MA50177: Scientific Computing
Case Study

Nuclear Reactor Simulation – Generalised Eigenvalue Problems

Introduction

While people are still looking for (energy and cost) efficient ways to exploit renewable energies (solar, aeolian, fuel cells, etc.), nuclear power stations will continue to play a major role as an energy source in the world. Since it is not possible to test safety procedures in the case of a malfunction or of an accident experimentally, the numerical simulation of nuclear reactors is of utmost importance.

Physical Background

The simulation of the reactor core is usually decoupled into (i) a thermal-hydraulics model treating the heat flux in the reactor coolant and (ii) a neutron balance equation describing the nuclear fission in the fuel rods. As neutrons are both the initiator and the result of nuclear fissions, they are a good local measure of the number of fissions taking place. The standard model to describe the neutron balance in a nuclear reactor are the two-group neutron diffusion equations

$$\left.\begin{array}{rcl} -\mathrm{div}\,(K_1 \nabla u_1) \;+\; (\Sigma_{a,1} + \Sigma_s)\, u_1 & = & \dfrac{1}{\lambda_0}\,(\Sigma_{f,1}\, u_1 + \Sigma_{f,2}\, u_2) \\ -\mathrm{div}\,(K_2 \nabla u_2) \;+\; \Sigma_{a,2}\, u_1 \;-\; \Sigma_s\, u_2 & = & 0 \end{array}\right\} \quad \text{for } \boldsymbol{x} \in \Omega \qquad (1)$$

where $\mathrm{div}\,(K\nabla u) := \frac{\partial}{\partial x_1}\left(K\frac{\partial u}{\partial x_1}\right) + \frac{\partial}{\partial x_2}\left(K\frac{\partial u}{\partial x_2}\right)$ in 2D. In this form, (1) is an eigenvalue problem. The parameter λ_0 has to be found such that the resulting system of PDEs in (1) has a nontrivial solution u_1, u_2.

The two-group neutron diffusion equations describe the conservation of the total number of free neutrons in the reactor core. The neutrons are divided into *fast* neutrons which initiate fissions because of their high kinetic energy (energy group 1) and into *thermic* neutrons (energy group 2). The factors that play a role in the construction of the neutron balance are

- *diffusion*,
- *absorption* of a neutron by an atomic nucleus,
- *scattering* of a neutron from one energy group into another, and
- *fissions* which represent the source of free neutrons.

The functions $\Sigma_{a,1}$, $\Sigma_{a,2}$, Σ_s, $\Sigma_{f,1}$, and $\Sigma_{f,2}$ are so-called *cross sections* and they measure the probablitity of the above interactions taking place. They will take different values in different regions of the reactor. $\Sigma_{a,1}$ and $\Sigma_{a,2}$ are the absorption cross sections for each energy group; Σ_s is the scattering cross section from Group 1 to Group 2; and

$\Sigma_{f,1}$ and $\Sigma_{f,2}$ are the fission cross sections for each group. The functions K_1 and K_2 are the neutron diffusion coefficients.

The largest eigenvalue λ_0 for (1) is very important. It represents the multiplication constant, i.e.

$$\lambda_0 := \frac{\text{neutron production}}{\text{neutron loss}} ,$$

and is a measure of the criticality of a reactor. The corresponding eigenfunctions $u_1(\boldsymbol{x})$ and $u_2(\boldsymbol{x})$ define the neutron flux density distributions of the fast and of the thermic neutrons respectively. A reactor is called: *subcritical* if $\lambda_0 < 1$ and the chain reaction diminishes, it is called *critical* if $\lambda_0 = 1$ and the chain reaction is constant, and it is called *supercritical* if $\lambda_0 > 1$ and the chain reaction is growing. The aim is to maintain the reactor in the critical phase.

The Generalised Eigenvalue Problem

Let us assume for simplicity that the domain $\Omega = [0,1] \times [0,1]$. Then we can discretise this problem as in Example 4.4 or in Question 5 on Problem Sheet 5 using a finite difference approximation to obtain the following $2N \times 2N$ (discrete) problem

$$\begin{pmatrix} A_1 & 0 \\ S & A_2 \end{pmatrix} \begin{pmatrix} \boldsymbol{U}_1 \\ \boldsymbol{U}_2 \end{pmatrix} = \frac{1}{\lambda} \begin{pmatrix} F_1 & F_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{U}_1 \\ \boldsymbol{U}_2 \end{pmatrix} \tag{2}$$

where $N = m^2$, $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ are approximations to u_1 and u_2 at points (ih, jh), $i, j = 1, \ldots, m$, and λ is an approximation of λ_0 in (1). (2) can be written in the more abstract form

$$A\boldsymbol{U} = \rho F \boldsymbol{U} \tag{3}$$

where A, F are the $2N \times 2N$ matrices defined in (2) and $\rho := 1/\lambda$. The matrix A will in general be non-symmetric, but non-singular. The matrix F on the other hand is obviously singular.

The problem of finding a pair $(\rho, \boldsymbol{U})$ with $\boldsymbol{U} \neq \boldsymbol{0}$ which satisfy (3) is called a *generalised eigenvalue problem*. (Recall that for $F = I$ this reduces to a standard eigenvalue problem.) The smallest possible positive value of ρ corresponds to the largest possible value of λ which determines the criticality of the reactor.

Theorem 1 *There exists a unique positive eigenvector $\boldsymbol{U}^* > 0$ for problem (3) with real positive eigenvalue ρ^* smaller than the modulus of all other eigenvalues of (3).*

PROOF. This is given in Wachspress [2, Theorem 3-2].

Inexact Inverse Iteration

A classical method to find the eigenpair $(\rho^*, \boldsymbol{U}^*)$ for (3) is *inverse iteration* (This is very similar to inverse iteration for the standard eigenvalue problem; see lectures). In Figure 1 we state the inverse iteration algorithm for the abstract generalised eigenvalue problem (3). At each iteration k we get an approximation $(\rho_k, \boldsymbol{U}_k)$ for the eigenvalue

Inverse Iteration:

Choose an initial vector $\boldsymbol{U}_0$, with $\|\boldsymbol{U}_0\|_2 = 1$, an initial shift $\sigma_0 \in \mathbb{R}$, and a maximal number of iterations `maxit`.

`do k=0,maxit`

Solve $(A - \sigma_k F)\,\boldsymbol{U}_{k+1} = (F\boldsymbol{U}_k)$ (Solve shifted system for $\boldsymbol{U}_{k+1}$)

$$\boldsymbol{U}_{k+1} = \frac{\boldsymbol{U}_{k+1}}{\|\boldsymbol{U}_{k+1}\|_2} \qquad \text{(Normalise eigenvector)}$$

$$\rho_{k+1} = \frac{\boldsymbol{U}_{k+1}^T A \boldsymbol{U}_{k+1}}{\boldsymbol{U}_{k+1}^T F \boldsymbol{U}_{k+1}} \qquad \text{(Rayleigh quotient)}$$

$$\boldsymbol{r}_{k+1} = (A - \rho_{k+1} F)\boldsymbol{U}_{k+1} \qquad \text{(Calculate residual)}$$

`if` ($\|\boldsymbol{r}_{k+1}\|_2 \leq \varepsilon$) `exit` (Stopping criterion)

`end do`

Figure 1: Algorithm for Inverse Iteration

and the eigenvector. And we stop the iteration when the norm of the eigenvalue residual $\boldsymbol{r}_{k+1}$ is smaller than a predefined tolerance ε.

Inexact inverse iteration now simply means that we do not solve the linear equation systems

$$(A - \sigma_k F)\,\boldsymbol{U}_{k+1} \;=\; (F\boldsymbol{U}_k) \tag{4}$$

exactly at each (outer) iteration k, but that we use an (inner) iterative method, like Jacobi, Conjugate Gradients, or GMRES. Thus at each (outer) iteration k, given an intial guess $\boldsymbol{U}_{k+1}^{(0)} \in \mathbb{R}^{2N}$ our *inner iteration* will produce a sequence of approximations $\boldsymbol{U}_{k+1}^{(i)}$, $i = 1, 2, \ldots$, to the exact solution $\boldsymbol{U}_{k+1}$ of (4). We stop this inner iteration when the norm of the (inner) residual

$$\boldsymbol{r}_{inner}^{(i)} = \Big((F\boldsymbol{U}_k) - (A - \sigma_k F)\boldsymbol{U}_{k+1}^{(i)}\Big)$$

is smaller than a chosen tolerance τ_k, i.e.

$$\|\boldsymbol{r}_{inner}^{(i)}\|_2 \;\leq\; \tau_k * \|\boldsymbol{r}_{inner}^{(0)}\|_2 \;. \tag{5}$$

The choice of this tolerance τ_k and of the shift σ_k is crucial for the convergence of inexact inverse iteration.

Definition of a Model Problem

In Assignment 2 we will look at the model problem with geometry depicted in Figure 2. Note that using symmetry in the reactor we only model a quarter of the core as shown in Figure 2. The functions K_1, K_2, $\Sigma_{a,1}$, $\Sigma_{a,2}$, Σ_s, $\Sigma_{f,1}$, and $\Sigma_{f,2}$ are piecewise

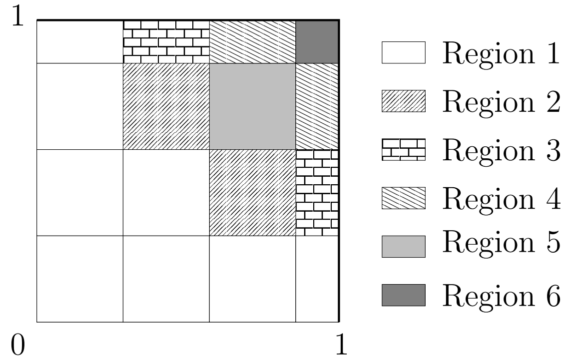

Figure 2: Model problem geometry.

	K_1	K_2	$\Sigma_{a,1}$	$\Sigma_{a,2}$	Σ_s	$\Sigma_{f,1}$	$\Sigma_{f,2}$
Region 1	$2.939 * 10^{-5}$	$1.306 * 10^{-5}$	0.0089	0.109	0.0	0.0	0.0079
Region 2	$4.245 * 10^{-5}$	$1.306 * 10^{-5}$	0.0105	0.025	0.0	0.0	0.0222
Region 3	$4.359 * 10^{-5}$	$1.394 * 10^{-5}$	0.0092	0.093	0.0066	0.140	0.0156
Region 4	$4.395 * 10^{-5}$	$1.355 * 10^{-5}$	0.0091	0.083	0.0057	0.109	0.0159
Region 5	$4.398 * 10^{-5}$	$1.355 * 10^{-5}$	0.0097	0.098	0.0066	0.124	0.0151
Region 6	$4.415 * 10^{-5}$	$1.345 * 10^{-5}$	0.0093	0.085	0.0057	0.107	0.0157

Table 1: Data for the model problem.

constant. Their (constant) values in each region are given in Table 1. Moreover we assume (as in Question 5 on Problem Sheet 5 again) that for $g = 1, 2$ we have the following boundary conditions

$$\begin{aligned} u_g &= 0 && \text{if } x_1 = 0 \text{ or } x_2 = 0, \\ K_g \frac{\partial u_g}{\partial x_i} &= 0 && \text{if } x_i = 1, \text{ for } i = 1, 2. \end{aligned} \tag{6}$$

After subdivision of Ω into $N := m^2$ equal squares of size $h \times h$, where $h := 1/m$, the finite difference approximations of each one of the equations in (1) for this model problem are calculated exactly like in Question 5 on Problem Sheet 5. The $N \times N$ matrices A_1 and A_2 in (2) are of the exact same form as the matrix A there. The matrices S, F_1 and F_2 are diagonal matrices:

$$S := \begin{pmatrix} [\Sigma_s]_1 & & \\ & \ddots & \\ & & [\Sigma_s]_N \end{pmatrix}, \qquad F_g := \begin{pmatrix} [\Sigma_{f,g}]_1 & & \\ & \ddots & \\ & & [\Sigma_{f,g}]_N \end{pmatrix}, \quad \text{for } g = 1, 2.$$

Note that routines which assemble these matrices and the matrices A and F in (3) will be provided.

References

[1] R. Scheichl, *Parallel solution of the multigroup neutron diffusion equations with multigrid and preconditioned Krylov methods*, Masters Thesis, University Linz, Austria (1997). [`http://www.maths.bath.ac.uk/~masrs/publications.html`]

[2] E.L. Wachspress, *Iterative Solution of Elliptic Systems And Applications to the Neutron Diffusion Equations of Reactor Physics*, Prentice Hall, Englewood Cliffs, N.J., 1966 (512.978 WAC, two copies on 28 day loan).