Notes

1 Conjugate Gradient Method

1.1 Overview

The conjugate gradient method (cgm) is an algorithm used to solve a linear system of the form

$$Ax = b \tag{1}$$

Where A is a symmetric ($A^T = A$) positive definite ($x^T Ax > 0$) $n \times n$ matrix, $x,\ b$ vectors.
The algorithm is iterative, starting from a guess solution x_0 and taking a step towards the solution at each cycle.
The search directions are calculated from the residual term, defined as $r_i = b - Ax_i$.
It is possible to prove that by choosing the step direction to be A-orthogonal to all the previous ones, the solution converges the fastest (i.e. the error term $\|e_i\| = \|x_i - x\|$ is minimized).

1.2 Steepest descent

A simpler algorithm is the steepest descent.
The idea is to take a step in the direction of the residual so that the quadratic form is minimized.

$$x_{i+1} = x_i + \alpha_i r_i \tag{2}$$

$$\alpha_i \text{ such that } \frac{df(x_{i+1})}{d\alpha_i} = 0 \implies \alpha_i = \frac{r_i^T r_i}{r_i^T A r_i} \tag{3}$$

This method is inefficient as x_i often finds itself oscillating around the solution, since the search directions explore non-disjoint subspaces.

1.3 The algorithm

A better alternative is to set the search direction to be A-orthogonal to the error at the next iteration. If this is the case, it can be proven that the components of the error term are reduced to zero at each iteration, implying a convergence to the exact solution in n steps.

$$d_i^T A e_{i+1} = 0 \implies \frac{df(x_{i+1})}{d\alpha_i} = -r_{i+1}^T d_i = 0 \tag{4}$$

$$\alpha_i = \frac{r_i^T d_i}{d_i^T A d_i} \tag{5}$$

By definition, the residual is orthogonal to the previous search directions, we also have $r_i^T r_j = \delta_{ij}$. Since

$$r_{i+1} = -A(e_{i+1}) = -A(e_i + \alpha_i d_i) = r_i - \alpha_i A d_i \tag{6}$$

1.3.1 Procedure

The cgm algorithm can be summed up as follows:
Start with a guess solution x_0.
Let the first direction be the residual in x_0

$$d_0 = r_0 = b - Ax_0 \tag{7}$$

Now, at each iteration, we can compute

$$\begin{aligned} \alpha_i &= \frac{r_i^T r_i}{d_i^T A d_i} \\ x_{i+1} &= x_i + \alpha_i d_i \\ r_{i+1} &= r_i - \alpha_i A d_i \\ \beta_{i+1} &= \frac{r_{i+1}^T r_{i+1}}{r_i^T r_i} \\ d_{i+1} &= r_{i+1} + \beta_{i+1} d_i \end{aligned}$$

1.4 Preconditioning

The rate of convergance of cgm depends on the conditioning of the matrix A, defined as $\kappa(A) = \frac{\max \lambda_i}{\min \lambda_i}$, where λ_i are the eigenvalues of the matrix.
The closer $\kappa(A)$ is to 1, the faster the convergence of the method.
Given a certain matrix M, symmetric, positive definite and easily invertible and such that $M^{-1}A$ has better conditioning than A, which is to say M well approximates A, we can hope to solve the problem

$$M^{-1}Ax = M^{-1}b \tag{8}$$

much faster than the original problem, where the two solutions will be the same.
The problem is that $M^{-1}A$ is not necessarily symmetric or positive definite.
The fact that $\exists E$ such that $M = EE^T$ and $E^{-1}AE^{-T}$ is symmetric and positive definite, we can solve the problem.

$$E^{-1}AE^{-T}x = E^{-1}b \tag{9}$$

By using some clever substitutions, we can go back to the original problem with the aid of the preconditioner, giving the following algorithm

$$\begin{aligned}
r_0 &= b - Ax_0 \\
d_0 &= M^{-1}r_0 \\
\alpha_i &= \frac{r_i^T M^{-1} r_i}{d_i^T A d_i} \\
x_{i+1} &= x_i + \alpha_i d_i \\
r_{i+1} &= r_i - \alpha_i A d_i \\
\beta_{i+1} &= \frac{r_{i+1}^T M^{-1} r_{i+1}}{r_i^T M^{-1} r_i} \\
d_{i+1} &= M^{-1} r_{i+1} + \beta_{i+1} d_i
\end{aligned}$$

2 Finding the smallest eigenvalue

Finding the smallest/biggest eigenvalue-eigenvector pair of a matrix amounts to evaluating the unconstrained minimum/maximum of the Reyleigh quotient

$$\lambda(x) = \frac{x^T A x}{x^T x} \tag{10}$$

Or, more generally

$$Ax = B\omega x \implies \lambda(x) = \frac{x^T A x}{x^T B x} \tag{11}$$

λ is not a quadratic form, hence the cgm needs to be modified to use it.

2.1 Useful multivariable relations

Given $f(x) = x^T A x$ and taking the derivative of f in the direction of v

$$f(x + hv) = (x + hv)^T A(x + hv) = f(x) + hv^T Ax + hx^T Av + o(h) \tag{12}$$

$$\frac{\mathrm{d}f}{\mathrm{d}v} = \lim_{h \to 0} \frac{f(x + hv) - f(x)}{h} = v^T Ax + x^T Av = v^T Ax + v^T A^T x \tag{13}$$

We can now evaluate the gradient of f in x

$$\nabla_x f(x) = \frac{\mathrm{d}f}{\mathrm{d}v} = (A + A^T)x \tag{14}$$

We can now take the gradient of the Rayleigh quotient

$$\nabla\lambda(x) = \frac{(A + A^T)xx^T Bx - (B + B^T)xx^T Ax}{(x^T Bx)^2} \tag{15}$$

Using the fact that A and B are symemtric

$$\nabla\lambda(x) = 2\frac{Axx^T Bx - Bxx^T Ax}{(x^T Bx)^2} = 2\frac{Ax - \lambda(x)Bx}{x^T Bx} \tag{16}$$

2.2 Non linear conjugate gradient

Using a non quadratic form as function to be minimized, the things that will change will be

- The step size α_i will be different, we may now have multiple zeros regarding the orthogonality of the gradient and search direction.
- The factor β to compute conjugated directions no longer has equivalent forms.
- The residual needs to be computed each time as $-\nabla f(x_i)$

Let's take a look at each problem and find a workaround.

2.2.1 Step size

We ought to find the step size for which λ is minimized at each iteration. Being non linear (and non quadratic), an approximation must be done.
We can Taylor expand the function around x_i, in the direction αd_i, and find the minimum of the polynomial.
Regarding the Rayleigh quotient, it amounts to finding the positive roots of the following polynomial:

$$\begin{aligned} a\alpha_i^2 + b\alpha_i + c = 0 \\ a = (d_i^T A d_i)(x_i^T B d_i) - (x_i^T A d_i)(d_i^T B d_i) \\ b = (d_i^T A d_i)(x_i^T B x_i) - (x_i^T A x_i)(d_i^T B d_i) \\ c = (x_i^T A d_i)(x_i^T B x_i) - (x_i^T A x_i)(x_i^T B d_i) \end{aligned}$$

Being the search direction always descending, we can simply select the positive root.

2.2.2 Factor β

The choice for β is neither trivial nor unique, different formulations lead to distinct convergence properties and applicabilities.
Two possible choices are

$$\beta_{i+1}^{\text{FR}} = \frac{r_{i+1}^T r_{i+1}}{r_i^T r_i} \quad \text{or} \quad \beta_{i+1}^{\text{PR}} = \max\left\{\frac{r_{i+1}^T (r_{i+1} - r_i)}{r_i^T r_i}, 0\right\} \tag{17}$$

The max operation will restart the method if β is negative in the Polak Ribière, guaranteeing convergence.

2.2.3 Algorithm

The algorithm for minimizing the Rayleigh quotient can now be formulated as follows.

Choose an initial guess x_0.
Set the first search direction as the residual in x_0: $d_0 = r_0 = -g(x_0)$.
At each iteration, we can compute

$$\begin{aligned}\alpha_i \text{ such that } f(x + \alpha_i d_i) \text{ minimized}\\ x_{i+1} = x_i + \alpha_i d_i\\ r_{i+1} = -g(x_{i+1})\\ \beta_{i+1} \text{ from one of the possible choices}\\ d_{i+1} = r_{i+1} + \beta_{i+1} d_i\end{aligned}$$

Since $\lambda(x)$ is not a quadratic form, the algorithm won't converge in n steps, so that we will need to check for convergence at each iteration.

2.2.4 Stopping

As suggested in [painless conjugate gradient], a possible stopping criterion can be to check whether

$$\|g(x_i)\| < \epsilon \|g(x_0)\| \tag{18}$$

3 Finite Differences

We can employ discretization methods such as finite differences to approximate the solution of a differential equation.

3.1 Second order ODEs

The taylor expansion of a function $\psi(x \pm h)$ around a point x is given by

$$\psi(x \pm h) = \psi(x) \pm h\psi'(x) + \frac{h^2}{2!}\psi''(x) + \dots \tag{19}$$

3.1.1 First and second derivative

By subtracting $\psi(x + h)$ and $\psi(x - h)$, we get an approximation for the first derivative

$$\psi'(x) \approx \frac{\psi(x + h) - \psi(x - h)}{2h} \tag{20}$$

By adding them, we can get an approximation for the second derivative

$$\psi''(x) \approx \frac{\psi(x + h) - 2\psi(x) + \psi(x - h)}{h^2} \tag{21}$$

3.2 Discretization

Given an eigenvalue boundary problem, formulated as

$$\psi''(x) = f(x, \psi, \psi', E) \;\; \forall x \in [a, b] \tag{22}$$

We can build a lattice of n points

$$X = \{x_i = a + ih \; \forall i = 0, \ldots, n-1\} \tag{23}$$

Writing $\psi(x_i) = \psi_i$, and the equation $\psi_i'' = f(x_i, \psi_i, \psi_i', E)$ $\forall i$ we get a linear system of the form

$$A\underline{\psi} = E\underline{\psi} \tag{24}$$

Finding the eigenvalues and eigenvectors of A amounts to finding the solutions ψ and the corrisponding eigenvalues E of the eigenvalue problem 22.

3.3 1D Harmonic oscillator

In quantum mechanics, one often finds necessary to solve the reduced Schrödinger equation

$$\hat{H}\psi = (\hat{T} + \hat{V})\psi = E\psi \tag{25}$$

Where $\hat{H}$ is the Hamiltonian, a differential operator, and E the energy associated to a state ψ.
A simple but rather useful example is the harmonic oscillator, where the potential is given by

$$V(x) = \frac{1}{2}m\omega^2(x - x_0)^2 \tag{26}$$

3.3.1 Harmonic oscillator and equilibrium

The power of the harmonic oscillator comes from the fact that a system at equilibrium will roughly have its particles in the minium of the potential energy. From a single particle point of view, we can say that the potential to which it's subjected is a function of its position x, which at equilibrium can be expanded as

$$V(x) = V(x_0) + \left.\frac{\mathrm{d}V}{\mathrm{d}x}\right|_{x_0}(x - x_0) + \frac{1}{2}\left.\frac{\mathrm{d}^2V}{\mathrm{d}x^2}\right|_{x_0}(x - x_0)^2 \tag{27}$$

Since the first derivative is zero at equilibrium, and the potential additive constant can be ignored, we can write

$$V(x) = \frac{1}{2}m\omega^2(x - x_0)^2 \tag{28}$$

Where

$$m\omega^2 = \left.\frac{\mathrm{d}^2V}{\mathrm{d}x^2}\right|_{x_0} \tag{29}$$

3.4 Matrix solution

Given the Hamiltonian

$$\hat{H} = \frac{-\hbar^2}{2m}\frac{\mathrm{d}^2}{\mathrm{d}x^2} + \frac{1}{2}m\omega^2 x^2 \tag{30}$$

We can combine 22 and 21, using a step size Δ to get

$$\frac{(\frac{K}{2}m\omega^2 x_i^2\Delta^2 - 2)\psi_j + \psi_{j-1} + \psi_{j+1}}{\Delta^2 K} = E\psi_j \tag{31}$$

Where $K = -2m/\hbar^2$.
The left hand side of 31 gives the entries of matrix A, for which the smallest eigenvalue can be found by minimizing the Rayleigh quotient with the non linear cgm.

3.4.1 Harmonic oscillator applied to nuclei

A numerical solution to the harmonic oscillator is now given for the applied case of a nucleon in the nucleus.
The value of ω can be calculated from the empirical density of nuclei, which can be written as a function of $\sqrt{\langle r^2 \rangle}$, analytically known in the case of an harmonic oscillator.

$$\hbar\omega = \frac{41}{A^{1/3}}\text{MeV} \tag{32}$$

This may seem tautological but the aim is to verify the validity of the numerical solution while seeing the method in action for a real case.
The mass of the particle is assumed to be 939 MeV.
A calculation was performed on a grid of 1000 points, in an interval $[-a, a]$ such that $a = 10$ fm.
The resulting wavefunction is shown in figure 1.
Assuming a mass number of $A = 16$, the eigenvalue (energy) associated to the graound state is 8.143 MeV.
Since the computation was done in one dimension, a factor of 3 is needed to compare it to a real nucleus.
Assuming the nucleon to be bound through a potential well of $\approx$ 40 MeV, the binding energy will be $\approx$ 40 - 8.143 $\cdot$ 3 = 15.571 MeV.

3.5 Second order PDEs

The method of finite differences can be extended to second order PDEs.
For the moment, we will consider only a 2 dimensional problem, without losing generality.
Given the boundary eigenvalue problem

$$\frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial y^2} = f(\psi, \psi_x, \psi_y, x, y, E) \tag{33}$$

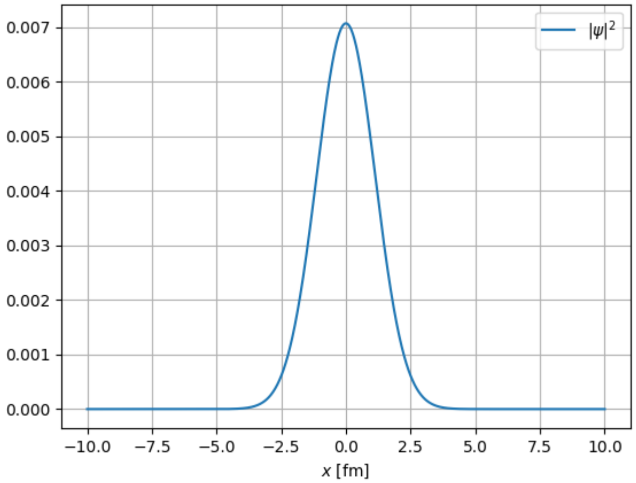

Figure 1: Ground state wavefunction of the harmonic oscillator. The solution starts roughly vanishes for $|x| > 3$ fm, as expected for a nucleus of this size.

Here, coordinate indeces x and y, are used to denote partial derivatives of ψ with respect to x and y respectively.
We can build a lattice of n_x, n_y points as follows

$$X = \{x_i = a_x + ih_x \ \forall i = 0, \dots, N_x - 1\},\ Y = \{y_j = a_y + jh_y \ \forall j = 0, \dots, N_y - 1\} \tag{34}$$

For simplicity, we can assume that f does not depend on ψ_x, ψ_y, as most simple Hamiltonians do.
By applying the finite differences method on x we get

$$\frac{\partial^2 \psi_i}{\partial y^2} + \frac{\psi_{i+1}(y) - 2\psi_i(y) + \psi_{i-1}(y)}{h_x^2} = f(\psi_i(y), x_i, y, E) \tag{35}$$

We can now write $\psi_{yy,i}$ as a function of $\psi_i(y), y, E, x_i$ and apply the finite differences method on y, which yields:

$$\frac{\psi_{i,j+1} + \psi_{i,j-1} - 2\psi_{i,j}}{h_y^2} + \frac{\psi_{i+1,j} + \psi_{i-1,j} - 2\psi_{i,j}}{h_x^2} = f(\psi_{i,j}, x_i, y_j, E) \tag{36}$$

Rearranging the terms, it's possible to obtain the following system of equations:

$$\sum_{i'j'} A_{i,j,i',j'} \psi_{i'j'} = E\psi_{i,j} \tag{37}$$

Where $A_{i,j,i',j'}$ is a hypermatrix of size $N_x \times N_y \times N_x \times N_y$.
The next task will be to find a way to express 37 as a linear system of equations,

by finding a single index μ that encapsulates the relation for all pairs (i, j), with the constraint that the resulting matrix is symmetric.
Formally

$$F : (i, j) \mapsto \mu \text{ such that } \mu = F(i, j),\ \mu' = F(i', j') \implies A_{\mu\mu'} = A_{\mu'\mu} \tag{38}$$

3.5.1 Indeces parametrization

A possible map $(i, j) \mapsto \mu$ is

$$F(i, j) = \mu = i + N_x j \tag{39}$$

Using this map, the entries of ψ_μ will be organized as

$$\psi = (\psi_{(0,0)}, \psi_{(1,0)}, \ldots, \psi_{(N_x-1,0)}, \psi_{(0,1)}, \ldots, \psi_{(N_x-1,N_y-1)}) \tag{40}$$

And the problem will be reduced to

$$\sum_{\mu'=0}^{N_x N_y} A_{\mu\mu'} \psi_{\mu'} = E\psi_\mu \tag{41}$$

Which can be diagonalized to find the solutions to the eigenvalue problem in multiple dimensions. Moreover, an approximate solution can be found using cgm, since the resulting matrix will be symmetric.

3.5.2 3D case

The 3D case follows trivially by performing discretization on the z-axis, adding a new index k and using the map $F(i, j, k) = i + N_x(j + N_y k)$.