Assignment-1: Introduction to Numerical Analysis and Simulation using MATLAB

Name:- Bajarang Mishra

Roll no:- 22CH10016

Group:- 3

Question:-

Given the values of all the constants and the Mathematical Relationship for this system, evaluate the optimal values using Numerical Techniques like the Bisection Method, the Regula-Falsi Method, Secant Method, and Newton-Raphson Method, et,c and evaluate the convergence rate for each of them.

Assignment-1 MATLAB Code:-

```
%Initialization of given constants and calculation of required variables
 R = 0.08206; %unit :- L·atm/(mol·K)
 T = 250 + 273.15; %unit :-Temperature conversion from degree to Kelvin
 P = 10; %unit :-Pressure in atm
 %calculation of a and b
 Tc = 407.5; %unit :-Temperature already in Kelvin
 Pc = 113.3; %unit :-Pressure in atm
 a = (27*R^2*Tc^2)/(64*Pc); %unit :-L^2·atm/mol^2
 b= (R*Tc)/(8*Pc); %unit :-L/mol

 %Van der Waals equation given as a function
 vdw_eqn = @(V) (P + a/V^2)*(V - b) - R*T;
 %Derivative of the Van der Waals equation for Newton-Raphson
 vdw_deriv = @(V) -2*a*(V-b)/V^3 + (P + a/V^2);
 %Initial guess and tolerance
 V_initial = 0.5; %An initial guess for molar volume
 tol = 1e-6; %Tolerance for convergence
 max_iter = 1000; %Maximum number of iterations
 %% Fixed-Point Iteration
 % Rearrange equation: V = g(V)
 g = @(V) (R*T + b*P + (a*(V-b)/V^2)) / P;

V_fixed = V_initial;
% Initial guess
for i = 1:max_iter
```

```
    V_new = g(V_fixed);
    if abs(V_new - V_fixed) < tol
        break;
    end
    V_fixed = V_new;
end
disp(['Fixed-Point Iteration: V = ', num2str(V_fixed)]);
fprintf('Molar Volume from fixed point method is :%0.6f L/mol\n',V_fixed);

%% Newton-Raphson Method
V_newton = V_initial;
% Initial guess
for i = 1:max_iter
    f_val = vdw_eqn(V_newton);
    f_deriv = vdw_deriv(V_newton);
    V_new = V_newton - (f_val / f_deriv);
    if abs(V_new - V_newton) < tol
        break;
    end
    V_newton = V_new;
end
disp(['Newton-Raphson Method: V = ', num2str(V_newton)]);
fprintf('Molar Volume from Newton-Raphson function is :%0.6f L/mol\n',V_newton);

%% Bisection Method
% Define interval [V1, V2]
V1 = 0;
% Lower bound
V2 = 10;
% Upper bound
if vdw_eqn(V1) * vdw_eqn(V2) > 0
    error('No root in the interval. Adjust bounds.');
end
for i = 1:max_iter
    V_mid = (V1 + V2) / 2;
    if abs(vdw_eqn(V_mid)) < tol
        break;
    elseif vdw_eqn(V1) * vdw_eqn(V_mid) < 0
        V2 = V_mid;
    else
        V1 = V_mid;
    end
end
disp(['Bisection Method: V = ', num2str(V_mid)]);
```

```
fprintf('Molar Volume from Bisection function is :%0.6f L/mol\n',V_mid);

%% MATLAB BUILT-IN fzero function
disp('MATLAB fzero :');
v_fzero = fzero(f,[V1,V2]);
fprintf('Molar Volume from fzerofunction is :%0.6f L/mol\n',v_fzero);
```