterate is used to compute the new shift $\alpha^{(i+1)}$ via some form of Rayleigh quotient. In this case the different signs in $\Psi^{(i+1)}$ cancel out in the computation of $\alpha^{(i+1)}$ since $\Psi^{(i+1)}$ appears in the numerator and denominator of the Rayleigh quotient. Hence, both algorithms produce the same sequence of eigenpair approximations.*

Remark 5.5. *From* (5.12) *we observe that we might encounter numerical problems when $(\alpha^{(i)} - \alpha^{(i+1)}) \to 0$. On page 150 in Section 5.6 we suggest an adjustment to the shift $\alpha^{(i)}$ on the right-hand side of the linear system in Algorithm 8 in order to prevent these problems.*

Let us now consider the case when the linear systems are not solved exactly, i.e. the solution of the linear system in Algorithm 8 satisfies

$$\|(\mathcal{T} - \mathcal{S} - \alpha^{(i+1)}\mathcal{F})\Delta\Psi^{(i+1)} + (\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\Psi_0\| \;\leq\; \tau^{(i)} \;, \tag{5.13}$$

and for the next iterate in Algorithm 9 we have

$$\|(\mathcal{T} - \mathcal{S} - \alpha^{(i+1)}\mathcal{F})\widetilde{\Psi}_{\text{II}}^{(i+1)} - \mathcal{F}\Psi_0\| \;\leq\; \tau^{(i)} \;. \tag{5.14}$$

Assumption 5.6. *We assume in the following that we use a fixed (iterative) solver for all linear systems and that this solver yields for problems of the form $A\mathbf{x} = \mathbf{b}$, where $A \in \mathbb{R}^{n\times n}$, $\mathbf{b}, \mathbf{x} \in \mathbb{R}^n$, a unique approximate solution $\mathbf{x}(\tau) \in \mathbb{R}^n$ satisfying*

$$\|A\mathbf{x}(\tau) - \mathbf{b}\| \;\leq\; \tau \;, \quad \textit{for all} \quad \tau \geq 0 \;. \tag{5.15}$$

Furthermore, we assume that the approximate solution $\mathbf{y}(\tau) \in \mathbb{R}^n$ of $A\mathbf{y} = -\mathbf{b}$, which satisfies

$$\|A\mathbf{y} + \mathbf{b}\| \;\leq\; \tau \;, \tag{5.16}$$

with the same τ as in (5.15)*, lies in the one-dimensional subspace spanned by the solution $\mathbf{x}(\tau)$ of* (5.15)*, i.e. $\mathbf{y}(\tau) \in \operatorname{span}\{\mathbf{x}(\tau)\} := \{\beta\mathbf{x}(\tau) \;,\; \beta \in \mathbb{R}\}$.*

Remark 5.7. *Assumption 5.6 is, for example, satisfied by Krylov subspace based solvers such as GMRES (see e.g. [100] for further details of Krylov subspace methods). These solvers search for solutions $\mathbf{x}$ to $A\mathbf{x} = \mathbf{b}$ in subspaces that are spanned (in the simplest case) by vectors obtained by repeatedly applying the matrix A to the right-hand side $\mathbf{b}$ (or $-\mathbf{b}$ in the case of* (5.16)*). Hence, for the problems* (5.15) *and* (5.16)*, the same subspaces are constructed and therefore $\mathbf{y}(\tau) \in \operatorname{span}\{\mathbf{x}(\tau)\}$. The uniqueness of the obtained solution follows from the deterministic nature of the methods.*

Using this assumption we can now show the following relation between Algorithms 8 and 9 for inexact solves of the linear systems.

Lemma 5.8. *Let Algorithms 8 and 9 use the same shifts $\alpha^{(i)}$ and let the inner tolerances of Algorithm 9 be given by $\tau^{(i)}$. Assume that the tolerances for the linear system of Algorithm 8 are set to*

$$\widetilde{\tau}^{(i)} \;:=\; |\alpha^{(i)} - \alpha^{(i+1)}|\, \tau^{(i)} \;, \tag{5.17}$$

and that the same normalisation is applied. If, in addition, a linear solver that satisfies Assumption 5.6 is used, then both algorithms produce, up to a different sign, the same iterates.

Proof. After rearranging and applying the adjusted tolerances $\widetilde{\tau}^{(i)}$ from (5.17), the inner problem (5.13) for Algorithm 8 becomes

$$\|(\mathcal{T} - \mathcal{S} - \alpha^{(i+1)}\mathcal{F})\widetilde{\Psi}_{\text{IC}}^{(i+1)} - (\alpha^{(i)} - \alpha^{(i+1)})\mathcal{F}\Psi_0\| \;\leq\; |\alpha^{(i)} - \alpha^{(i+1)}|\, \tau^{(i)} \;,$$

where $\widetilde{\Psi}_{\text{IC}}^{(i+1)} = \Psi_0 + \Delta\Psi^{(i+1)}$. Dividing this equation by $|\alpha^{(i)} - \alpha^{(i+1)}|$ then gives

$$\left\|(\mathcal{T} - \mathcal{S} - \alpha^{(i+1)}\mathcal{F})\frac{1}{|\alpha^{(i)} - \alpha^{(i+1)}|}\widetilde{\Psi}_{\text{IC}}^{(i+1)} \pm \mathcal{F}\Psi_0\right\| \;\leq\; \tau^{(i)} \;. \tag{5.18}$$

If we now compare (5.18) with (5.14) and use that the inner solver satisfies Assumption 5.6, we obtain that $\widetilde{\Psi}_{\text{II}}^{(i+1)} = \pm\frac{\beta}{|\alpha^{(i)}-\alpha^{(i+1)}|}\widetilde{\Psi}_{\text{IC}}^{(i+1)}$ for some $\beta \in \mathbb{R}$. Hence, provided the same normalisation is used for both methods,

$$\Psi_{\text{II}}^{(i+1)} \;=\; \frac{\widetilde{\Psi}_{\text{II}}^{(i+1)}}{\|\widetilde{\Psi}_{\text{II}}^{(i+1)}\|} \;=\; \pm\frac{|\beta|}{\beta}\frac{|\alpha^{(i)} - \alpha^{(i+1)}|}{|\alpha^{(i)} - \alpha^{(i+1)}|}\frac{\widetilde{\Psi}_{\text{IC}}^{(i+1)}}{\|\widetilde{\Psi}_{\text{IC}}^{(i+1)}\|} \;=\; \pm\Psi_{\text{IC}}^{(i+1)} \;.$$

□

Corollary 5.9. *The inexact versions of Algorithms 7 and 9 produce, up to a different sign, the same iterates, provided Algorithm 9 uses the shifts* (5.11)*, the inner tolerances for Algorithm 7 are given by* (5.17)*, the same normalisation is applied, and the inner solver satisfies Assumption 5.6. (The inexact version of Algorithm 9 is stated below as Algorithm 10.)*

Proof. Lemma 5.2 showed that for shifts (5.11) Algorithm 8 agrees with Algorithm 7. Lemma 5.8 now establishes the relation between Algorithm 8 and Algorithm 9 for inexact solves with adjusted tolerances (5.17) which finishes the proof. □

Note that this result also includes the equivalence in the case of exact solves, i.e. $\tau^{(i)} = \widetilde{\tau}^{(i)} = 0$. In the next section we will provide convergence results for the variation of shifted inverse iteration given in Algorithm 9.

5.3 Convergence of inexact shifted inverse iteration with a fixed right-hand side

The analysis for this method is analogous to the convergence analysis in Section 3.2. We consider again the homogeneous monoenergetic model problems with isotropic scatter from Section 2.1, and use the relation to the scalar flux problem given in Corollary 2.5. As before we allow for inexact solves of the linear system and make use of the $*$-norm defined in (3.12). The method that we are analysing is given by the following inexact version of Algorithm 9.

Algorithm 10 Inexact shifted inverse iteration with a fixed right-hand side

```
Require: Fixed right-hand side (base problem solution) $\Psi_0$.
  for i=0,1,2,... do
    Choose a shift $\alpha^{(i+1)}$ and an inner tolerance $\tau^{(i)} \geq 0$.
    Compute $\widetilde{\Psi}^{(i+1)}$ such that $\|(\mathcal{T} - \mathcal{S} - \alpha^{(i+1)}\mathcal{F})\widetilde{\Psi}^{(i+1)} - \mathcal{F}\Psi_0\|_* \leq \tau^{(i)}$.        (††)
    Normalise $\Psi^{(i+1)} = \widetilde{\Psi}^{(i+1)}/\|\mathcal{P}\widetilde{\Psi}^{(i+1)}\|_{L^2(V)}$.
  end for
```

Again, we implicitly require $\Psi^{(i)}, \widetilde{\Psi}^{(i)} \in L^2(V, L^1(\mathbb{S}^2))$. Using the same approach as in Section 3.2, we can apply Lemma 2.3 to get the following result.

Lemma 5.10. *If $\widetilde{\Psi}^{(i)}$ and $\Psi^{(i)}$ are computed by Algorithm 10, and if we introduce the corresponding scalar fluxes $\widetilde{\phi}^{(i)} := \mathcal{P}\widetilde{\Psi}^{(i)}$ and $\phi^{(i)} := \mathcal{P}\Psi^{(i)}$, then*

$$\begin{aligned}\|(\mathcal{I} - (\sigma_s + \alpha^{(i+1)}\nu\sigma_f)\mathcal{K}_\sigma)\widetilde{\phi}^{(i+1)} - \nu\sigma_f\mathcal{K}_\sigma\phi_0\|_{L^2(V)} \;&\leq\; \tau^{(i)}\,, \quad \textit{and} \\ \phi^{(i+1)} \;&=\; \frac{\widetilde{\phi}^{(i+1)}}{\|\widetilde{\phi}^{(i+1)}\|_{L^2(V)}}\,,\end{aligned}$$

where $\phi_0 := \mathcal{P}\Psi_0$.

Using Lemma 5.10 and a proof that is analogous to the proof of Theorem 3.3, we obtain the following result.

Theorem 5.11. *Suppose* $s(\phi_0) \neq 0$,

$$\|(\mathcal{I} - (\sigma_s + \alpha^{(i+1)}\,\nu\sigma_f)\mathcal{K}_\sigma)\widetilde{\phi}^{(i+1)} - \nu\sigma_f\mathcal{K}_\sigma\phi_0\|_{L^2(V)} \;\leq\; \tau^{(i)} \;,$$

$$\textit{and set} \quad \phi^{(i+1)} \;=\; \frac{\widetilde{\phi}^{(i+1)}}{\|\widetilde{\phi}^{(i+1)}\|_{L^2(V)}} \;.$$

Then, if $\tau^{(i)} < \nu\sigma_f\omega_1 c(\phi_0)$, *we have with constant* $C_1 = 1/(\nu\sigma_f\omega_2)$,

$$t(\phi^{(i+1)}) \;\leq\; \left(\frac{s(\phi_0) + C_1\,\tau^{(i)}}{c(\phi_0) - C_1\,\tau^{(i)}}\right)\left|\frac{\lambda_1 - \alpha^{(i+1)}}{\lambda_2 - \alpha^{(i+1)}}\right| \;. \tag{5.19}$$

The inequality (5.19) shows immediately that convergence of Algorithm 10 can only be guaranteed if the shifts $\alpha^{(i+1)}$ approach λ_1 and that in general decreasing tolerances to zero does not necessarily guarantee convergence.

Corollary 5.12. *Suppose that for every step in Algorithm 10 the conditions of Theorem 5.11 are satisfied, and that shifts* $\alpha^{(i+1)}$ *are applied with*

$$\alpha^{(i+1)} \;=\; \lambda_1 + \mathcal{O}(s(\phi^{(i)})^k) \;. \tag{5.20}$$

Then,

$$t(\phi^{(i+1)}) \;\leq\; \left(\frac{s(\phi_0) + C_1\,\tau^{(i)}}{c(\phi_0) - C_1\,\tau^{(i)}}\right)\left|\frac{C_2}{\lambda_2 - \lambda_1}\right| \; t(\phi^{(i)})^k \;, \quad C_1, C_2 \ \textit{constant} \;. \tag{5.21}$$

Hence, Algorithm 10 converges with rate k.

If we apply Algorithm 10 with sufficiently small fixed inner tolerances and the shifts $\alpha^{(i+1)}$ are the Rayleigh quotients $\widetilde{\rho}^{(i)}$ defined in Lemma 3.4, i.e.

$$\widetilde{\rho}^{(i)} \;=\; \frac{(\mathcal{P}\Psi^{(i)}, \mathcal{P}\mathcal{T}^{-1}(\mathcal{T} - \mathcal{S})\Psi^{(i)})_{L^2(V)}}{(\mathcal{P}\Psi^{(i)}, \mathcal{P}\mathcal{T}^{-1}\mathcal{F}\Psi^{(i)})_{L^2(V)}} \;=\; \frac{(\phi^{(i)}, (\mathcal{I} - \sigma_s\mathcal{K}_\sigma)\phi^{(i)})_{L^2(V)}}{(\phi^{(i)}, \nu\sigma_f\mathcal{K}_\sigma\phi^{(i)})_{L^2(V)}} \;,$$

then we obtain quadratic convergence. This is the same convergence rate that we obtained for the standard shifted inverse iteration method in Algorithm 5 on page 62 when using constant tolerances.

To guarantee linear convergence in the case when $k = 1$ in (5.20), the coefficient in (5.21) must satisfy

$$\left(\frac{s(\phi_0) + C_1\,\tau^{(i)}}{c(\phi_0) - C_1\,\tau^{(i)}}\right)\left|\frac{C_2}{\lambda_2 - \lambda_1}\right| \;<\; 1 \quad \text{for all} \quad i = 0, 1, 2, \ldots \;. \tag{5.22}$$

This requires sufficiently small inner tolerances $\tau^{(i)}$ and a sufficiently "close" (base problem solution) Ψ_0 as approximation of the eigenfunction. The condition (5.22) can be considered to determine a ball of convergence which depends on the choice of the fixed right-hand side Ψ_0 and the tolerances $\tau^{(i)}$ for the inner solves.

Contrary to the case of the standard inverse iteration in Section 3.2, using decreasing tolerances, given by

$$\tau^{(i)} \;\leq\; C_3 s(\phi^{(i)}) \ , \quad C_3 \text{ constant} \ , \tag{5.23}$$

does not improve the convergence rate in theory since the initial error $s(\phi_0)/c(\phi_0)$ remains present. In particular, for fixed shifts we get the following result.

Corollary 5.13. *If in every iteration of Algorithm 10 the conditions of Theorem 5.11 are met and fixed shifts* $\alpha^{(i+1)} = \alpha_0$ *are used, then*

$$t(\phi^{(i+1)}) \;\leq\; \left(\frac{s(\phi_0) + C_1\tau^{(i)}}{c(\phi_0) - C_1\tau^{(i)}}\right) \left|\frac{\lambda_1 - \alpha_0}{\lambda_2 - \alpha_0}\right| \quad .$$

The corollary clearly shows that there is no guaranteed convergence for fixed shifts, even if the solves were performed exactly ($\tau^{(i)} = 0$). This is not surprising as in this case the linear systems that are solved remain the same in every iteration.

Tables 3.1 and 3.2 contained numerical results for Algorithms 1 to 4 (the *standard* iterative methods, i.e. power method, shifted inverse iteration, inverse correction and simplified Jacobi-Davidson) which we described in Chapter 3. We showed that these methods are related and discussed that, under certain conditions, they are equivalent.

Table 5.2 now summarises the variations of the standard iterative methods that we introduced in this chapter. We investigated the relation of the different schemes which we will use in the next section to obtain a form of the method of perturbation in Algorithm 7 (using a special shift and adjusted tolerances) for which linear convergence of the method can be shown. This is done by exploiting the convergence properties of shifted inverse iteration with a fixed right-hand side which we established in this section.

Algorithms with a fixed right-hand side	
method of perturbation	Algorithm 7
inverse correction with a fixed right-hand side	Algorithm 8
shifted inverse iteration with a fixed right-hand side	Algorithms 9 & 10

Table 5.2: *Overview of the different algorithms considered in Chapter 5.*

5.4 Convergence theory for the method of perturbation

In order to analyse the convergence of the method of perturbation, we use the relation to the shifted inverse iteration approach with a fixed right-hand side from Section 5.2 and the convergence analysis from Section 5.3. Recall that by Corollary 5.12, the convergence rate of Algorithm 10 is determined by the convergence of the shift $\alpha^{(i+1)}$ to the true eigenvalue.

As shown in Corollary 5.9, Algorithm 10 with shifts from (5.11), i.e.

$$\alpha^{(i+1)} \;=\; \frac{\langle \Psi_0^*, (\mathcal{T}-\mathcal{S})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle} \;,$$

produces the same iterates as the inexact version of the method of perturbation in Algorithm 7, provided the latter uses the adjusted inner tolerances $\widetilde{\tau}^{(i)}$ defined in (5.17). Hence, the convergence of Algorithm 7 is determined by how well the shifts $\alpha^{(i+1)}$ approximate the smallest eigenvalue λ_1.

In the same way as in Lemma 3.4, we consider special versions of the shifts $\alpha^{(i+1)}$.

Lemma 5.14. *Given* $\Psi^{(i)} \in L^2(V, L^1(\mathbb{S}^2))$ *, consider the shift*

$$\widetilde{\alpha}^{(i+1)} \;:=\; \frac{(\mathcal{P}\Psi_0^*, \mathcal{P}\mathcal{T}^{-1}(\mathcal{T}-\mathcal{S})\Psi^{(i)})_{L^2(V)}}{(\mathcal{P}\Psi_0^*, \mathcal{P}\mathcal{T}^{-1}\mathcal{F}\Psi^{(i)})_{L^2(V)}} \;=\; \frac{(\phi_0^*, (\mathcal{I}-\sigma_s\mathcal{K}_\sigma)\phi^{(i)})_{L^2(V)}}{(\phi_0^*, \nu\sigma_f\mathcal{K}_\sigma\phi^{(i)})_{L^2(V)}} \;. \tag{5.24}$$

Then,

$$\widetilde{\alpha}^{(i+1)} \;=\; \lambda_1 \;+\; \mathcal{O}(s(\phi^{(i)})) \;.$$

Proof. We use again the orthogonal splitting

$$\phi^{(i)} \;=\; c(\phi^{(i)})e_1 \;+\; s(\phi^{(i)})u^{(i)} \;,$$

where $\|u^{(i)}\|_{L^2(V)} = 1$ and $(u^{(i)}, e_1)_{L^2(V)} = 0$, as well as $\phi_0^* \;=\; c_0^* e_1 \;+\; s_0^* u_0^*$, where $\|u_0^*\|_{L^2(V)} = 1$ and $(u_0^*, e_1)_{L^2(V)} = 0$. Since $(1-\sigma_s\omega_1) = \lambda_1\,\nu\sigma_f\omega_1$, as given in (2.24), we get

$$\widetilde{\alpha}^{(i+1)} \;=\; \frac{(1-\sigma_s\omega_1)\,c_0^* c(\phi^{(i)}) \;+\; \mathcal{O}(s(\phi^{(i)}))}{\nu\sigma_f\omega_1\,c_0^* c(\phi^{(i)}) \;+\; \mathcal{O}(s(\phi^{(i)}))} \;=\; \lambda_1 \;+\; \mathcal{O}(s(\phi^{(i)})) \;.$$

Hence $\widetilde{\alpha}^{(i+1)}$ approximates λ_1 linearly. □

Combining the above results, we can construct the following version of the method of perturbation and give an estimate for the convergence rate.

Algorithm 11 Method of perturbation with linear convergence rate

```
Require: Base problem solutions $\lambda_0$, $\Psi_0$, $\Psi_0^*$, starting guesses $\Delta\Psi^{(0)}$, $\Delta\lambda^{(0)}$.
  for i=0,1,2,... do
    Compute $\Delta\widetilde{\lambda}^{(i+1)} = \frac{(\mathcal{P}\Psi_0^*, \mathcal{P}\mathcal{T}^{-1}(\Delta\mathcal{T}-\Delta\mathcal{S}-\lambda_0\Delta\mathcal{F})(\Psi_0+\Delta\Psi^{(i)}))_{L^2(V)}}{(\mathcal{P}\Psi_0^*, \mathcal{P}\mathcal{T}^{-1}\mathcal{F}(\Psi_0+\Delta\Psi^{(i)}))_{L^2(V)}}$ .
    Obtain $\Delta\Psi^{(i+1)}$ such that
    $\big\|\big(\mathcal{T}-\mathcal{S}-(\lambda_0+\Delta\widetilde{\lambda}^{(i+1)})\mathcal{F}\big)\Delta\Psi^{(i+1)}+(\Delta\mathcal{T}-\Delta\mathcal{S}-\lambda_0\Delta\mathcal{F})\Psi_0-\Delta\widetilde{\lambda}^{(i)}\mathcal{F}\Psi_0\big\|_* \leq \widetilde{\tau}^{(i)}$ .
    Define $\widetilde{\Psi}^{(i+1)} = \Psi_0 + \Delta\Psi^{(i+1)}$ .
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$ .
  end for
```

We can now prove the following convergence result.

Theorem 5.15. *The method of perturbation as given in Algorithm 11 converges linearly if the solution Ψ_0 of the base problem is close enough to the required eigendirection (i.e. if the perturbation is not too large) and the tolerances for the linear systems are given by*

$$\widetilde{\tau}^{(i)} := |\widetilde{\alpha}^{(i)} - \widetilde{\alpha}^{(i+1)}|\,\tau^{(i)} = |\Delta\widetilde{\lambda}^{(i)} - \Delta\widetilde{\lambda}^{(i+1)}|\,\tau^{(i)} , \tag{5.25}$$

with $\tau^{(i)}$ sufficiently small and $\widetilde{\alpha}^{(i)}$ given by (5.24).

Proof. Analogous to the proof of Lemma 5.2 it can be shown that $\widetilde{\alpha}^{(i+1)} = \lambda_0 + \Delta\widetilde{\lambda}^{(i+1)}$. The same arguments as used in Section 5.2 to obtain Corollary 5.9 can be used, for the shifts (5.24) and tolerances (5.25), to give that Algorithm 11 produces, up to a different sign, the same iterates as Algorithm 10 with shifts (5.24). By (5.21) and Lemma 5.14 we then obtain

$$t(\phi^{(i+1)}) \leq \left(\frac{s(\phi_0) + C_1\,\tau^{(i)}}{c(\phi_0) - C_1\,\tau^{(i)}}\right) \left|\frac{C_2}{\lambda_2 - \lambda_1}\right| t(\phi^{(i)}) , \quad C_1, C_2 \text{ constant} . \tag{5.26}$$

Now, for a suitable Ψ_0, the corresponding ϕ_0, together with sufficiently small inner tolerances $\tau^{(i)}$, satisfies the condition (5.22), and therefore the method converges linearly. □

Note that, even for exact solves, the coefficient $\frac{s(\phi_0)}{c(\phi_0)}\left|\frac{C_2}{\lambda_2-\lambda_1}\right|$ in (5.26) can be larger than one if the base problem solution (and right-hand side) Ψ_0 leads to a $\phi_0 = \mathcal{P}\Psi_0$ that is a poor approximation of the eigenfunction e_1. This suggests that for large perturbations, where $s(\phi_0)/c(\phi_0)$ is not small (and when additionally λ_1 is close to λ_2), the method of perturbation might not converge.

However, by dividing the large perturbation up into a sequence of smaller perturbations, and solving the problems in succession, using the latest solution as the fixed right-hand side Ψ_0 of the next problem, convergence can possibly be regained. Note also that the inner tolerances $\tau^{(i)}$ in (5.26) (and hence $\widetilde{\tau}^{(i)}$ via (5.25)) have to be sufficiently small in order to guarantee convergence.

5.5 A modified version of the method of perturbation as used in industry

In practice it can be very expensive to solve problems of the form

$$(\mathcal{T} - \mathcal{S} - \alpha\mathcal{F})\Psi \;=\; Q \;, \tag{5.27}$$

with the transport, scatter and fission operator on the left-hand side. However, it is comparably cheap (especially when using Monte Carlo methods, as we saw in Chapter 4) to solve a problem that only contains the transport operator on the left, i.e.

$$\mathcal{T}\Psi \;=\; Q \;.$$

We discussed in Section 4.3.2 a splitting approach, the source iteration, that makes use of these cheap "transport sweeps" to solve iteratively for the solution of (5.27). In the following we describe how this technique has been applied in practice at Serco Technical and Assurance Services, resulting in an alternative version of the method of perturbation.

An approximation of the solution for

$$\big(\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda^{(i+1)})\mathcal{F}\big)\Delta\Psi^{(i+1)} \;=\; -(\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 + \Delta\lambda^{(i)}\mathcal{F}\Psi_0 \tag{5.28}$$

can be obtained by solving

$$\mathcal{T}\Delta\Psi_k^{(i+1)} = -(\Delta\mathcal{T}-\Delta\mathcal{S}-\lambda_0\Delta\mathcal{F})\Psi_0+\Delta\lambda^{(i)}\mathcal{F}\Psi_0+(\mathcal{S}+(\lambda_0+\Delta\lambda^{(i+1)})\mathcal{F})\Delta\Psi_{k-1}^{(i)} \tag{5.29}$$

for $k = 1, 2, \ldots$. However, in practice the problem (5.29) is not solved repeatedly for $k = 1, 2, \ldots$ up to convergence. Instead, only one iteration is performed before (5.10) is used to update $\Delta\lambda$. This is an example where the outer (eigenvalue) iteration is intertwined with an inexact inner iteration. The algorithm is stated below.

Algorithm 12 also differs from Algorithm 7 in the sense that the linear system (5.29) contains the same current approximation $\Delta\lambda^{(i+1)}$ twice on the right-hand side while we used $\Delta\lambda^{(i)}$ and $\Delta\lambda^{(i+1)}$ in the first version of the method of perturbation. The reason

Algorithm 12 Method of perturbation – second version

```
Require: Base problem solutions $\lambda_0$, $\Psi_0$, $\Psi_0^*$, starting guess $\Delta\Psi^{(0)}$.
  for i=0,1,2,... do
    Compute $\Delta\lambda^{(i+1)} = \frac{\langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0 \Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle}$ .
    Obtain $\Psi^{(i+1)}$ by solving
    $\mathcal{T}\Delta\Psi^{(i+1)} = -(\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 + \Delta\lambda^{(i+1)}\mathcal{F}\Psi_0 + (\mathcal{S} + (\lambda_0 + \Delta\lambda^{(i+1)})\mathcal{F})\Delta\Psi^{(i)}$ .
    Define $\widetilde{\Psi}^{(i+1)} = \Psi_0 + \Delta\Psi^{(i+1)}$ .
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$ .
  end for
```

for this was that, for exact solves, the solution of the linear system is $\Delta\Psi^{(i+1)} = -\Psi_0$ which would lead to $\widetilde{\Psi}^{(i+1)} = 0$. By moving the scatter and fission terms together with $\Delta\Psi^{(i)}$ to the right-hand side, and only performing one transport sweep, this problem for exact solves is avoided.

Remark 5.16. *We have no guarantee that this algorithm converges. If the shift in front of the $\mathcal{F}\Psi_0$ term had remained $\Delta\lambda^{(i)}$ and the inner problems* (5.29) *were solved repeatedly until*

$$\left\| (\mathcal{T} - \mathcal{S} - (\lambda_0 + \Delta\lambda^{(i+1)})\mathcal{F})\Delta\Psi^{(i+1)} + (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 - \Delta\lambda^{(i)}\mathcal{F}\Psi_0 \right\| \leq \tau^{(i)} ,$$

the method would just be a certain implementation of Algorithm 7, where an operator splitting approach is used to provide an inner solver. Unfortunately, by mixing the inner solves with the outer eigenvalue iterations our analysis above no longer applies.

We investigate convergence of Algorithm 12 numerically at the end of Section 5.6, where we will observe that it is possible to reduce the number of outer iterations by doing several "transport sweeps" (i.e. solving (5.29)) before updating the eigenvalue.

5.6 Numerical results

We now discuss numerical experiments for the two model problems described in Section 3.3 and start with results for the inexact shifted inverse iteration method with a fixed right-hand side as described in Algorithm 10.

5.6.1 Los Alamos benchmark test set problem

Let us consider again the problem number 2 of the Los Alamos benchmark test set [109] which corresponds to the 1D problem (2.4), (2.5). In order to preserve the underlying symmetry, we employ the Euler scheme discussed in Section 2.4.2.

As in Chapter 3 we use $M = 128$ equally sized spatial intervals and $2N = 128$ angular Gauss points leading to a non-symmetric generalised matrix eigenvalue problem of dimension 16384 × 16384. The eigenvalues nearest zero of the discrete problem are $\lambda_1 \approx 0.99570$ and $\lambda_2 \approx 2.60907$.

The stopping criterion for the outer iteration demands again that $\|\mathbf{res}^{(i)}\|_2 < 10^{-14}$, where $\mathbf{res}^{(i)} = (T - S - \rho^{(i)} F)\mathbf{\Psi}^{(i)}$ with T, S, F, and $\mathbf{\Psi}$ being the discrete versions of $\mathcal{T}$, $\mathcal{S}$, $\mathcal{F}$, and Ψ, respectively. The eigenvalue approximation is given by $\rho^{(i)} = \rho(\mathbf{\Psi}^{(i)})$, where

$$\rho(\mathbf{\Psi}^{(i)}) \;=\; \frac{\langle \mathbf{\Psi}^{(i)}, (T-S)\mathbf{\Psi}^{(i)} \rangle}{\langle \mathbf{\Psi}^{(i)}, F\mathbf{\Psi}^{(i)} \rangle} \tag{5.30}$$

is the standard Rayleigh quotient with $\langle \cdot\,, \cdot \rangle$ representing the ℓ_2 inner product over all spatial and angular discrete variables.

Problem (††) in Algorithm 10 is solved using the GMRES function in MATLAB 2009b with an LU factorisation of T as preconditioner. As fixed right-hand side $\mathbf{\Psi}_0$ we use a normalised vector with equal positive entries. To measure the convergence rate of Algorithm 10, we consider again the eigenvalue error $\Delta^{(i)} = |\lambda_1 - \rho^{(i)}|$, where λ_1 is the computed eigenvalue that satisfies the stopping criterion.

Our first test uses fixed shifts and decreasing tolerances $\tau^{(i)} \le 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$ for the inner solves, where P denotes the discrete version of the projection operator $\mathcal{P}$. The standard inverse iteration in Chapter 3 converged linearly for fixed shifts $\alpha_0 = 0.9$ and $\alpha_0 = 0.99$ and decreasing tolerances, but, as already predicted by Corollary 5.13, the Table 5.3 shows that the fixed right-hand side inverse iteration in Algorithm 10 does not converge for fixed shifts.

	$\alpha_0 = 0.9$		$\alpha_0 = 0.99$	
i	$\Delta^{(i)}$	$\|\mathtt{res}^{(i)}\|_2$	$\Delta^{(i)}$	$\|\mathtt{res}^{(i)}\|_2$
0	3.28E-02	6.20E-02	3.28E-02	6.20E-02
1	2.63E-04	1.11E-04	1.39E-05	6.74E-06
2	2.63E-04	1.11E-04	1.38E-05	6.73E-06
3	2.63E-04	1.11E-04	1.38E-05	6.73E-06
2000	2.63E-04	1.11E-04	1.38E-05	6.73E-06

Table 5.3: *Numerical results of Algorithm 10 for fixed shifts and decreasing tolerances* $\tau^{(i)} \le 0.1\,\|PT^{-1}\,\textbf{\textit{res}}^{(i)}\|_2$.

However, Corollary 5.12 suggests that the method will converge, provided the applied shifts approach the eigenvalue λ_1 and the initial guess is "close enough" to the true eigenvector. We now investigate this by testing different choices for the shift. The first variable shift that we test is the standard Rayleigh quotient $\rho^{(i)}$ given in (5.30). The second shift is the discrete version of the non-standard Rayleigh quotient

$$\widetilde{\rho}^{(i)} = \frac{(\mathcal{P}\Psi^{(i)}, \mathcal{P}\mathcal{T}^{-1}(\mathcal{T}-\mathcal{S})\Psi^{(i)})_{L^2(V)}}{(\mathcal{P}\Psi^{(i)}, \mathcal{P}\mathcal{T}^{-1}\mathcal{F}\Psi^{(i)})_{L^2(V)}} = \frac{(\phi^{(i)}, (\mathcal{I}-\sigma_s\mathcal{K}_\sigma)\phi^{(i)})_{L^2(V)}}{(\phi^{(i)}, \nu\sigma_f\mathcal{K}_\sigma\phi^{(i)})_{L^2(V)}} . \qquad (5.31)$$

For constant tolerances $\tau_0 = 0.1$, Table 5.4 shows that Algorithm 10 converges linearly when using the standard Rayleigh quotient $\rho^{(i)}$, and the table suggests quadratic convergence for the non-standard Rayleigh quotient $\widetilde{\rho}^{(i)}$. These are the same convergence rates that we observed for the standard inexact inverse iteration approach in Algorithm 5 with constant tolerances (see Table 3.8).

	$\alpha^{(i+1)} = \rho^{(i)}$			$\alpha^{(i+1)} = \widetilde{\rho}^{(i)}$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	9.7E-05	3.0E-03	9.1E-02	9.4E-05	2.9E-03	8.8E-02
2	2.7E-07	2.8E-03	2.9E+01	4.7E-09	5.0E-05	5.3E-01
3	7.1E-10	2.6E-03	9.6E+03	0.0E+00	0.0E+00	0.0E+00
4	1.8E-12	2.5E-03	3.6E+06			
5	0.0E+00	0.0E+00	0.0E+00			

Table 5.4: *Numerical results for Algorithm 10 with constant tolerances $\tau_0 = 0.1$ and two different Rayleigh quotient shifts $\rho^{(i)}$ and $\widetilde{\rho}^{(i)}$ defined in* (5.30) *and* (5.31) *for matrices arising from the application of the symmetry preserving Euler scheme described in Section 2.4.2.*

We saw in Section 3.2 that Algorithm 5 was able to gain an additional order in the convergence rate when demanding decreasing tolerances satisfying (5.23). However, we also noted in Corollary 5.13 that this is not guaranteed for Algorithm 10 as the error from the initial guess remains present. Table 5.5 confirms that there is no further improvement in the convergence rate of Algorithm 10 compared to the constant tolerance case in Table 5.4.

	$\alpha^{(i+1)} = \rho^{(i)}$			$\alpha^{(i+1)} = \widetilde{\rho}^{(i)}$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	7.5E-05	2.3E-03	7.0E-02	7.2E-05	2.2E-03	6.7E-02
2	1.8E-07	2.4E-03	3.2E+01	4.4E-09	6.1E-05	8.5E-01
3	4.3E-10	2.4E-03	1.3E+04	0.0E+00	0.0E+00	0.0E+00
4	1.0E-12	2.4E-03	5.6E+06			
5	0.0E+00	0.0E+00	0.0E+00			

Table 5.5: *Numerical results for the fixed right-hand side shifted inverse iteration method in Algorithm 10 with decreasing tolerances $\tau^{(i)} \leq 0.1 \, \|PT^{-1}\, \mathbf{res}^{(i)}\|_2$ and the two Rayleigh quotient shifts $\rho^{(i)}$ and $\widetilde{\rho}^{(i)}$.*

Note that when using Monte Carlo methods to solve the linear systems, the accuracy of the solutions is in general fixed and does not decrease to zero. In this case the linear

	$\alpha^{(i+1)} = \rho_0^{(i)}$			$\alpha^{(i+1)} = \widetilde{\rho}_0^{(i)}$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	9.7E-05	3.0E-03	9.1E-02	9.4E-05	2.9E-03	8.8E-02
2	1.4E-06	1.4E-02	1.5E+02	6.1E-07	6.4E-03	6.8E+01
3	3.4E-08	2.5E-02	1.8E+04	3.9E-09	6.5E-03	1.1E+04
4	7.1E-10	2.1E-02	6.1E+05	2.8E-11	7.2E-03	1.8E+06
5	1.6E-11	2.3E-02	3.3E+07	1.8E-13	6.5E-03	2.3E+08
6	5.3E-13	3.3E-02	2.0E+09	0.0E+00	0.0E+00	0.0E+00
7	1.1E-14	2.0E-02	3.7E+10			
8	0.0E+00	0.0E+00	0.0E+00			

Table 5.6: *Numerical results for Algorithm 10 with constant tolerances* $\tau_0 = 0.1$ *and the two shifts* $\rho_0^{(i)}$ *and* $\widetilde{\rho}_0^{(i)}$ *defined in* (5.32) *and* (5.33).

systems in Algorithm 5 are not solved with decreasing tolerances (or even exactly) to achieve an additional order in the convergence rate. Therefore, Algorithms 5 and 10 are likely to obtain the same convergence rates when Monte Carlo techniques are used to solve the linear systems.

In addition, for deterministic methods that use GMRES or other Krylov subspace based methods as an inner solver, Algorithm 10 can provide computational benefits over Algorithm 5 if the inner solver is adjusted. We will describe this briefly in Section 5.6.3.

In order to test now a shift that is similar to the shift (5.11), for which we established the equivalence between Algorithm 10 and Algorithm 11, we use the discrete versions of the two shifts

$$\rho_0^{(i)} := \frac{\langle \Psi_0, (\mathcal{T} - \mathcal{S})\Psi^{(i)} \rangle}{\langle \Psi_0, \mathcal{F}\Psi^{(i)} \rangle} , \tag{5.32}$$

and its scalar flux counterpart, defined by

$$\widetilde{\rho}_0^{(i)} := \frac{(\mathcal{P}\Psi_0, \mathcal{P}\mathcal{T}^{-1}(\mathcal{T} - \mathcal{S})\Psi^{(i)})_{L^2(V)}}{(\mathcal{P}\Psi_0, \mathcal{P}\mathcal{T}^{-1}\mathcal{F}\Psi^{(i)})_{L^2(V)}} = \frac{(\phi_0, (\mathcal{I} - \sigma_s \mathcal{K}_\sigma)\phi^{(i)})_{L^2(V)}}{(\phi_0, \nu\sigma_f \mathcal{K}_\sigma \phi^{(i)})_{L^2(V)}} . \tag{5.33}$$

Using a proof that is analogous to the proof of Lemma 5.14, we obtain that this latter shift also approximates the dominant eigenvalue to first order, and therefore we expect linear convergence of Algorithm 10 in this case. Table 5.6 confirms the linear convergence rate (for both shifts) when a constant tolerance $\tau_0 = 0.1$ is used.

As in the case for the Rayleigh quotient shifts, using decreasing tolerances, instead of a constant tolerance τ_0, does not improve the convergence rate. Table 5.7 confirms that this is also the case for the shifts $\rho_0^{(i)}$ and $\widetilde{\rho}_0^{(i)}$.

	$\alpha^{(i+1)} = \rho_0^{(i)}$			$\alpha^{(i+1)} = \widetilde{\rho}_0^{(i)}$		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	3.3E-02			3.3E-02		
1	7.5E-05	2.3E-03	7.0E-02	7.2E-05	2.2E-03	6.7E-02
2	1.9E-06	2.5E-02	3.4E+02	5.2E-07	7.2E-03	1.0E+02
3	4.5E-08	2.4E-02	1.3E+04	3.5E-09	6.8E-03	1.3E+04
4	1.1E-09	2.4E-02	5.3E+05	2.4E-11	6.8E-03	1.9E+06
5	2.6E-11	2.4E-02	2.2E+07	1.6E-13	6.8E-03	2.9E+08
6	6.3E-13	2.5E-02	9.5E+08	0.0E+00	0.0E+00	0.0E+00
7	0.0E+00	0.0E+00	0.0E+00			

Table 5.7: *Numerical results for Algorithm 10 with the two shifts* $\rho_0^{(i)}$ *and* $\widetilde{\rho}_0^{(i)}$ *defined in* (5.32) *and* (5.33) *and decreasing tolerances* $\tau^{(i)} \leq 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$.

5.6.2 Control rod insertion model problem

For the next tests we consider the control rod model problem described in Section 3.3.2 which we also used for the numerical results in Section 3.4.2. We apply a Gauss quadrature and Crank-Nicolson scheme with 128 uniform spatial intervals in the fuel region and 8 equally sized intervals in the absorber part of the problem (resolving the material boundary). For the angular discretisation we consider again 128 angular directions.

As before, three different material compositions in the absorber region are investigated: (i) The pure absorber case; (ii) a mix of 10% absorber and 90% water; and (iii) the homogeneous case, where the absorber and fuel region have the same cross-sections. The principal eigenvalues in cases (i) to (iii) are $\lambda_1 \approx 1.18$, 0.92, and 0.85, respectively.

The cross-section details and the problem dimensions were given in Table 3.11. For the fixed $\mathbf{\Psi}_0$ on the right-hand side (the base problem solution), we use, in the first test, a vector with entries that are chosen randomly from a uniform distribution on the interval $(0, 1)$. As already noted before, our theory does not apply directly to this problem since, in particular, no underlying symmetry is known.

When we use a fixed shift and fixed (as well as decreasing) tolerances in Algorithm 10, the method stagnates. However, by applying the standard Rayleigh quotient $\rho^{(i)}$ from (5.30) instead of a fixed shift, Table 5.8 shows that we obtain linear convergence of the method.

While we observed quadratic convergence for the Los Alamos problem in the previous section when using the special Rayleigh quotient $\widetilde{\rho}^{(i)}$ in (5.31), we now obtain only linear convergence (see Table 5.9). This agrees with the results in Table 3.14 for the standard shifted inverse iteration approach.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	3.1E-03	3.4E-03	3.8E-03	8.9E-03	1.0E-02	1.1E-02	3.1E-03	3.6E-03	4.3E-03
2	9.4E-05	3.1E-02	1.0E+01	8.6E-05	9.6E-03	1.1E+00	2.4E-06	7.9E-04	2.6E-01
3	2.2E-06	2.3E-02	2.5E+02	1.6E-06	1.9E-02	2.2E+02	3.6E-10	1.5E-04	6.3E+01
4	7.4E-09	3.4E-03	1.6E+03	1.1E-09	6.8E-04	4.3E+02	4.2E-12	1.2E-02	3.2E+07
5	1.8E-11	2.4E-03	3.2E+05	6.1E-12	5.6E-03	5.2E+06	0.0E+00	0.0E+00	0.0E+00
6	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00			

Table 5.8: *Numerical results for Algorithm 10 with* $\alpha^{(i+1)} = \rho^{(i)}$ *and constant tolerances* $\tau_0 = 0.1$.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	1.1E-02	1.3E-02	1.4E-02	1.7E-05	1.9E-05	2.2E-05	1.9E-05	2.3E-05	2.7E-05
2	2.4E-04	2.1E-02	1.9E+00	1.0E-06	5.9E-02	3.4E+03	1.1E-07	5.8E-03	3.0E+02
3	3.3E-07	1.4E-03	5.6E+00	6.6E-08	6.6E-02	6.5E+04	3.0E-09	2.7E-02	2.4E+05
4	9.5E-08	2.9E-01	8.9E+05	1.1E-09	1.7E-02	2.6E+05	3.2E-11	1.1E-02	3.7E+06
5	1.5E-09	1.6E-02	1.6E+05	7.2E-12	6.5E-03	5.9E+06	0.0E+00	0.0E+00	0.0E+00
6	4.3E-11	2.9E-02	2.0E+07	0.0E+00	0.0E+00	0.0E+00			
7	0.0E+00	0.0E+00	0.0E+00						

Table 5.9: *Numerical results for Algorithm 10 with shifts* $\alpha^{(i+1)} = \widetilde{\rho}^{(i)}$ *and constant tolerances* $\tau_0 = 0.1$.

Table 5.10 suggests that applying decreasing tolerances does in this case reduce the number of iterations needed compared to Table 5.8, but the convergence rate appears to remain linear and the faster convergence is only due to smaller convergence coefficients.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	1.3E-05	1.5E-05	1.6E-05	2.7E-05	3.0E-05	3.4E-05	1.0E-05	1.2E-05	1.4E-05
2	1.7E-09	1.3E-04	9.8E+00	8.2E-10	3.1E-05	1.1E+00	2.5E-12	2.4E-07	2.3E-02
3	2.7E-13	1.6E-04	9.2E+04	1.3E-14	1.6E-05	2.0E+04	0.0E+00	0.0E+00	0.0E+00
4	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00			

Table 5.10: *Numerical results for Algorithm 10 with* $\alpha^{(i+1)} = \rho^{(i)}$ *and decreasing tolerances* $\tau^{(i)} \le 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$.

Next we apply the shifts $\rho_0^{(i)}$ and $\widetilde{\rho}_0^{(i)}$ from (5.32) and (5.33). Note that the right-hand side $\mathbf{\Psi}_0$ is now a vector with random entries, while for the benchmark test in Section 5.6.1 we used a vector with equal entries. Table 5.11 shows that the method still converges linearly for shifts $\rho_0^{(i)}$. We obtained similar numerical results with linear convergence for shifts $\widetilde{\rho}_0^{(i)}$.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	3.1E-03	3.4E-03	3.8E-03	8.9E-03	1.0E-02	1.1E-02	3.1E-03	3.6E-03	4.3E-03
2	3.8E-04	1.2E-01	4.1E+01	9.4E-05	1.1E-02	1.2E+00	2.9E-05	9.5E-03	3.1E+00
3	1.7E-06	4.4E-03	1.2E+01	1.5E-06	1.6E-02	1.7E+02	1.0E-07	3.6E-03	1.2E+02
4	1.7E-08	1.0E-02	6.1E+03	1.2E-09	8.0E-04	5.4E+02	1.7E-10	1.6E-03	1.5E+04
5	2.1E-11	1.2E-03	7.1E+04	9.4E-12	7.8E-03	6.5E+06	0.0E+00	0.0E+00	0.0E+00
6	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00			

Table 5.11: *Numerical results for Algorithm 10 with constant tolerances* $\tau_0 = 0.1$ *and shifts* $\alpha^{(i+1)} = \rho_0^{(i)}$ *defined in* (5.32).

In order to compare the numerical results for the control rod problem with those from Table 5.6 for the Los Alamos test problem above, we construct the shift

$$\rho_1^{(i)} \ := \ \frac{\langle f_1, (\mathcal{T} - \mathcal{S})\Psi^{(i)} \rangle}{\langle f_1, \mathcal{F}\Psi^{(i)} \rangle} \tag{5.34}$$

with $f_1 \equiv 1$. Note that this is also the Rayleigh quotient (4.38) that is used in the Monte Carlo calculations to obtain an estimate for the eigenvalue λ_1.

We additionally test the discrete version of the scalar flux counterpart of $\rho_1^{(i)}$, which is defined by

$$\widetilde{\rho}_1^{(i)} \quad := \quad \frac{(\mathcal{P}f_1, \mathcal{P}\mathcal{T}^{-1}(\mathcal{T} - \mathcal{S})\Psi^{(i)})_{L^2(V)}}{(\mathcal{P}f_1, \mathcal{P}\mathcal{T}^{-1}\mathcal{F}\Psi^{(i)})_{L^2(V)}} \ . \tag{5.35}$$

Table 5.12 shows that Algorithm 10 shifts with $\alpha^{(i+1)} = \rho_1^{(i)}$ converges linearly.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	2.0E-04	2.2E-04	2.5E-04	6.1E-05	6.9E-05	7.8E-05	6.5E-06	7.7E-06	9.1E-06
2	2.7E-06	1.3E-02	6.6E+01	3.1E-07	5.1E-03	8.3E+01	5.6E-09	8.6E-04	1.3E+02
3	1.3E-08	4.8E-03	1.8E+03	1.6E-09	5.1E-03	1.6E+04	2.2E-11	3.8E-03	6.8E+05
4	2.1E-11	1.7E-03	1.3E+05	8.0E-12	5.1E-03	3.2E+06	0.0E+00	0.0E+00	0.0E+00
5	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00			

Table 5.12: *Numerical results for Algorithm 10 with constant tolerances* $\tau_0 = 0.1$ *and discrete versions of the shifts* $\alpha^{(i+1)} = \rho_1^{(i)}$ *defined in* (5.34).

For $\alpha^{(i+1)} = \widetilde{\rho}_1^{(i)}$, we obtain again a linear convergence rate but more iterations are needed (see Table 5.13). These are the same convergence rates that we observed for the Los Alamos benchmark test set problem in Table 5.6.

For the next test we use the shift $\alpha^{(i+1)}$ from (5.11) in Algorithm 10. We also change

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	1.1E-02	1.2E-02	1.4E-02	1.4E-05	1.5E-05	1.7E-05	1.1E-04	1.3E-04	1.5E-04
2	2.5E-04	2.3E-02	2.1E+00	1.0E-04	7.3E+00	5.4E+05	2.7E-07	2.5E-03	2.2E+01
3	8.3E-07	3.3E-03	1.3E+01	1.9E-06	1.9E-02	1.9E+02	2.5E-09	9.3E-03	3.4E+04
4	9.3E-09	1.1E-02	1.3E+04	5.4E-08	2.9E-02	1.5E+04	1.4E-11	5.6E-03	2.2E+06
5	3.7E-10	4.0E-02	4.3E+06	9.9E-10	1.9E-02	3.4E+05	1.8E-13	1.3E-02	9.4E+08
6	2.6E-12	7.0E-03	1.9E+07	6.1E-13	6.2E-04	6.2E+05	0.0E+00	0.0E+00	0.0E+00
7	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00			

Table 5.13: *Numerical results for Algorithm 10 with constant tolerances* $\tau_0 = 0.1$ *and shifts* $\alpha^{(i+1)} = \widetilde{\rho}_1^{(i)}$ *defined in* (5.35).

the vector $\mathbf{\Psi}_0$ that is used to compute the fixed right-hand side to the more realistic solution of a control rod base problem with pure water in the absorber region.

We showed in Corollary 5.9 that this method is then equivalent to the method of perturbation in Algorithm 7 if the latter uses the adjusted tolerances $\tau^{(i)}$ in (5.17). We will discuss this relationship further below in more detail. Table 5.14, indicating a fast linear convergence rate, describes the convergence properties of Algorithm 10 for this new fixed right-hand side and constant tolerances of $\tau_0 = 0.1$.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	2.2E-03	2.4E-03	2.7E-03	4.6E-07	5.2E-07	5.9E-07	6.3E-07	7.4E-07	8.8E-07
2	1.5E-06	7.0E-04	3.3E-01	1.8E-10	3.8E-04	8.1E+02	1.2E-10	1.9E-04	3.0E+02
3	1.0E-10	6.8E-05	4.5E+01	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
4	2.5E-14	2.4E-04	2.4E+06						
5	0.0E+00	0.0E+00	0.0E+00						

Table 5.14: *Numerical results for Algorithm 10 with constant tolerances* $\tau_0 = 0.1$ *and shifts* $\alpha^{(i+1)}$ *defined in* (5.11)*, when using a base problem solution for* $\mathbf{\Psi}_0$*.*

When we apply a very small tolerance $\tau_0 = 10^{-14}$, we obtain almost identical results to the ones given in Table 5.15, where the decreasing tolerances case is treated. The convergence – particularly for the more homogeneous cases with smaller or no jumps in the material cross-sections – is very fast. Especially the accuracy of the first iterate is extremely good and outperforms all the other initial approximations that we considered so far. This is probably due to the good guess that the new $\mathbf{\Psi}_0$ provides.

As we already observed for the other numerical results in this subsection, using a shift that is computed from the scalar fluxes does not improve the convergence for this test problem, but actually results in the need for slightly more iterations. Table 5.16 shows the results when the shifts $\widetilde{\alpha}^{(i+1)}$ from (5.24) and decreasing tolerances are used.

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	0.0E+00			0.0E+00			0.0E+00		
1	2.0E-06	2.2E-06	2.5E-06	2.6E-08	2.9E-08	3.3E-08	2.0E-09	2.3E-09	2.8E-09
2	3.5E-09	1.8E-03	9.1E+02	4.3E-13	1.6E-05	6.2E+02	5.2E-14	2.7E-05	1.4E+04
3	2.1E-12	5.8E-04	1.6E+05	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
4	0.0E+00	0.0E+00	0.0E+00						

Table 5.15: *Numerical results for Algorithm 10 with shifts* $\alpha^{(i+1)}$ *defined in* (5.11) *and decreasing tolerances* $\tau^{(i)} \leq 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$ *.*

	pure absorber			10% absorber, 90% water			homogeneous material		
i	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$	$\Delta^{(i)}$	$\frac{\Delta^{(i)}}{\Delta^{(i-1)}}$	$\frac{\Delta^{(i)}}{(\Delta^{(i-1)})^2}$
0	9.0E-01			8.9E-01			8.5E-01		
1	5.3E-05	5.9E-05	6.6E-05	1.3E-06	1.4E-06	1.6E-06	3.0E-08	3.5E-08	4.2E-08
2	8.3E-07	1.6E-02	2.9E+02	3.5E-09	2.8E-03	2.2E+03	1.8E-13	6.0E-06	2.0E+02
3	1.8E-08	2.2E-02	2.7E+04	8.0E-12	2.3E-03	6.4E+05	5.4E-15	3.0E-02	1.7E+11
4	4.1E-10	2.2E-02	1.2E+06	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00	0.0E+00
5	9.1E-12	2.2E-02	5.5E+07						
6	2.4E-13	2.7E-02	2.9E+09						
7	0.0E+00	0.0E+00	0.0E+00						

Table 5.16: *Numerical results for Algorithm 10 with shifts* $\widetilde{\alpha}^{(i+1)}$ *defined in* (5.24) *and decreasing tolerances* $\tau^{(i)} \leq 0.1\,\|PT^{-1}\,\mathbf{res}^{(i)}\|_2$ *.*

In all tables in this subsection the case (iii) with a homogeneous material mix and no jumps in the cross-sections converged fastest, suggesting that heterogeneity introduces additional difficulties to the problem and might have an impact on the convergence of the method.

Numerical results for the first version of the method of perturbation

Let us now consider the convergence properties of an inexact version of the method of perturbation described in Algorithm 7 with respect to the accuracy of the inner solves. We will see that the method fails to converge unless the inner solve tolerance τ_0 is small enough. We noted in (5.26) that the base problem solution $\mathbf{\Psi}_0$ must be sufficiently close to the perturbed solution, and that the inner tolerances have to be sufficiently small, to obtain

$$\left(\frac{s(\phi_0) + C_1\,\tau^{(i)}}{c(\phi_0) - C_1\,\tau^{(i)}}\right) \left|\frac{C_2}{\lambda_2 - \lambda_1}\right| \;<\; 1\;, \quad i = 0, 1, 2, \ldots \tag{5.36}$$

and to therefore guarantee convergence. This criterion appears to be violated for large inner tolerances τ_0.

Table 5.17 gives details for the perturbation from a base problem with pure water in the absorber region to the three cases (i) to (iii). The first column denotes the inner tolerance τ_0, which was used in the GMRES solver at each step of the algorithm and also when computing the solution $\mathbf{\Psi}_0$ of the base problem and the solution $\mathbf{\Psi}_0^*$ of its adjoint problem. We remark here that fixing the tolerance of the computation of the base problem solutions to a tolerance of 10^{-14}, did not have any influence on the convergence behaviour in our tests.

The column "outer" in Table 5.17 contains the number of outer iterations needed by the method of perturbation to converge to the desired outer tolerance $\|\mathbf{res}^{(i)}\|_2 < 10^{-14}$. A "-" indicates that the method stagnated and did not converge (the iteration was stopped after 50 outer iterations as closer investigations showed that for our problems the method either converged within 10 iterations or stagnated). The next column "inner" contains the average number of inner iterations that GMRES needed to solve the linear systems to the desired accuracy. The last column for each problem contains the average computing time in seconds which was calculated from 10 consecutive computations.

	pure absorber			10% absorber, 90% water			homogeneous material		
τ_0	outer	inner	time	outer	inner	time	outer	inner	time
10^{-1}	-	4	-	-	5	-	-	4	-
10^{-2}	-	7	-	-	8	-	-	7	-
10^{-3}	-	9	-	-	10	-	-	9	-
10^{-4}	-	11	-	-	11	-	-	11	-
10^{-5}	-	12	-	-	12	-	-	12	-
10^{-6}	-	13	-	-	13	-	-	13	-
10^{-7}	-	14	-	-	15	-	-	15	-
10^{-8}	-	15	-	3	25	55	3	25	56
10^{-9}	4	24	57	3	26	56	3	25	57
10^{-10}	4	28	58	3	28	57	3	26	57
10^{-11}	4	26	59	3	27	57	3	27	58
10^{-12}	4	26	60	3	28	58	3	28	60
10^{-13}	4	27	62	3	29	59	3	29	61
10^{-14}	4	28	62	3	29	60	3	29	61

Table 5.17: *Convergence of the inexact version of the method of perturbation in Algorithm 7 with respect to different fixed inner tolerances* τ_0 *for the linear solver.*

The table shows that the method fails to converge for large inner tolerances. For the pure absorber case the inner tolerance had to be 10^{-9} or smaller to achieve convergence. Reducing the inner tolerance further below 10^{-9} did not reduce the number of outer iterations, but it did slightly increase the number of inner iterations required by the GMRES solver, leading to longer computing times. These results appear to suggest that for large inner tolerances we can be outside the ball of convergence described in (5.36), i.e. $\left(\frac{s(\phi_0)+C_1\,\tau_0}{c(\phi_0)-C_1\,\tau_0}\right)\left|\frac{C_2}{\lambda_2-\lambda_1}\right| \geq 1$, but manage to satisfy (5.36), and therefore to regain convergence, if we decrease the inner tolerances sufficiently.

Table 5.18 now considers the convergence behaviour and computing times of Algorithm 10 for the same $\mathbf{\Psi}_0$ with respect to different fixed inner tolerances. The shifts are chosen according to (5.11).

	pure absorber			10% absorber, 90% water			homogeneous material		
τ_0	outer	inner	time	outer	inner	time	outer	inner	time
10^{-1}	5	16	58	3	17	55	3	17	55
10^{-2}	4	16	54	3	19	54	3	19	54
10^{-3}	4	17	55	3	20	53	3	20	53
10^{-4}	4	18	55	3	21	54	3	21	54
10^{-5}	4	19	54	3	22	53	3	22	53
10^{-6}	4	20	55	3	22	53	3	22	53
10^{-7}	4	21	56	3	23	55	3	23	55
10^{-8}	4	22	57	3	24	56	3	24	56
10^{-9}	4	22	58	3	25	56	3	25	57
10^{-10}	4	23	59	3	27	57	3	27	57
10^{-11}	4	24	60	3	26	58	3	26	58
10^{-12}	4	25	61	3	27	59	3	27	59
10^{-13}	4	25	63	3	28	60	3	28	61
10^{-14}	4	26	62	3	29	60	3	29	60

Table 5.18: *Convergence of Algorithm 10 with respect to different fixed inner tolerances τ_0 for the shifts $\alpha^{(i+1)}$ defined in* (5.11).

Comparing Tables 5.17 and 5.18, we observe that when Algorithm 7 converges (for the case of small inner tolerances) the number of GMRES iterations is comparable to the inner iterations needed for Algorithm 10. However, we note a considerable difference in the number of GMRES iterations between the two tables for larger inner tolerances which indicates that the work needed to solve the inner problems in Algorithm 10 is bigger than the work for the solves in Algorithm 7. The resulting additional iterations in the inner solver seem to lead to better iterates and finally convergence of the overall method already for larger tolerances.

We saw in Corollary 5.9 that for inexact solves Algorithms 10 and 7 only produce the same iterates if the inner tolerances of the method of perturbation are adjusted according to (5.17). Table 5.19 now contains the results for the inexact version of Algorithm 7 with the adjusted tolerances $\widetilde{\tau}^{(i)} := |\alpha^{(i)} - \alpha^{(i+1)}|\,\tau_0$, showing that the method of perturbation regains convergence in all cases with these adjusted tolerances.

Numerical results for the second version of the method of perturbation

We now consider the convergence of the second version of the method of perturbation given in Algorithm 12 with inexact solves. As for Algorithm 7, we encounter convergence problems if the inner tolerances are not adjusted according to (5.17).

	pure absorber			10% absorber, 90% water			homogeneous material		
τ_0	outer	inner	time	outer	inner	time	outer	inner	time
10^{-1}	4	19	55	3	22	55	3	23	55
10^{-2}	4	21	54	3	23	53	3	23	53
10^{-3}	4	22	55	3	24	53	3	24	53
10^{-4}	4	23	55	3	25	53	3	25	54
10^{-5}	4	23	54	3	26	53	3	27	53
10^{-6}	4	25	55	3	27	53	3	26	54
10^{-7}	4	25	56	3	27	54	3	27	55
10^{-8}	4	26	57	3	28	55	3	28	56
10^{-9}	4	27	58	3	28	57	3	29	57
10^{-10}	4	27	58	3	29	57	3	30	58
10^{-11}	4	27	59	3	29	57	3	29	60
10^{-12}	4	28	60	3	30	58	3	30	61
10^{-13}	4	28	61	3	30	59	3	30	61
10^{-14}	4	28	62	3	31	58	3	31	61

Table 5.19: *Convergence of the inexact version of Algorithm 7 with respect to tolerances* $\widetilde{\tau}^{(i)} := |\alpha^{(i)} - \alpha^{(i+1)}|\,\tau_0$ *using the shifts* $\alpha^{(i+1)}$ *defined in* (5.11).

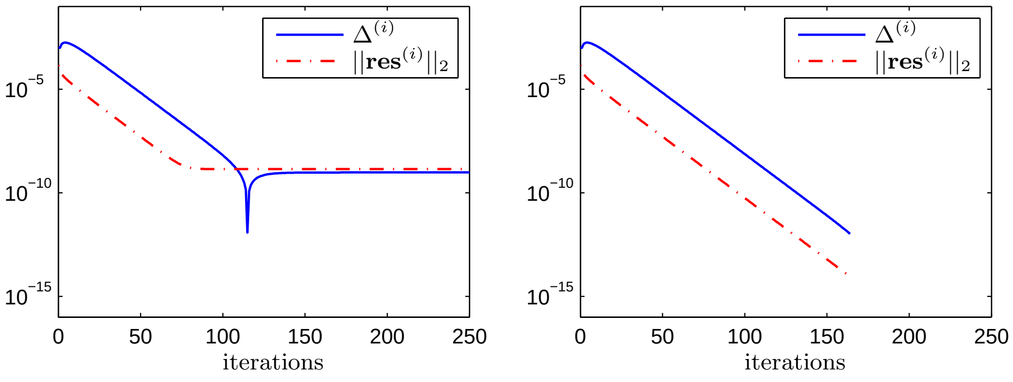
Table 5.20 gives further details of the numerical results for Algorithm 12 when using fixed tolerances $\tau^{(i)} = \tau_0$. Note that this method needs more outer iterations to converge (if it converges) than the first version of the method of perturbation.

	pure absorber			10% absorber, 90% water			homogeneous material		
τ_0	outer	inner	time	outer	inner	time	outer	inner	time
10^{-1}	-	2	-	-	2	-	-	2	-
10^{-2}	-	3	-	-	3	-	-	3	-
10^{-3}	-	3	-	-	3	-	-	3	-
10^{-4}	-	4	-	-	4	-	-	4	-
10^{-5}	-	5	-	-	5	-	-	5	-
10^{-6}	-	5	-	-	5	-	-	5	-
10^{-7}	-	6	-	-	6	-	-	6	-
10^{-8}	-	7	-	-	7	-	-	6	-
10^{-9}	-	7	-	-	7	-	-	7	-
10^{-10}	-	8	-	-	8	-	-	8	-
10^{-11}	-	9	-	164	9	182	-	8	-
10^{-12}	-	9	-	164	9	185	171	9	185
10^{-13}	171	10	201	164	10	192	171	10	191
10^{-14}	171	11	215	164	11	205	171	11	207

Table 5.20: *Convergence of Algorithm 12 with respect to different fixed tolerances* τ_0.

The results in Table 5.20 show that, compared to the other approaches considered so far, Algorithm 12 needs fewer inner iterations to solve the linear systems. However, due to the large number of outer iterations the overall computing time is longer than, for example, in the case of Algorithm 7 and the method is therefore not competitive in this deterministic case. The main advantage of this approach occurs when Monte Carlo methods are used for the solution of the inner problems since inverting the transport operator is then significantly faster than solving a source problem with the transport, scatter and fission operator on the left-hand side (see Section 4.3). This reduces the computational cost of each outer iteration and Algorithm 12 becomes more competitive.

Let us consider the convergence of Algorithm 12 for two different inner tolerances in some detail. Figure 5.21 shows how the residual and the eigenvalue error behave during the iteration process. The base problem with pure water in the control rod region is perturbed to a mix of 10% absorber material and 90% water. For the larger inner tolerance of $\tau_0 = 10^{-5}$ the norm of the outer residual stops decreasing after approximately 80 iterations. If the inner systems are solved to higher accuracy (such as $\tau_0 = 10^{-11}$), the residual norm continues to decrease until the stopping criterion is satisfied (here after 164 iterations).



(a) $\tau_0 = 10^{-5}$

(b) $\tau_0 = 10^{-11}$

Figure 5.21: *Convergence of the residual and eigenvalue error when applying Algorithm 12 for two different inner tolerances.*

We now increase the "separation" of the inner and outer iteration in Algorithm 12 by performing several solves for the inner system (5.29), before updating the eigenvalue approximation and continuing the outer iterations. This reduces the number of outer iterations as Table 5.22 shows, where the inner tolerances for all problems are set to $\tau_0 = 10^{-12}$ but the number of transport sweeps is varied.

We observe that performing between 1 and 30 transport sweeps before continuing the outer iteration results in roughly the same total number of solves for the linear system (5.29). The results show that the best computing times for the cases (i) and (iii) occur when 15−20 transport sweeps were used before updating the eigenvalue and stepping to the next outer iterate. For the problem (ii) the minimal computing time was achieved for 35 solves of (5.29) before continuing the outer iteration.

	pure absorber			10% absorber, 90% water			homogeneous material		
sweeps	outer	inner	time	outer	inner	time	outer	inner	time
1	171	10	188	164	10	182	171	10	185
2	86	10	151	82	10	141	86	10	149
3	58	10	142	55	10	132	58	10	137
4	43	10	130	41	10	127	43	10	127
5	35	10	129	33	10	122	35	10	125
10	18	10	123	17	10	114	18	10	118
15	11	10	114	12	10	116	12	10	118
20	10	10	129	9	10	117	9	10	116
25	9	10	134	7	10	114	8	10	128
30	9	10	155	6	10	117	7	10	126
35	8	10	155	5	10	112	6	10	128
40	8	10	168	5	10	123	6	10	136
45	8	10	189	5	10	129	5	10	133
50	8	10	198	5	10	138	5	10	139
100	8	10	343	5	10	220	5	10	226

Table 5.22: *Convergence of Algorithm 12 for varying numbers of "transport sweeps" and inner tolerances* $\tau_0 = 10^{-12}$ *.*

Other variations of the method of perturbation

There are several other variations of the method of perturbation that can be applied. For example, instead of moving the scatter *and* fission contribution to the right-hand side, as we did in Algorithm 12, only the fission part could be moved to the right. This would increase the work needed to invert the operator on the left-hand side compared to Algorithm 12, but it would still be cheaper than inverting the full transport, scatter and fission operator as is done in Algorithm 7.

Another variation is a perturbation method stated in [49, eq. (35), p. 194]. The resulting algorithm, using our notation, is given by Algorithm 13.

Algorithm 13 Perturbation method from [49]

```
Require: Unperturbed solutions $\lambda_0$, $\Psi_0$, $\Psi_0^*$, initial guess $\Delta\Psi^{(0)}$.
  for i=0,1,2,... do
    Compute $\Delta\lambda^{(i+1)} = \frac{\langle \Psi_0^*, (\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0 \Delta\mathcal{F})(\Psi_0 + \Delta\Psi^{(i)})\rangle}{\langle \Psi_0^*, \mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})\rangle}$ .
    Obtain $\Delta\Psi^{(i+1)}$ by solving
    $(\mathcal{T} - \mathcal{S} - \lambda_0\mathcal{F})\Delta\Psi^{(i+1)} = -(\Delta\mathcal{T} - \Delta\mathcal{S} - \lambda_0\Delta\mathcal{F})\Psi_0 + \Delta\lambda^{(i+1)}\mathcal{F}(\Psi_0 + \Delta\Psi^{(i)})$ .
    Define $\widetilde{\Psi}^{(i+1)} = \Psi_0 + \Delta\Psi^{(i+1)}$ .
    Obtain $\Psi^{(i+1)}$ by normalisation of $\widetilde{\Psi}^{(i+1)}$ .
  end for
```

We can interpret this scheme again as a way to obtain an approximate solution to (5.28) by using another variation of operator splitting, i.e. moving the term $\Delta\lambda\mathcal{F}\Delta\Psi$ to the right-hand side. As in Algorithm 12 the resulting form of "source iteration" is not performed until convergence, but the inner iteration is intertwined with the eigenvalue update in the outer iteration. Since in this method the transport, scatter and fission operators are all still present on the left-hand side, we do not expect to gain the same benefits when solving the linear systems using Monte Carlo techniques as we expected for the previous two approaches, where only the transport, or the transport and scatter, operators remained on the left-hand side.

However, for matrix-based methods this scheme provides an advantage since the operator on the left of the linear system remains fixed during the outer iteration process. This makes the computation of a good (fixed) preconditioner very attractive and should then result in comparatively cheap solutions of the linear system.

Table 5.23 now looks at the convergence behaviour of this method with respect to different tolerances for the solution of the linear system. Comparing the table to the results from Algorithm 7 in Table 5.17, we note that this method needs smaller inner tolerances to converge than the method of perturbation. However, if convergence is achieved, the computing times are similar.

	pure absorber			10% absorber, 90% water			homogeneous material		
τ_0	outer	inner	time	outer	inner	time	outer	inner	time
10^{-1}	-	4	-	-	5	-	-	4	-
10^{-2}	-	8	-	-	8	-	-	7	-
10^{-3}	-	13	-	-	14	-	-	9	-
10^{-4}	-	14	-	-	15	-	-	16	-
10^{-5}	-	15	-	-	17	-	-	17	-
10^{-6}	-	17	-	-	18	-	-	18	-
10^{-7}	-	18	-	-	19	-	-	19	-
10^{-8}	-	18	-	-	20	-	-	20	-
10^{-9}	-	19	-	-	21	-	-	21	-
10^{-10}	-	20	-	-	21	-	-	22	-
10^{-11}	-	21	-	4	22	56	3	23	54
10^{-12}	5	22	59	4	23	57	3	24	55
10^{-13}	5	23	59	4	24	58	3	25	56
10^{-14}	5	25	61	4	25	60	3	29	57

Table 5.23: *Convergence of Algorithm 13 with respect to fixed tolerances* τ_0.

Finally, we now consider a variation of the method of perturbation that aims to improve the stability of the approach. We observed in (5.12) in the proof of Lemma 5.3 that

$$\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)} \;=\; (\alpha^{(i)} - \alpha^{(i+1)})\widetilde{\Psi}_{\mathrm{II}}^{(i+1)} \;.$$

In Remark 5.5 we noted that this might lead to numerical problems as $(\alpha^{(i)} - \alpha^{(i+1)}) \to 0$ when the method converges. To avoid these problems we use, following the idea of Rüde and Schmid for inverse correction in [96], $0.95\Delta\lambda^{(i+1)}$ instead of $\Delta\lambda^{(i)}$ on the right-hand side of the linear system in Algorithm 7. This leads to $\widetilde{\Psi}_{\mathrm{IC}}^{(i+1)} = 0.05\Delta\lambda^{(i+1)}\widetilde{\Psi}_{\mathrm{II}}^{(i+1)}$ and in practice to more robust numerical results as Table 5.24 shows. The approach converges for less accurate inner solves than those in Table 5.17 where the original method of perturbation was considered, making this variation a viable computational alternative.

	pure absorber			10% absorber, 90% water			homogeneous material		
τ_0	outer	inner	time	outer	inner	time	outer	inner	time
10^{-1}	-	4	-	-	5	-	-	4	-
10^{-2}	-	7	-	4	17	53	-	7	-
10^{-3}	4	19	54	3	21	52	4	22	54
10^{-4}	4	20	56	3	22	52	3	22	53
10^{-5}	4	20	54	3	23	51	3	22	51
10^{-6}	4	21	55	3	24	51	3	23	51
10^{-7}	4	22	55	3	24	53	3	24	52
10^{-8}	4	23	55	3	25	54	3	25	54
10^{-9}	4	24	56	3	26	54	3	26	54
10^{-10}	4	25	56	3	26	54	3	26	55
10^{-11}	4	25	57	3	27	55	3	27	55
10^{-12}	4	26	58	3	28	56	3	28	56
10^{-13}	4	27	59	3	29	57	3	30	57
10^{-14}	4	29	60	3	29	58	3	29	57

Table 5.24: *Convergence of Algorithm 7 with respect to fixed tolerances* $\tau^{(i)} = \tau_0$ *and when replacing* $\Delta\lambda^{(i)}$ *on the right-hand side of the linear system with* $0.95\Delta\lambda^{(i+1)}$*.*

A final variation of the method of perturbation, that similarly aims to avoid the numerical problems arising from $(\alpha^{(i)} - \alpha^{(i+1)}) \to 0$ when the method converges, sets $\Delta\lambda^{(i)}$ on the right-hand side of the linear system to zero (instead of $0.95\Delta\lambda^{(i+1)}$). This approach resulted in very good numerical results as Table 5.25 shows.

	pure absorber			10% absorber, 90% water			homogeneous material		
τ_0	outer	inner	time	outer	inner	time	outer	inner	time
10^{-1}	9	17	64	4	18	56	-	4	-
10^{-2}	4	18	54	3	18	53	4	19	55
10^{-3}	4	20	54	3	20	53	3	20	52
10^{-4}	4	20	55	3	21	54	3	21	54
10^{-5}	4	21	54	3	22	52	3	22	52
10^{-6}	4	23	55	3	23	53	3	23	53
10^{-7}	4	23	56	3	23	54	3	25	55
10^{-8}	4	23	57	3	24	55	3	24	55
10^{-9}	4	24	59	3	25	56	3	25	56
10^{-10}	4	25	59	3	25	57	3	26	57
10^{-11}	4	26	60	3	28	58	3	27	57
10^{-12}	4	27	62	3	27	59	3	29	59
10^{-13}	4	27	63	3	28	60	3	28	60
10^{-14}	4	28	63	3	29	59	3	29	60

Table 5.25: *Convergence of Algorithm 7 with respect to different fixed tolerances* τ_0 *when using* $\Delta\lambda^{(i)} = 0$ *on the right-hand side of the linear system.*

5.6.3 Efficient GMRES implementation

We observed that Algorithm 10 with fixed inner tolerances obtained the same convergence rate as the standard shifted inverse iteration in Algorithm 5 with fixed tolerances. Due to the constant right-hand side in Algorithm 10, a computationally more efficient version of GMRES can be implemented. We now briefly describe the concept for such a method.

As many other iterative methods for solving linear systems, GMRES is based on the construction of so-called *Krylov subspaces*. Let us consider a linear problem arising from Algorithm 5, i.e.

$$(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\Psi^{(i+1)} \;=\; \mathcal{F}\Psi^{(i)} \;.$$

The Krylov subspaces that GMRES constructs are, in the simplest case,

$$\mathcal{K}_J\big((\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F}), \mathcal{F}\Psi^{(i)}\big) \;:=\; \text{span}\{\, (\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})^j \mathcal{F}\Psi^{(i)} \;,\; j = 0, \ldots, J \,\} \;.$$

Clearly, if $\Psi^{(i)}$ changes from one outer iteration to the next, then these Krylov spaces will change as well.

In the case of Algorithm 10, however, the linear systems are

$$(\mathcal{T} - \mathcal{S} - \alpha^{(i)}\mathcal{F})\Psi^{(i+1)} \;=\; \mathcal{F}\Psi_0 \tag{5.37}$$

with constant right-hand sides. Multiplying (or in the context of linear solvers, preconditioning) this problem with $(\mathcal{T} - \mathcal{S})^{-1}$ and defining $\mathcal{C} := (\mathcal{T} - \mathcal{S})^{-1}\mathcal{F}$, (5.37) becomes

$$(\mathcal{I} - \alpha^{(i)}\mathcal{C})\Psi^{(i+1)} \;=\; \mathcal{C}\Psi_0 \;. \tag{5.38}$$

Now the corresponding Krylov subspaces for (5.38) are (using analogous arguments as in the proof of Lemma 4.1 in [113])

$$\begin{aligned} \mathcal{K}_J\big((\mathcal{I} - \alpha^{(i)}\mathcal{C}), \mathcal{C}\Psi_0\big) \;&=\; \text{span}\{\, (\mathcal{I} - \alpha^{(i)}\mathcal{C})^j \mathcal{C}\Psi_0 \;,\; j = 0, \ldots, J \,\} \\ &=\; \text{span}\{\, \mathcal{C}^j \mathcal{C}\Psi_0 \;,\; j = 0, \ldots, J \,\} \;, \end{aligned}$$

which do not depend on the iteration index i and therefore do not change from one outer iteration to the next.

Hence an optimised GMRES method that makes use of these fixed Krylov spaces could be implemented and might lead to a computationally more efficient Algorithm

10 compared to the standard inexact inverse iteration given in Algorithm 5. However, further investigations are needed to ensure that the solutions of the linear systems guarantee convergence of the overall method towards the true eigenfunction and not to an eigenfunction of a problem in a lower dimensional subspace determined by the (fixed) Krylov spaces used to solve the inner problems.

Chapter 6

Conclusions

This thesis studied the criticality problem in neutron transport theory using the example of monoenergetic homogeneous model problems with isotropic scattering. A rigorous mathematical foundation was provided in the space of square-integrable functions. We showed for the model cases that, using the scalar flux and the integral form of the neutron transport equation, the unsymmetric eigenvalue problem for the angular flux is equivalent to a symmetric problem in a space of reduced dimension.

As well as proving that the integral operator in the problem of reduced dimension is self-adjoint and compact, we also established a norm estimate for it. This allowed us to show in Section 2.3 that the criticality problem is well-defined in the sense that a unique smallest positive real eigenvalue exists. In addition, we proved that the corresponding square-integrable eigenfunction is strictly positive in the interior of the reactor. The proof used the Krein-Rutman theorem and properties of positive operators on cones.

Furthermore, we discussed in Section 2.4 discretisations of the neutron transport equation, emphasising the danger of losing the underlying symmetry in the reduction to a matrix problem. Symmetry preserving discretisations were presented and the second chapter concluded with discretisation error estimates for both the solution of source problems and the eigenvalue.

Chapter 3 considered iterative methods for the solution of the criticality problem with an emphasis on the fact that the inner problems are in practice only solved inexactly. We applied four different eigenvalue solvers (power method, inverse iteration, inverse correction, and simplified Jacobi-Davidson) and gave numerical results comparing these methods for a discrete ordinates discretisation using small and large inner tolerances.

A convergence analysis for inexact shifted inverse iteration was also presented in Chapter 3. This provided conditions on the accuracy of the inner solves to guarantee convergence of the eigenvalue iteration. It is important to note that the analysis was done for the original *continuous* problem. We considered different shift strategies and suggested a non-standard Rayleigh quotient shift on $L^2(V, L^\infty(\mathbb{S}^2))$. Numerical results showed that this shift gained an additional order in the convergence rate compared to the standard Rayleigh quotient shift on the same space, provided a symmetry preserving discretisation was used. Several additional numerical tests were performed and showed good agreement with the theory, while emphasising the need for symmetry preserving discretisations to obtain optimal convergence rates for iterative eigenvalue solvers. Some experiments lay outside the regime of applicability of the theory, but the non-standard Rayleigh quotient turned out to be still useful in this case.

In Chapter 4 Monte Carlo methods were discussed and motivated by establishing links to the mathematics of the problem without the use of advanced probability theory. We considered their application to source problems with and without scatter and fission, as well as, by using a variation of the power method, to the criticality problem. For the latter we noted that the Monte Carlo approach does not need a spatial or angular mesh if the interest is only in the eigenvalue and no flux estimate is needed. Therefore, Monte Carlo results are, in this case, not affected by discretisation errors.

However, due to the statistical nature of the Monte Carlo method the results contain statistical uncertainties and reliable solutions can only be obtained if sufficiently many particles are simulated. We showed in Chapter 4 how to obtain confidence intervals and how their size depends on the number of simulations and scoring stages that are used. The numerical results of our Monte Carlo implementation showed good agreement with reference solutions. We remarked that possibly the biggest advantage of the Monte Carlo approach is the ability to model very complex geometries. On the other hand, the method has a major drawback of being slow to converge (order $n^{-1/2}$ where n is the total number of simulated neutrons). This leads to the need for simulating many particles in order to reduce the statistical uncertainties. Although the method can be easily parallelised, performing reliable computations for the difference between two similar model problems, for example, demands long computing times. Note that in this thesis we have not included discussions of the many variance reduction techniques which are vital for commercial particle transport codes. These are believed to reduce the computational cost but do not increase the order of the convergence rate.

We presented in Chapter 5 a new approach, the method of perturbation, that computes the differences in the eigenvalue and eigenfunction between two problems directly. The goal of this method is, when applying Monte Carlo techniques for the steps in Algorithm

7, to reduce the computing time compared to using Monte Carlo calculations for each of the problems individually. We provided a convergence analysis for the method of perturbation by establishing links between this new approach and variations of inverse correction and inverse iteration. The analysis allowed for inexact solves of the arising linear problem and we showed that the new method obtained, for sufficiently large fixed inner tolerances and variable shifts, the same convergence rates as standard inexact inverse iteration with an equivalent shift. We gave numerical results that supported the theory and indicated the existence of a radius of convergence that depends on the accuracy of the inner solves as well as on the solution to the base problem. We identified examples where the new approach failed to converge and recommended an adjustment to the method that resulted in more robust convergence for the numerical tests. Finally, we suggested an efficient inner solver for the linear problems in the new method and other approaches that use a fixed right-hand side.

By presenting a systematic theoretical study of inexact iterative methods for the criticality problem this thesis closes an open gap in neutron transport theory. The convergence results of the iterative methods for the continuous problem are new and provide a foundation for investigating more complicated eigenvalue solvers. The highlighted links between the Monte Carlo method and the underlying mathematics emphasise the existing connections between statistical and deterministic approaches, two areas that evolved independently and have only recently been combined in the use of automatic variance reduction (hybrid) methods (see [52] and references therein).

Finally, we suggest three possible directions for further studies that arise from this thesis.

- Further investigation of the method of perturbation, in particular in combination with the use of Monte Carlo methods. The computational and theoretical gains of the new iterative scheme, when using Monte Carlo techniques to solve for the difference between two similar problems, could be established. In addition variance reduction techniques can be considered.

- Implementation and further investigation of the efficient Krylov subspace based solver for the linear systems in the method of perturbation or fixed right-hand side iterative solvers as suggested in Section 5.6.3.

- Extension of the analysis in Chapter 2 (and subsequently of the convergence analysis for iterative methods in Chapters 3 and 5) to heterogeneous problems. A self-adjoint problem can be obtained by symmetrising the operator using techniques similar to those applied in [21].

Bibliography

[1] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, Dover, New York, 1970.

[2] M. L. Adams, *"I have an idea!" An appreciation of Edward W. Larsen's contributions to particle transport*, Annals of Nuclear Energy, 31 (2004), pp. 1963–1986.

[3] M. L. Adams and E. W. Larsen, *Fast iterative methods for discrete-ordinates particle transport calculations*, Progress in Nuclear Energy, 40 (2002), pp. 3–159.

[4] E. J. Allen and R. M. Berry, *The inverse power method for calculation of multiplication factors*, Annals of Nuclear Energy, 29 (2002), pp. 929–935.

[5] M. Asadzadeh, *Analysis of a fully discrete scheme for neutron transport in two-dimensional geometry*, SIAM Journal on Numerical Analysis, 23 (1986), pp. 543–561.

[6] ——, *L_p and eigenvalue error estimates for the discrete ordinates methods for two-dimensional neutron transport*, SIAM Journal on Numerical Analysis., 26 (1989), pp. 66–87.

[7] S. F. Ashby, P. N. Brown, M. R. Dorr, and A. C. Hindmarsh, *A linear algebraic analysis of diffusion synthetic acceleration for the Boltzmann transport equation*, SIAM Journal on Numerical Analysis, 32 (1995), pp. 128–178.

[8] J. R. Askew, *A characteristics formulation to the neutron transport equation in complicated geometries*, tech. report, UKAEA, Winfrith, 1972.

[9] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. A. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, 2000.

[10] D. J. Baker, K. A. Cliffe, P. Houston, and P. N. Smith, *Development and convergence of the long characteristic method for the two-dimensional neutron transport equation.* in preparation.

[11] C. L. Barrett and E. W. Larsen, *A variationally-based variance reduction method for Monte Carlo neutron transport calculations*, Annals of Nuclear Energy, 28 (2001), pp. 457–475.

[12] G. I. Bell and S. Glasstone, *Nuclear Reactor Theory*, Reinhold, 1970.

[13] J. Berns-Müller, *Inexact Inverse Iteration using Galerkin Krylov Solvers*, PhD thesis, Department of Mathematics, University of Bath, UK, 2003.

[14] J. Berns-Müller, I. G. Graham, and A. Spence, *Inexact inverse iteration for symmetric matrices*, Linear Algebra and its Applications, 416 (2006), pp. 389–413.

[15] J. Berns-Müller and A. Spence, *Inexact inverse iteration and GMRES*, tech. report, University of Bath, 2005.

[16] ——, *Inexact inverse iteration with variable shift for nonsymmetric generalized eigenvalue problems*, SIAM Journal on Matrix Analysis and Applications, 28 (2006), pp. 1069–1082.

[17] R. Brissenden and A. Garlick, *Biases in the estimation of k_{eff} and its error by Monte Carlo methods*, Annals of Nuclear Energy, 13 (1986), pp. 63–83.

[18] F. Brown, *Fundamentals of Monte Carlo particle transport*, tech. report, Los Alamos National Laboratory, 2005. LA-UR-05-4983.

[19] P. N. Brown, *A linear algebraic development of diffusion synthetic acceleration for three-dimensional transport equations*, SIAM Journal on Numerical Analysis, 32 (1995), pp. 179–214.

[20] K. M. Case and P. F. Zweifel, *Linear Transport Theory*, Addison-Wesley, 1967.

[21] B. Chang, *The conjugate gradient method solves the neutron transport equation h-optimally*, Numerical Linear Algebra with Applications, 14 (2007), pp. 751–769.

[22] G. Chiba and K. Numata, *Neutron transport benchmark problem proposal for fast critical assembly without homogenizations*, Annals of Nuclear Energy, 34 (2007), pp. 443–448.

[23] J. L. Conlin, *Explicitly Restarted Arnoldi's Method for Monte Carlo Nuclear Criticality Calculations*, PhD thesis, Nuclear Engineering and Radiological Sciences, University of Michigan, USA, 2009.

[24] B. Davison and J. B. Sykes, *Neutron Transport Theory*, Clarendon Press, Oxford, 1957.

[25] G. C. de Buffon, *Essai d'arithmétique morale*, Supplément à l'Histoire Naturelle, 4 (1777), pp. 46–123.

[26] K. Deimling, *Nonlinear Functional Analysis*, Springer–Verlag, 1985.

[27] J. D. Densmore and E. W. Larsen, *Variational variance reduction for particle transport eigenvalue calculations using Monte Carlo adjoint simulation*, Journal of Computational Physics, 192 (2003), pp. 387–405.

[28] J. J. Duderstadt and L. J. Hamilton, *Nuclear Reactor Analysis*, John Wiley & Sons, New York, 1976.

[29] J. J. Duderstadt and W. R. Martin, *Transport Theory*, John Wiley & Sons, New York, 1979.

[30] N. Dunford and J. T. Schwartz, *Linear Operators – Part II: Spectral Theory*, Wiley-Interscience, 1963.

[31] S. A. Dupree and S. K. Fraley, *A Monte Carlo Primer: A Practical Approach to Radiation Transport*, Kluwer Academic, New York, 2002.

[32] R. Eckhardt, *Stan Ulam, John von Neumann and the Monte Carlo method*, Los Alamos Science Special Issue, (1987), pp. 131–136.

[33] A. Erdélyi, W. Magnus, F. Oberhettinger, and F. G. Tricomi, *Tables of Integral Transforms (Bateman Manuscript Project)*, McGraw-Hill, 1954.

[34] V. Faber and T. A. Manteuffel, *A look at transport theory from the point of view of linear algebra*, in Transport Theory, Invariant Imbedding, and Integral Equations: Proceeding in Honor of G.M. Wing's 65th Birthday, vol. 115 of Lecture Notes in Pure and Applied Mathematics, 1989, pp. 37–61.

[35] W. Feller, *An Introduction to Probability Theory and its Applications*, Wiley, New York, Chichester, 1968.

[36] M. A. Freitag, *Inner-outer Iterative Methods for Eigenvalue Problems – Convergence and Preconditioning*, PhD thesis, Department of Mathematics, University of Bath, UK, 2007.

[37] M. A. Freitag and A. Spence, *Convergence of inexact inverse iteration with application to preconditioned iterative solves*, BIT Numerical Mathematics, 47 (2007), pp. 27–44.

[38] ——, *Convergence theory for inexact inverse iteration applied to the generalised nonsymmetric eigenproblem*, Electronic Transactions on Numerical Analysis, 28 (2007), pp. 40–64.

[39] ——, *A tuned preconditioner for inexact inverse iteration applied to Hermitian eigenvalue problems*, IMA Journal of Numerical Analysis, 28 (2008), pp. 522–551.

[40] ——, *Rayleigh quotient iteration and simplified Jacobi-Davidson method with preconditioned iterative solves*, Linear Algebra and its Applications, 428 (2008), pp. 2049–2060.

[41] M. A. Freitag, A. Spence, and E. Vainikko, *Rayleigh quotient iteration and simplified Jacobi-Davidson with preconditioned iterative solves for generalised eigenvalue problems*, tech. report, Department of Mathematics, University of Bath, 2008.

[42] G. Frosali, C. van der Mee, and V. Protopopescu, *Transport equations with boundary conditions of reverse reflection type*, Mathematical Methods in the Applied Sciences, 10 (1988), pp. 15–35.

[43] S. Glasstone and M. C. Edlund, *The Elements of Nuclear Reactor Theory*, Van Nostrand Company, Inc, 1952.

[44] G. H. Golub and H. A. van der Vorst, *Eigenvalue computation in the 20th century*, Journal of Computational and Applied Mathematics, 123 (2000), pp. 35–65.

[45] G. H. Golub and C. F. van Loan, *Matrix Computations*, Johns Hopkins University Press, 1996.

[46] G. H. Golub and Q. Ye, *Inexact inverse iteration for generalized eigenvalue problems*, BIT Numerical Mathematics, 40 (2000), pp. 671–684.

[47] I. G. Graham and I. H. Sloan, *On the compactness of certain integral operators*, Journal of Mathematical Analysis and Applications, 68 (1979), pp. 580–594.

[48] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, 1997.

[49] E. Greenspan, *Developments in perturbation theory*, Advances in Nuclear Science and Technology, 9 (1976), pp. 181–268.

[50] G. Grimmett and D. Stirzaker, *Probability and Random Processes*, Oxford University Press, 1992.

[51] A. Gupta and R. S. Modak, *Krylov sub-space methods for k-eigenvalue problem in 3-D neutron transport*, Annals of Nuclear Energy, 31 (2004), pp. 2113 – 2125.

[52] A. Haghighat and J. C. Wagner, *Monte Carlo variance reduction with deterministic importance functions*, Progress in Nuclear Energy, 42 (2003), pp. 25–53.

[53] S. Hamilton, M. Benzi, and J. Warsa, *Negative flux fixups in discontinuous finite element S_N transport*, American Nuclear Society, LaGrange Park, Illinois, USA, 2009. International Conference on Mathematics, Computational Methods & Reactor Physics, on CD-ROM.

[54] G. E. Hansen and C. Maier, *Perturbation theory of reactivity coefficients for fast-neutron critical systems*, Nuclear Science and Engineering, 8 (1960), pp. 532–542.

[55] E. Hewitt and K. A. Ross, *Abstract Harmonic Analysis*, Springer–Verlag, 1979.

[56] H. Hochstadt, *Integral Equations*, Wiley-Interscience, 1973.

[57] M. Hochstenbach, *A Jacobi-Davidson type method for the generalized singular value problem*, Linear Algebra and its Applications, 431 (2009), pp. 471–487.

[58] M. E. Hochstenbach and Y. Notay, *The Jacobi-Davidson method*, in GAMM Mitteilungen, vol. 29, 2006, pp. 368–382.

[59] L. J. Hutton and N. R. Smith, *Use of a hybrid Monte Carlo technique for power shape calculations*, in Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications, Proceedings of the Monte Carlo 2000 Conference, Lisbon, A. Kling et al., ed., 2001, pp. 697–702.

[60] I. C. F. Ipsen, *A history of inverse iteration*, vol. 2 of Helmut Wielandt, Mathematische Werke, Mathematical Works, Walter de Gruyter, Berlin, 1996, pp. 464–472.

[61] ——, *Computing an eigenvector with inverse iteration*, SIAM Review, 39 (1997), pp. 254–291.

[62] T. Jevremovic, J. Vujic, and K. Tsuda, *Anemona – a neutron transport code for general geometry reactor assemblies based on the method of characteristics and R-function solid modeler*, Annals of Nuclear Energy, 28 (2001), pp. 125–152.

[63] C. Johnson and J. Pitkäranta, *Convergence of a fully discrete scheme for two-dimensional neutron transport*, SIAM Journal on Numerical Analysis, 20 (1983), pp. 951–966.

[64] M. H. Kalos and P. A. Whitlock, *Monte Carlo Methods*, John Wiley & Sons, 1986.

[65] L. V. Kantorovich and G. P. Akilov, *Functional Analysis*, Pergamon Press, 1982.

[66] H. G. Kaper and R. B. Kellogg, *Continuity and differentiability properties of the solution of the linear transport equation*, SIAM Journal on Applied Mathematics, 32 (1977), pp. 201–214.

[67] B. Kirk, *Overview of Monte Carlo radiation transport codes*, Radiation Measurements, (2010). doi: 10.1016/j.radmeas.2010.05.037.

[68] A. Klenke, *Probability Theory – A Comprehensive Course*, Springer, 2008.

[69] M. A. Krasnosel'skij, *Positive Solutions of Operator Equations*, Noordhoff Ltd., Groningen, 1964.

[70] Y. Lai, K. Lin, and W. Lin, *An inexact inverse iteration for large sparse eigenvalue problems*, Numerical Linear Algebra with Applications, 4 (1997), pp. 425–437.

[71] B. Lapeyre, E. Pardoux, and R. Sentis, *Introduction to Monte Carlo Methods for Transport and Diffusion Equations*, Oxford University Press, 2003.

[72] P.-S. Laplace, *Theorie analytique des probabilités, livre 2*, Oeuvres complètes de Laplace, 7 (1886).

[73] E. W. Larsen, *Diffusion theory as an asymptotic limit of transport theory for nearly critical systems with small mean free paths*, Annals of Nuclear Energy, 7 (1980), pp. 249–255.

[74] ——, *Final report: Hybrid Monte Carlo–deterministic methods for nuclear reactor-related criticality calculations*, tech. report, University of Michigan, USA, 2003. Department of Energy (DOE) Project: DE-FG-00ID13920.

[75] ——, *An overview of neutron transport problems and simulation techniques*, in Computational Methods in Transport, F. Graziani, ed., vol. 48 of Lecture Notes in Computational Science and Engineering, Springer Berlin Heidelberg, 2006, pp. 513–534.

[76] E. W. Larsen and W. F. Miller, Jr., *Convergence rates of spatial difference equations for the discrete-ordinates neutron transport equations in slab geometry*, Nuclear Science and Engineering, 73 (1980), pp. 76–83.

[77] E. W. Larsen and P. Nelson, *Finite-difference approximations and superconvergence for the discrete-ordinate equations in slab geometry*, SIAM Journal on Numerical Analysis, 19 (1982), pp. 334–348.

[78] D. Lathouwers, *Iterative computation of time-eigenvalues of the neutron transport equation*, Annals of Nuclear Energy, 30 (2003), pp. 1793–1806.

[79] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *Arpack User's Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods (Software, Environments, Tools)*, SIAM, 1998.

[80] E. E. Lewis and W. F. Miller, Jr., *Computational Methods of Neutron Transport*, John Wiley & Sons, New York, 1984.

[81] J. Lieberoth, *A Monte Carlo technique to solve the static eigenvalue problem of the Boltzmann transport equation*, Nukleonik, (1968), pp. 213–219.

[82] T. A. Manteuffel and K. J. Ressel, *Least-squares finite-element solution of the neutron transport equation in diffusive regimes*, SIAM Journal on Numerical Analysis., 35 (1998), pp. 806–835.

[83] G. Marchuk and V. Lebedev, *Numerical Methods in the Theory of Neutron Transport*, Harwood Academic, New York, 1986.

[84] M. Mendelson, *Monte Carlo criticality calculations for thermal reactors*, Nuclear Science and Engineering, 32 (1968), pp. 319–331.

[85] N. Metropolis and S. Ulam, *The Monte Carlo method*, Journal of the American Statistical Association, 44 (1949), pp. 335–341.

[86] J. Mika, *Existence and uniqueness of the solution to the critical problem in neutron transport theory*, Studia Mathematica, 37 (1971), pp. 213–225.

[87] R. S. Modak and A. Gupta, *New applications of Orthomin(1) algorithm for k-eigenvalue problem in reactor physics*, Annals of Nuclear Energy, 33 (2006), pp. 538–543.

[88] M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics*, Oxford, Clarendon Press, 1999.

[89] G. Orengo, M. T. M. B. de Vilhena, C. O. Graça, A. D. Caldeira, and G. A. Gonçalves, *Recent advances in the LTS_N method for criticality calculations in slab geometry*, Annals of Nuclear Energy, 31 (2004), pp. 2195–2202.

[90] B. N. Parlett, *The Rayleigh quotient iteration and some generalizations for nonnormal matrices*, Mathematics of Computation, 28 (1974), pp. 679–693.

[91] ——, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, 1980.

[92] J. Pitkäranta, *On the spatial differencing of the discrete ordinate neutron transport equation*, SIAM Journal on Numerical Analysis, 15 (1978), pp. 859–869.

[93] J. Pitkäranta and L. R. Scott, *Error estimates for the combined spatial and angular approximations of the transport equation in slab geometry*, SIAM Journal on Numerical Analysis, 20 (1983), pp. 922–950.

[94] M. Renardy and R. C. Rogers, *An Introduction to Partial Differential Equations*, Springer, New York, London, 2003.

[95] M. Robbé, M. Sadkane, and A. Spence, *Inexact inverse subspace iteration with preconditioning applied to non-Hermitian eigenvalue problems*, SIAM Journal on Matrix Analysis and Applications, 31 (2009), pp. 92–113.

[96] U. Rüde and W. Schmid, *Inverse multigrid correction for generalized eigenvalue computations*, tech. report, Report No. 338, Universität Augsburg, Germany, 1995.

[97] A. Ruhe and T. Wiberg, *The method of conjugate gradients used in inverse iteration*, BIT Numerical Mathematics, 12 (1972), pp. 543–554.

[98] B. P. Rynne and M. A. Youngson, *Linear Functional Analysis*, Springer Undergraduate Mathematics Series. Springer, London, 2008.

[99] Y. Saad, *Numerical Methods for Large Eigenvalue Problems (2nd edition)*, Manchester University Press, 1992.

[100] ——, *Iterative Methods for Sparse Linear Systems (1st edition)*, PWS Publishing Company, Boston, 1996.

[101] D. C. Sahni and N. G. Sjöstrand, *Criticality and time eigenvalues in one-speed neutron transport*, Progress in Nuclear Energy, 23 (1990), pp. 241–289.

[102] R. Sanchez, L. Mao, and S. Santandrea, *Treatment of boundary conditions in trajectory-based deterministic transport methods*, Nuclear Science and Engineering, 140 (2002), pp. 23–50.

[103] R. Scheichl, *Parallel Solution of the Transient Neutron Diffusion Equations with Multi-Grid and Preconditioned Krylov-Subspace Methods*, diploma thesis, Johannes Kepler Universität Linz, 1997.

[104] V. Simoncini and L. Eldén, *Inexact Rayleigh quotient-type methods for eigenvalue computations*, BIT Numerical Mathematics, 42 (2002), pp. 159–182.

[105] G. L. G. Sleijpen, A. G. L. Booten, D. R. Fokkema, and H. A. van der Vorst, *Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems*, BIT Numerical Mathematics, 36 (1996), pp. 595–633.

[106] G. L. G. Sleijpen and H. A. van der Vorst, *A Jacobi-Davidson iteration method for linear eigenvalue problems*, SIAM Journal on Matrix Analysis and Applications, 17 (1996), pp. 401–425.

[107] P. Smit and M. H. C. Paardekooper, *The effects of inexact solvers in algorithms for symmetric eigenvalue problems*, Linear Algebra and its Applications, 287 (1999), pp. 337–357.

[108] I. M. Sobol, *A Primer for the Monte Carlo Method*, CRC Press, 1994.

[109] A. Sood, R. A. Forster, and D. K. Parsons, *Analytical benchmark test set for criticality code verification*, tech. report, Los Alamos National Laboratory, USA, 1999. LA-13511.

[110] ——, *Analytical benchmark test set for criticality code verification*, Progress in Nuclear Energy, 42 (2003), pp. 55–106.

[111] J. Spanier and E. M. Gelbard, *Monte Carlo Principles and Neutron Transport Problems*, Dover Publications, 1968, reprint 2008.

[112] W. M. Stacey, *Nuclear Reactor Physics*, Wiley, 2007.

[113] A. Stathopoulos and Y. Saad, *Restarting techniques for the (Jacobi-) Davidson symmetric eigenvalue methods*, Electronic Transactions on Numerical Analysis, 7 (1998), pp. 163–181.

[114] G. W. Stewart, *Matrix Algorithms – Volume 2: Eigensystems*, SIAM, 2001.

[115] J. B. Taylor, *The Development of a Three-Dimensional Nuclear Reactor Kinetics Methodology Based on the Method of Characteristics*, PhD thesis, Graduate School College of Engineering, Pennsylvania State University, USA, 2008.

[116] J. F. Toland, *Self-adjoint operators and cones*, Journal of the London Mathematical Society, 53 (1996), pp. 167–183.

[117] T. J. Urbatsch, *Iterative Acceleration Methods for Monte Carlo and Deterministic Criticality Calculations*, PhD thesis, Nuclear Engineering and Scientific Computing Department, University of Michigan, USA, 1995.

[118] V. S. Vladimirov, *Mathematical Problems in the One-velocity Theory of Particle Transport*, Atomic Energy of Canada Limited, Chalk River, Ontario, Canada, 1963. Translated from "Transactions of the V. A. Steklov Mathematical Institute, 61, 1961".

[119] J. Wagner, A. Haghighat, B. Petrovic, and H. Hanshaw, *Benchmarking of synthesized 3D S_N transport methods for pressure vessel fluence calculations with Monte Carlo*, in International Conference on Mathematics and Computations, Reactor Physics and Environmental Analysis, 1995, pp. 1214–1222.

[120] J. S. Warsa, T. A. Wareing, J. E. Morel, J. M. McGhee, and R. B. Lehoucq, *Krylov subspace iterations for the calculation of k-eigenvalues with S_N transport codes*, in Nuclear Mathematical and Computational Sciences: A Century in Review, A Century Anew, American Nuclear Society, LaGrange Park, IL, USA, 2003. on CD-ROM.

[121] H. Wielandt, *Beiträge zur mathematischen Behandlung komplexer Eigenwertprobleme, Teil V: Bestimmung höherer Eigenwerte durch gebrochene Iteration*, tech. report, Bericht B 44/J/37 Aerodynamische Versuchsanstalt Göttingen, Germany, 1944.

[122] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.

[123] E. Woodcock, T. Murphy, P. Hemmings, and S. Longworth, *Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry*, Proceeding of the Conference for Applications of Computing Methods to Reactor Problems, Argonne National Laboratory, 1965, pp. 557–579.

[124] F. Xue and H. C. Elman, *Fast inexact subspace iteration for generalized eigenvalue problems with spectral transformation*, Linear Algebra and its Applications, (2010). doi:10.1016/j.laa.2010.06.021.

[125] T. Yamamoto and Y. Miyoshi, *Reliable method for fission source convergence of Monte Carlo criticality calculation with Wielandt's method*, Journal of Nuclear Science and Technology, 41 (2004).

[126] S. Yun and N. Z. Cho, *Acceleration of source convergence in Monte Carlo k-eigenvalue problems via anchoring with a p-CMFD deterministic method*, Annals of Nuclear Energy, 37 (2010), pp. 1649–1658.

[127] L. Y. Zaslavsky, *An adaptive algebraic multigrid for reactor criticality calculations*, SIAM Journal on Scientific Computing, 16 (1995), pp. 840–847.