Maddox Gonzalez

Contents

Due 3/3/25

Project 5

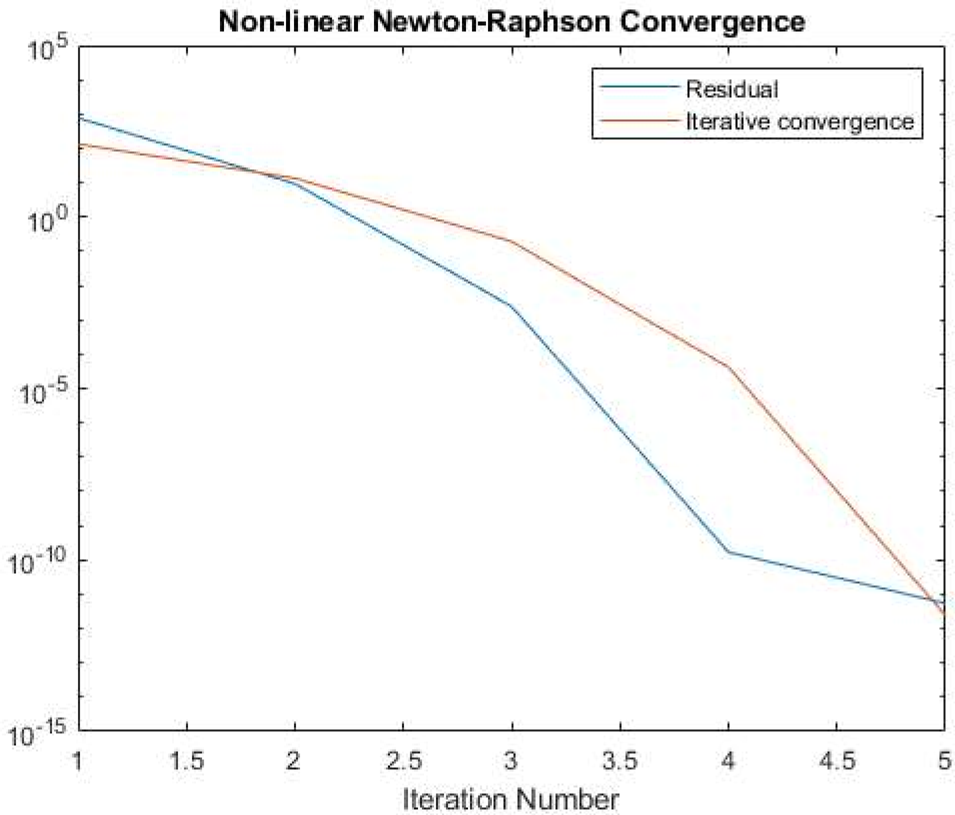
```
clc; clear; close all;
format long;

% initialize
k0 = 25;
a  = 0.075;
T0 = 375;
TL = 25;

dx = 1e-5;

tol = 1e-5;
max_iter = 100;

% initial guess
guess = [200; 150; 100; 20];

nonlinear_f = @(T) f_nonlinear(T, k0, a, T0, TL);

J_init = Jacobian(nonlinear_f, guess, dx);
fprintf('\nInitial Jacobian (5 decimals):\n');
fprintf('%12.5f %12.5f %12.5f %12.5f\n', J_init');

% newton raphson
[x_nl, res_hist, conv_hist, num_iters] = NLNR(nonlinear_f, guess, tol, dx, max_iter);

% temperatures
fprintf('\nNon-linear solution: Temperatures (T1, T2, T3, T4) = \n');
fprintf('%12.5f\n', x_nl);

linear_f = @(T) f_linear(T, k0, T0, TL);  % a is set to 0 in f_linear
[x_lin, res_hist_lin, conv_hist_lin, num_iters_lin] = NLNR(linear_f, guess, tol, dx, max_iter);

fprintf('\nLinear solution: Temperatures (T1, T2, T3, T4) = \n');
fprintf('%12.5f\n', x_lin);

iters = 1:num_iters;
figure;
semilogy(iters, res_hist, iters, conv_hist);
title('Non-linear Newton-Raphson Convergence');
xlabel('Iteration Number');
legend('Residual', 'Iterative convergence');
```

Function Definitions

```
function val = f_nonlinear(T, k0, a, T0, TL)

    val(1,1) = (k0 + (a/2)*(T(1)+T0))*T0 ...
         - (2*k0 + (a/2)*(T0+2*T(1)+T(2)))*T(1) ...
         + (k0 + (a/2)*(T(1)+T(2)))*T(2);

    val(2,1) = (k0 + (a/2)*(T(2)+T(1)))*T(1) ...
         - (2*k0 + (a/2)*(T(1)+2*T(2)+T(3)))*T(2) ...
         + (k0 + (a/2)*(T(2)+T(3)))*T(3);

    val(3,1) = (k0 + (a/2)*(T(3)+T(2)))*T(2) ...
         - (2*k0 + (a/2)*(T(2)+2*T(3)+T(4)))*T(3) ...
         + (k0 + (a/2)*(T(3)+T(4)))*T(4);

    val(4,1) = (k0 + (a/2)*(T(4)+T(3)))*T(3) ...
         - (2*k0 + (a/2)*(T(3)+2*T(4)+TL))*T(4) ...
         + (k0 + (a/2)*(T(4)+TL))*TL;
end

function val = f_linear(T, k0, T0, TL) %a = 0 here

    val(1,1) = k0*T0 - 2*k0*T(1) + k0*T(2);
    val(2,1) = k0*T(1) - 2*k0*T(2) + k0*T(3);
    val(3,1) = k0*T(2) - 2*k0*T(3) + k0*T(4);
    val(4,1) = k0*T(3) - 2*k0*T(4) + k0*TL;
end

function [x, res, conv, k] = NLNR(f, guess, tol, dx, max_iter)


    x = guess;
    k = 1;
    while k <= max_iter
        xold = x;

        J = Jacobian(f, x, dx);
        b = -f(x);

        delx = J \ b;
        x = xold + delx;

        res(k) = norm(f(x), inf);
        conv(k) = norm(delx, inf);

        if (res(k) < tol) && (conv(k) < tol)
            break;
        end

        if res(k) > 1e10
            warning('Divergence detected. Exiting.');
            break;
        end
        init_res = norm(f(x), inf);
        fprintf('Initial residual norm = %12.5f\n', init_res);
        k = k + 1;
```

```
    end

end

function J = Jacobian(f, x, dx)

    fx = f(x);
    n = length(x);
    m = length(fx);
    J = zeros(m, n);

    for j = 1:n
        xdel = x;
        xdel(j) = x(j) + dx;
        J(:,j) = (f(xdel) - fx) / dx;
    end
end
```

```
Initial Jacobian (5 decimals):
   -80.00000     36.25000      0.00000      0.00000
    40.00000    -72.50000     32.50000      0.00000
     0.00000     36.25000    -65.00000     26.50000
     0.00000      0.00000     32.50000    -53.00000
Initial residual norm =    774.59190
Initial residual norm =      9.30283
Initial residual norm =      0.00246
Initial residual norm =      0.00000

Non-linear solution: Temperatures (T1, T2, T3, T4) =
   320.17217
   260.30189
   193.67341
   117.29894
Initial residual norm =      0.00000

Linear solution: Temperatures (T1, T2, T3, T4) =
   305.00000
   235.00000
   165.00000
    95.00000
```


Non-linear Newton-Raphson Convergence
Residual
Iterative convergence
10^{5}
10^{0}
10^{-5}
10^{-10}
10^{-15}
1
1.5
2
2.5
3
3.5
4
4.5
5
Iteration Number