Practical Work: Fixed Point Iteration Method

Objectives

- Understand the concept of fixed point iteration and its relation to root finding.
- Derive a suitable fixed point function $g(x)$ for a given equation.
- Analyze convergence conditions using the Banach fixed point theorem.
- Implement the fixed point iteration algorithm in MATLAB.
- Experiment with different functions and convergence rates.
- Apply the method to solve practical problems.

1 Theoretical Background

A fixed point of a function $g(x)$ is a value p such that $g(p) = p$. To find a root of $f(x) = 0$, we can rewrite it as $x = g(x)$ (e.g., $g(x) = x - f(x)$ or other transformations). The **fixed point iteration** is defined by:

$$x_{n+1} = g(x_n), \quad n = 0, 1, 2, \dots$$

starting from an initial guess x_0.

Convergence

The Banach fixed point theorem (contraction mapping theorem) states that if g is continuous on $[a, b]$, differentiable on (a, b), and there exists a constant $0 < L < 1$ such that $|g'(x)| \leq L$ for all x in $[a, b]$, and if $g([a, b]) \subseteq [a, b]$, then:

- There exists a unique fixed point p in $[a, b]$.
- The iteration converges to p for any starting point $x_0 \in [a, b]$.
- The convergence is linear with rate L: $|x_{n+1} - p| \leq L|x_n - p|$.

If $g'(p) = 0$ and $g''(p) \neq 0$, the convergence can be quadratic (Newton's method is a special case).

Common Transformations

To solve $f(x) = 0$, we can set:

- $g(x) = x - f(x)$ (simple but not always contracting)
- $g(x) = x - \frac{f(x)}{f'(x)}$ (Newton's method, quadratic convergence)
- $g(x) = \phi(x)$ from rearranging $f(x) = 0$ in various ways.

2 Part 1: Manual Calculations

Consider the equation $x^3 - x - 2 = 0$. It has a root near 1.5.

(a) Rewrite the equation in the form $x = g(x)$ in three different ways:

(i) $g_1(x) = x^3 - 2$

(ii) $g_2(x) = \sqrt[3]{x+2}$

(iii) $g_3(x) = \frac{x^3-2}{x}$ (provided $x \neq 0$)

(b) For each g_i, compute $g_i'(x)$ and evaluate $|g_i'(1.5)|$. Determine which transformation satisfies the contraction condition in a neighborhood of the root.

(c) Choose the best $g(x)$ and perform the first three iterations starting from $x_0 = 1.5$. Fill the table below.

Table 1: Manual fixed point iterations

n	x_n	$g(x_n)$
0	1.5	
1		
2		
3		

3 Part 2: MATLAB Implementation

Write a MATLAB function ‘fixedpoint‘ that implements the fixed point iteration. The function should have the following signature:

```
function [root, iter, history] = fixedpoint(g, x0, tol, maxiter)
```

where:

- ‘g‘ is a function handle for the fixed point function.
- ‘x0‘ is the initial guess.
- ‘tol‘ is the tolerance (stop when $|x_{n+1} - x_n| < \text{tol}$).

- ‘maxiter‘ is the maximum number of iterations.
- ‘root‘ is the approximate fixed point.
- ‘iter‘ is the number of iterations performed.
- ‘history‘ is an optional output matrix containing $[n, x_n, g(x_n), |x_{n+1} - x_n|]$ for each iteration.

Requirements:

- For each iteration, compute the new value and check the stopping condition.
- Store iteration data if ‘history‘ is requested.
- If the stopping condition is not met within ‘maxiter‘, issue a warning and return the last value.
- Optional: detect divergence if the iterate goes out of bounds or if the difference starts increasing.

Example skeleton:

```
function [root, iter, history] = fixedpoint(g, x0, tol, maxiter)
    if nargout == 3
        history = [];
    end

    x = x0;
    for iter = 1:maxiter
        x_new = g(x);
        err = abs(x_new - x);

        if nargout == 3
            history = [history; iter-1, x, x_new, err];
        end

        if err < tol
            root = x_new;
            return;
        end

        x = x_new;
    end

    root = x;
    warning('Maximum iterations reached without convergence');
end
```

4 Part 3: Numerical Experiments

Use your ‘fixedpoint‘ function to study the behavior of the method on different functions.

Experiment 1: Linear vs. Nonlinear Contraction

Solve $x = e^{-x}$ on $[0, 1]$. The exact fixed point is $x \approx 0.567143290409784$ (Lambert W function).

(a) Use $g(x) = e^{-x}$ with $x_0 = 0.5$, tolerance 10^{-10}, maxiter 100.

(b) Record the number of iterations and the final approximation.

(c) Plot $|x_n - p|$ on a semilog scale. What is the approximate convergence rate?

(d) Compute $g'(p)$ numerically. How does it relate to the observed convergence?

Experiment 2: Divergence

Consider the equation $x^3 - x - 2 = 0$ with $g_1(x) = x^3 - 2$ and $x_0 = 1.5$.

(a) Run your fixedpoint function with this g. What happens?

(b) Explain the result using the contraction condition ($|g'(p)| > 1$).

Experiment 3: Quadratic Convergence (Newton's Method)

Implement Newton's method as a fixed point iteration with $g(x) = x - f(x)/f'(x)$. Use it to find the root of $f(x) = x^3 - x - 2$ starting from $x_0 = 1.5$. Use tolerance 10^{-12}. Record the number of iterations and compare with the linear convergence case from Experiment 1.

Experiment 4: Sensitivity to Initial Guess

Choose a function $g(x) = \sqrt{x+2}$ for the equation $x^3 - x - 2 = 0$. Test with initial guesses $x_0 = 0.5$, $x_0 = 5$, and $x_0 = -1$. For each, report convergence or divergence and explain based on the contraction condition.

5 Part 4: Application – Beam Deflection

In civil engineering, the deflection y of a simply supported beam under a uniform load is given by:

$$y = \frac{wx}{24EI}(L^3 - 2Lx^2 + x^3),$$

where w is the load per unit length, E is Young's modulus, I is the moment of inertia, and L is the length. For a given deflection measurement at a point, we may need to solve for the load w. Suppose we measure a deflection y_0 at $x = L/2$. The equation becomes:

$$y_0 = \frac{w(L/2)}{24EI}\left(L^3 - 2L(L/2)^2 + (L/2)^3\right) = \frac{5wL^4}{384EI}.$$

Solving for w is trivial. Instead, consider a more complex scenario where the load distribution is not uniform but described by a transcendental equation. For instance, the deflection due to a concentrated load P at the center is:

$$y = \frac{PL^3}{48EI}.$$

If we have a beam with an unknown material property such that E depends on the load (nonlinear behavior), we might get an equation like:

$$y_0 = \frac{PL^3}{48(E_0 + \alpha P)I},$$

where α is a constant. Solve for P given y_0, L, E_0, I, α.

Define $f(P) = P - \frac{48E_0Iy_0}{L^3} - \frac{48\alpha Iy_0}{L^3}P$? Actually rearrange to fixed point form: from $y_0 = \frac{PL^3}{48(E_0+\alpha P)I}$, we have:

$$P = \frac{48y_0I(E_0 + \alpha P)}{L^3}.$$

Let $C = \frac{48y_0I}{L^3}$, then $P = C(E_0 + \alpha P) = CE_0 + C\alpha P$. So $P(1 - C\alpha) = CE_0$, which is linear and solved directly. That is too simple. Instead, we can create a nonlinear example: suppose the deflection formula includes a nonlinear term due to large deformation:

$$y = \frac{PL^3}{48EI}\left(1 + \beta P^2\right).$$

Then we need to solve P from:

$$P = \sqrt{\frac{1}{\beta}\left(\frac{48EIy}{PL^3} - 1\right)} \quad \text{or} \quad P = \frac{48EIy}{L^3(1 + \beta P^2)}.$$

The second form is suitable for fixed point iteration. Choose reasonable parameters and solve for P.

Provide a concrete set of numbers: $L = 5$ m, $E = 200$ GPa, $I = 2 \times 10^{-4}$ m^4, $y = 0.01$ m, $\beta = 0.05$ kN^{-2}. Use fixed point iteration to find P in kN.

6 Part 5: Analysis Questions

Answer the following questions in your report.

1. What is the condition for convergence of the fixed point iteration? Explain with the contraction mapping theorem.
2. How does the convergence rate depend on $|g'(p)|$? What happens if $|g'(p)| = 0$?
3. Can the fixed point method converge to a root if the initial guess is not close to the root? Provide examples.
4. Compare fixed point iteration with the bisection method in terms of speed, conditions, and robustness.
5. What is Aitken's Δ^2 acceleration, and how could it be used to speed up convergence of a linearly convergent fixed point iteration?

Report Requirements

Submit a report containing:

- A brief introduction to the fixed point method.
- The completed manual calculations (Part 1).
- Your MATLAB code (with comments).
- Tables and plots from the numerical experiments (Part 3).
- Results and discussion for the application problem (Part 4).
- Answers to the analysis questions (Part 5).
- A conclusion summarizing what you learned.

A Appendix: Sample MATLAB Code for Experiments

Below is an example script to run Experiment 1.

```
clear; clc;
g = @(x) exp(-x);
x0 = 0.5;
tol = 1e-10;
maxiter = 100;
[root, iter, history] = fixedpoint(g, x0, tol, maxiter);
fprintf('Fixed␣point:␣%.12f\n', root);
fprintf('Iterations:␣%d\n', iter);

% Plot convergence
err = history(:,4); % absolute difference
semilogy(0:iter-1, err, 'o-');
xlabel('Iteration');
ylabel('|x_{n+1}␣-␣x_n|');
title('Convergence␣of␣fixed␣point␣iteration');
grid on;

% Compute derivative at fixed point
df = (g(root+1e-6) - g(root-1e-6)) / (2e-6);
fprintf('g''(p)␣$\approx$␣%.4f\n', df);
```

Listing 1: run_exp1.m