# Computational physics homework 2

Hao Wang

October 7, 2025

## Summary

This assignment consisted of three computational physics problems, each illustrating a different numerical method. In Problem 1, we implemented fixed-point iteration with relaxation and overrelaxation to solve a nonlinear equation and studied how the relaxation parameter $\omega$ affects convergence. In Problem 2, we derived and solved the transcendental equation defining Wien's displacement constant using binary search, and applied the result to estimate the Sun's surface temperature. In Problem 3, we implemented a robust gradient descent algorithm to fit the Schechter function to galaxy stellar mass function data, minimizing $\chi^2$ and verifying robustness of the fit. Overall, the exercises demonstrated the importance of algorithmic choices (relaxation, step control, gradient clipping) in ensuring convergence and stability of numerical methods.

## Methods

### Problem 1: Relaxation and Overrelaxation

We considered the nonlinear equation

$$x = 1 - e^{-cx}, \quad c = 2,$$

and solved it using fixed-point iteration. The ordinary relaxation method iterates $x_{n+1} = f(x_n)$ with $f(x) = 1 - e^{-cx}$. We then implemented overrelaxation:

$$x_{n+1} = x_n + (1+\omega)\big(f(x_n) - x_n\big),$$

where $\omega$ is the relaxation parameter. For $\omega = 0$ this reduces to ordinary relaxation, while $\omega > 0$ can accelerate convergence. Assume $x^*$ is the solution of $x^* = f(x^*)$, $x$ is the previous iteration, $x'$ is the current iteration. Then we have

$$\begin{aligned}
x' &= (1+\omega)f(x) - \omega x \\
x' - x^* &= (x - x^*)[(1+\omega)f'(x^*) - \omega] \\
\epsilon' &= \epsilon[(1+\omega)f'(x^*) - \omega] \\
x^* &= x + \epsilon = x + \epsilon'/[(1+\omega)f'(x^*) - \omega]
\end{aligned}$$

Thus we have

$$\epsilon = \frac{x - x'}{1 - 1/[(1+\omega)f'(x) - \omega]}$$

We monitored the number of iterations required to reach a tolerance of $10^{-6}$.

## Problem 2: Wien's Displacement Constant

Planck's law leads to the transcendental equation

$$5e^x = x + 5,$$

with $x = \frac{hc}{\lambda k_B T}$. We solved this equation numerically using the binary search method to accuracy $\epsilon = 10^{-6}$. The Wien displacement constant is then

$$b = \frac{hc}{k_B x},$$

and the Sun's surface temperature was estimated from $\lambda_{\text{peak}} = 502\,\text{nm}$ via $T = b/\lambda_{\text{peak}}$.

## Problem 3: Galaxy Stellar Mass Function

We fit the Schechter function

$$n(M_{gal}) = \phi^* \left(\frac{M_{gal}}{M_*}\right)^{\alpha+1} \exp\left(-\frac{M_{gal}}{M_*}\right) \ln 10$$

to COSMOS survey data by minimizing

$$\chi^2(\phi^*, M_*, \alpha) = \sum_i \left(\frac{n_{\text{obs},i} - n_{\text{model},i}}{\sigma_i}\right)^2.$$

We implemented gradient descent with numerical derivatives, gradient clipping, and learning-rate decay to ensure stability. Parameters were optimized in log-space for $\phi^\star$ and $M^\star$ to avoid overflow.

# Results

## Problem 1

The ordinary relaxation method converged more slowly than over relaxation. Overrelaxation with $\omega$ from 0.1 to 1.1 reduced the iteration count significantly, while very large $\omega$ values led to slightly slower converging rate. This confirms that carefully chosen $\omega$ accelerates convergence.

```
Relaxation: root ≈ 0.796813, iterations = 14, error estimate ≈ 5.01e-07
Overrelaxation (ω=0.1): root ≈ 0.796813, iterations = 12, error estimate ≈ 4.30e-07
Overrelaxation (ω=0.2): root ≈ 0.796813, iterations = 10, error estimate ≈ 5.00e-07
Overrelaxation (ω=0.3): root ≈ 0.796813, iterations = 8, error estimate ≈ 8.12e-07
Overrelaxation (ω=0.4): root ≈ 0.796812, iterations = 7, error estimate ≈ 3.01e-07
Overrelaxation (ω=0.5): root ≈ 0.796812, iterations = 4, error estimate ≈ 2.43e-07
Overrelaxation (ω=0.6): root ≈ 0.796812, iterations = 5, error estimate ≈ 1.01e-07
Overrelaxation (ω=0.7): root ≈ 0.796812, iterations = 4, error estimate ≈ 2.16e-08
Overrelaxation (ω=0.8): root ≈ 0.796812, iterations = 5, error estimate ≈ 5.04e-07
Overrelaxation (ω=0.9): root ≈ 0.796812, iterations = 7, error estimate ≈ 1.60e-07
Overrelaxation (ω=1.0): root ≈ 0.796813, iterations = 8, error estimate ≈ 3.90e-07
Overrelaxation (ω=1.1): root ≈ 0.796811, iterations = 9, error estimate ≈ 8.09e-07
```

## Problem 2

The binary search yielded the root

$$x \approx 4.965,$$

giving Wien's displacement constant

$$b \approx 2.898 \times 10^{-3}\,\mathrm{m{\cdot}K}.$$

From $\lambda_{\text{peak}} = 502\,\mathrm{nm}$, the estimated solar surface temperature is

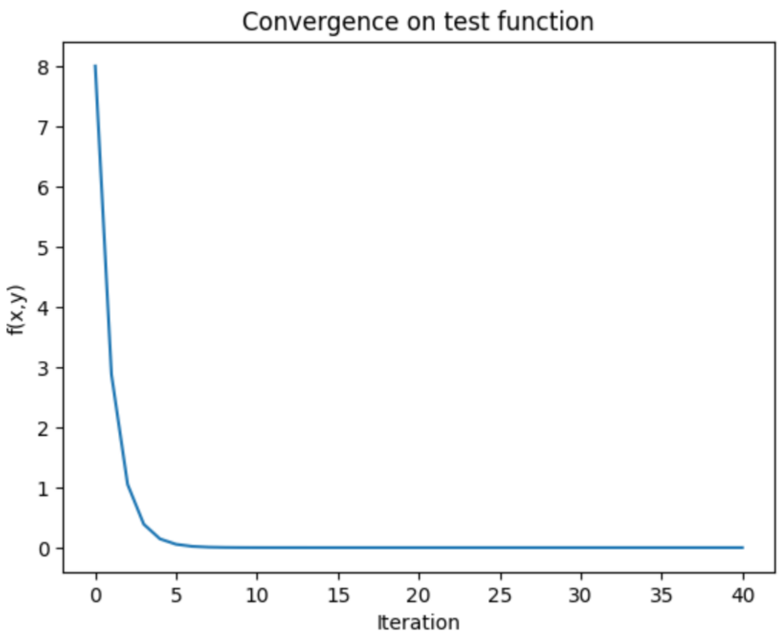
$$T \approx 5772\,\mathrm{K},$$

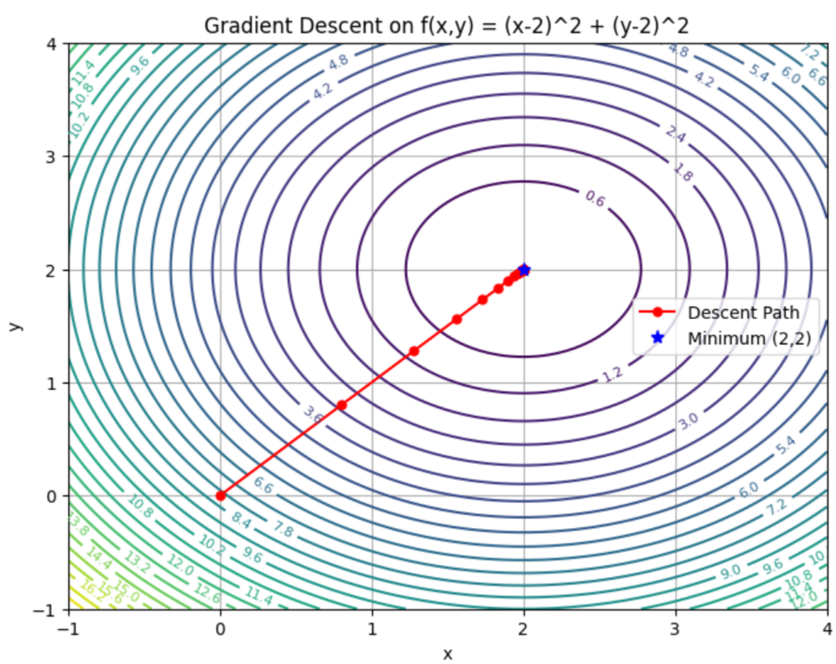
consistent with accepted values.

## Problem 3

We first tested our gradient descent on a test function

$$f(x) = (x-2)^2 + (y-2)^2$$



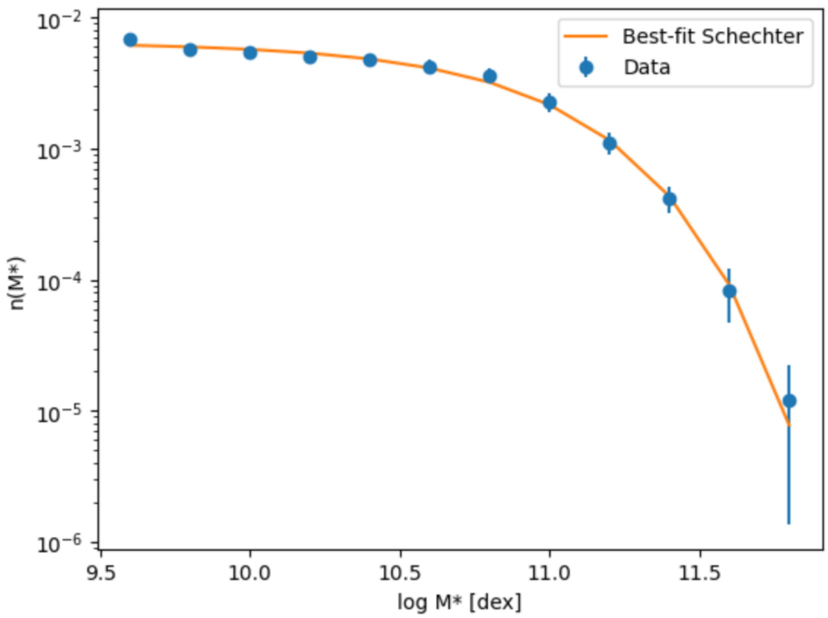

We can see this algorithm works well.

For the Schechter function, since parameters have very different order of magnitude, we do a gradient clipping to avoid blowing up. We also dynamically deccay the learning rate to avoid oscillation at the end of converging.

Our gradient descent successfully minimized $\chi^2$, converging smoothly with learning-rate decay and gradient clipping. The best-fit parameters were approximately:

$$\phi^{\star} \approx 0.002703, \quad M^{\star} \approx 94673085834, \quad \alpha \approx -1.009.$$

We can see the best-fit Schechter function matches the observed stellar mass function well on a log-log plot.




We also plotted the error-iteration curve with different initial parameters to show that the algorithm is pretty robust.

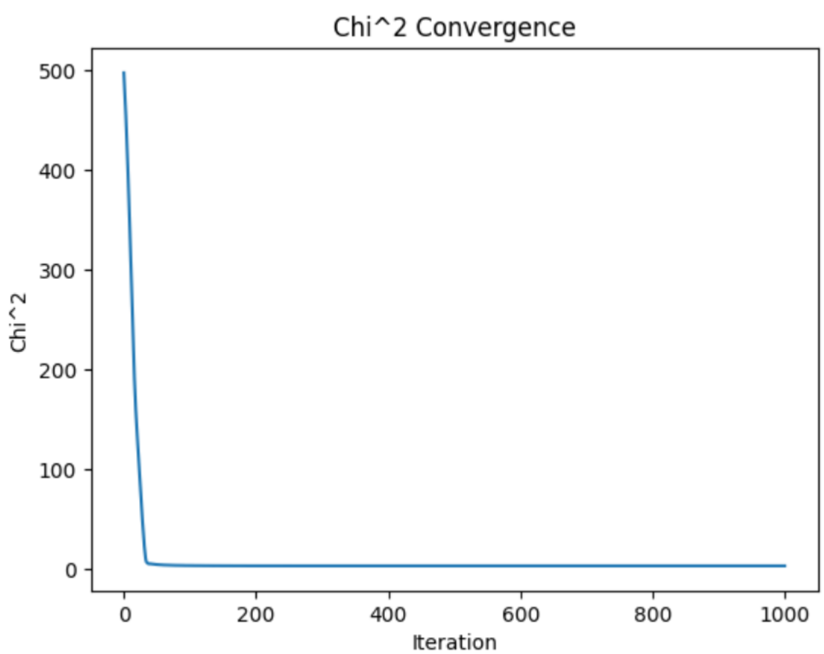



Figure 1: Initial parameters: $\phi = 10^{-3},\ M^{*} = 10^{10},\ \alpha = -1.$

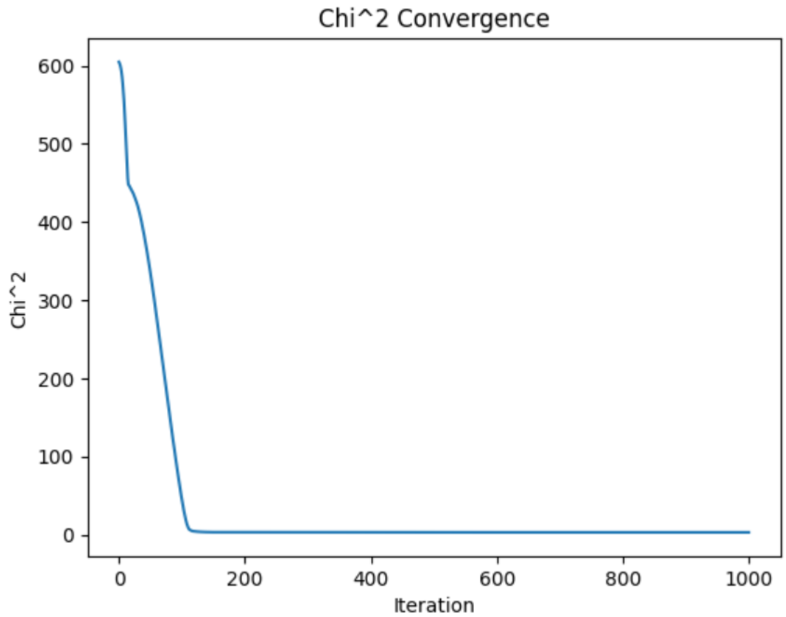



Figure 2: Initial parameters: $\phi = 10^{-2},\ M^{*} = 10^{9},\ \alpha = -2.$

# Code

https://github.com/hw3926/Computational-Physics/tree/main/HW2