# Computational Physics: Eigenvalue Problems 1

Stockholm University, Spring Term 2024
Max Maschke

Apr 11 2024

## Contents

# 1 Introduction

[1]

# 2 Discretising the 1D Schrödinger Equation

The time independent Schrödinger equation (SEQ) is famously an eigenvalue problem:

$$\mathcal{H} \left| \psi \right\rangle = E \left| \psi \right\rangle \tag{1}$$

For a single particle and moving to real-space representation, using the wave function $\psi(x) = \langle x | \psi \rangle$, it assumes the form

$$\left( -\frac{\hbar^2}{2m} \partial_x^2 + V(x) \right) \psi(x) = E\psi(x) \tag{2}$$

where $V(x)$ is the real-space representation of the potential operator $\mathcal{V}$, which is a diagonal operator in this basis. $\mathcal{H}$ itself is, however, not diagonal, which is why, considering the difficulty of obtaining analytical solutions to the SEQ, one might be interested in numerical approaches to the eigenproblem posed.

   The first necessary step is to translate the infinite dimensional problem stated above into a finite basis. Naturally, this involves some sort of approximation. Intuitively, one possible approach is to go from the infinite dimensional eigenbasis of the position operator $\mathcal{X}$

$$\mathcal{X} \left| x \right\rangle = x \left| x \right\rangle, \quad \langle x' | x \rangle = \delta(x' - x), \quad x \in \mathbb{R} \tag{3}$$

to a discretised set of positions $x_i$ with an associated discrete position operator $X$

$$X \left| x_i \right\rangle = x_i \left| x_i \right\rangle, \quad \langle x_j | x_i \rangle = \delta_{ij}, \quad i \in \mathbb{N}. \tag{4}$$

If we additionally impose a maximum and minimum value of $x_i$ (N.B. this implies assuming $\psi(x) = 0$ outside for the purposes of finding solutions to the SEQ) and require neighbouring positions to have constant spacing $h$, the set of allowed positions assumes a finite number $n$:

$$x_i = x_{\min} + i \cdot h, \quad i = 0, \dots, n-1 \tag{5}$$

It will be convenient to set $x_{\min} = -x_{\max}$ henceforth. For a given $n$, that yields

$$h = 2x_{\max}/(n-1). \tag{6}$$

What we have achieved by moving to this finite basis is that, if we can map $\mathcal{H}$ to its finite representation $H \in \mathbb{C}^n$, (1) can be represented as a matrix equation, which allows us to unleash the whole toolbox of numerical linear algebra onto it. Moreover, any real function $\mathcal{F}$ on $[-x_{\max}, x_{\max}]$ can be represented by a vector $f \in \mathbb{R}^n$ by the identification

$$f = \sum_i f(x_i) e_i. \tag{7}$$

Naturally, this also applies to the wave function, meaning a physical state $\left| \psi \right\rangle$ can be approximately represented in our finite dimensional space. The correct choice for the inner product is the standard inner product, the correct norm is the euclidean norm.

As previously mentioned, the potential operator $\mathcal{V}$ and respectively its finite representation $V$ are diagonal in the real-space basis, i.e.

$$\langle x_j|V|x_i\rangle = \delta_{ij}V(x_i) \Rightarrow V = \begin{bmatrix} V(x_0) & & & \\ & V(x_1) & & \\ & & \ddots & \\ & & & V(x_{n-1}) \end{bmatrix}. \tag{8}$$

The same is not true for the momentum operator $\mathcal{P}$ or, respectively, $P$. It is a differential operator in position space and thus has no "exact" representation in the finite basis. However, its action on a continuous state can be approximated in the discrete case using a finite-difference formula for the second derivative. Two examples for centred finite difference formulae are the three- and five-point formulae:

$$f''(x_i) = \frac{1}{h^2}\left(f(x_{i-1}) - 2f(x_i) + f(x_{i+1})\right) + O(h^2) \tag{9}$$

$$f''(x_i) = \frac{1}{h^2}\left(-\frac{1}{12}f(x_{i-2}) + \frac{4}{3}f(x_{i-1}) - \frac{5}{2}f(x_i) + \frac{4}{3}f(x_{i+1}) - \frac{1}{12}f(x_{i+2})\right) + O(h^4) \tag{10}$$

Note that the coefficients of the $f(x_j)$ in both of these formulae are symmetric around the centre point. This is important because we want to translate the Hermitian operator $(\mathcal{P}^2)^\dagger = \mathcal{P}^2$ into an Hermitian (actually symmetric because real-valued) matrix $(P^2)^H = P^2$, which, using the five-point formula, has the form

$$P^2 = -\frac{\hbar^2}{h^2}\begin{bmatrix} -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} & & & & \\ \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} & & & \\ -\frac{1}{12} & \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} & & \\ & -\frac{1}{12} & \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & -\frac{1}{12} & \frac{4}{3} & -\frac{5}{2} & \frac{4}{3} & -\frac{1}{12} \end{bmatrix}. \tag{11}$$

This way, for any given potential, we obtain a real, symmetric matrix

$$H = \frac{1}{2m}P^2 + V. \tag{12}$$

If we can find its eigenpairs numerically, we will obtain an approximate solution to the full SEQ. Note that while the full equation will usually have infinitely many solutions, we can only expect to find as many as we allow discrete positions, i.e. $n$. Furthermore, due to the sampling theorem, we should not expect to be able to represent quickly oscillating wave functions.

## 3  Numerical Eigenproblem Solvers

Analytically, eigenvalues may be obtained by finding the roots of the characteristic polynomial of a matrix $A$:

$$\det(A - \lambda I) = 0 \tag{13}$$

One then gets the corresponding eigenvectors by solving

$$(A - \lambda I)x = 0. \tag{14}$$

The problem with this approach for numerical use is that the determinant is highly non-linear in the matrix elements $a_{ij}$. Small errors can thus propagate and yield hugely wrong eigenvalues. Other methods are thus needed.

One example for an efficient and stable eigensolver is the $QR$-algorithm, based on the $QR$ decomposition of a matrix $A$:

$$A = QR, \tag{15}$$

where $Q \in \mathrm{U}(n)$ and $R$ upper triangular. The idea is to iteratively apply orthogonal or, respectively, unitary transformations to $A$ that turn the elements below the diagonal zero. Such transformations are, for example, Householder transformations or Givens rotations. When this has been achieved, the eigenvalues are the diagonal elements.

Eigenvalue solvers usually have complexity $O(n^3)$. If not all eigenvalues are needed, another possible approach that might save some resources is inverse iteration.

# References

[1] Max Maschke. *compphys*. URL: https://github.com/max-mas/compphys (visited on 04/03/2024).