IIT ROORKEE

NPTEL ONLINE CERTIFICATION COURSE

Computer aided power system analysis

Basic Gauss-Seidel numerical method

Prof. Biswarup Das
Department of Electrical Engineering

Power-flow equations

$$P_i = \sum_{k=1}^{N} V_i V_k Y_{ik} \cos(\theta_i - \theta_k - \alpha_{ik})$$

$$Q_i = \sum_{k=1}^{N} V_i V_k Y_{ik} \sin(\theta_i - \theta_k - \alpha_{ik})$$

Non-linear equations

Methods

i) Gauss-Seidel

ii) Newton-Raphson — Polar / Rectangular

iii) Fast Decoupled

Basic Gauss-Seidel numerical method

Suppose there are 'n' unknowns

$x_1, x_2 \text{ ---- } x_n$

There are 'n' non-linear algebraic equations

$$f_1(x_1, x_2, \text{ ---- } x_n) = 0$$

$$f_2(x_1, x_2 \text{ ---- } x_n) = 0$$

$$\vdots$$

$$f_n(x_1, x_2 \text{ ------ } x_n) = 0$$

Example

x_1, x_2, x_3

$$f_1(x_1, x_2, x_3) = x_1^2 x_2 + \tan\frac{x_2 x_3}{x_1} + 5$$

$$f_2(x_1, x_2, x_3) = \sqrt{x_1 x_2 / x_3} + \frac{x_3^2}{x_1^2 + x_2^3} + 6$$

$$f_3(x_1, x_2, x_3) = \cos\left(\frac{x_1}{x_2}\right) + \sin\left(\frac{x_3}{x_1 + x_2}\right) + 10$$

Transform the original equations as

$$x_1 = g_1(x_1, x_2 \cdots x_n)$$
$$x_2 = g_2(x_1, x_2 \cdots x_n)$$
$$\vdots$$
$$x_n = g_n(x_1, x_2 \cdots x_n)$$

Example (contd.)

$$x_1 = \pm\sqrt{-\left(5 + \tan\left(\frac{x_2 x_3}{x_1}\right)\right)/x_2}$$

Second eqn:

$$\frac{x_1 x_2}{x_3} = \left(6 + \frac{x_3^2}{x_1^2 + x_2^3}\right)^2$$

$$\Rightarrow \quad x_2 = \left(6 + \frac{x_3^2}{x_1^2 + x_2^3}\right)^2 \cdot \frac{x_3}{x_1}$$

$$x_3 = (x_1 + x_2)\sin^{-1}A \; ; \; A = -(10 + \cos\frac{x_1}{x_2})$$

We start with some initial guess of the unknowns:

$$x_1^{(0)}, x_2^{(0)}, \dots x_n^{(0)}$$

Then update

$$x_1^{(1)} = g_1\left(x_1^{(0)}, x_2^{(0)} \dots x_n^{(0)}\right)$$

$$x_2^{(1)} = g_2\left(x_1^{(1)}, x_2^{(0)} \dots x_n^{(0)}\right)$$

$$x_3^{(1)} = g_3\left(x_1^{(1)}, x_2^{(1)}, x_3^{(0)} \dots x_n^{(0)}\right)$$

$$\vdots$$

$$x_i^{(1)} = g_i\left(x_1^{(1)}, \dots x_i^{(0)}, x_{i+1}^{(0)} \dots x_n^{(0)}\right)$$

$$\vdots$$

$$x_n^{(1)} = g_n\left(x_1^{(1)}, x_2^{(1)} \dots x_{n-1}^{(1)}, x_n^{(0)}\right)$$

Calculate

$$e_i = \left| x_i^{(1)} - x_i^{(0)} \right|$$

$\forall\, i = 1, 2, \dots n$

max. error = $\max(e_1, e_2, \dots e_n)$

If max. error $< \epsilon$ Then the algorithm converges, otherwise we repeat the process.

Steps of the algorithm

1. Take initial guess $x_1^{(0)}, x_2^{(0)}, \ldots\ldots\ldots x_n^{(0)}$
2. Set iteration count $k = 1$.
3. Update the variables as;

$$x_1^{(k)} = g_1\left(x_1^{(k-1)}, x_2^{(k-1)}, \ldots\ldots\ldots x_n^{(k-1)}\right)$$

$$x_2^{(k)} = g_2\left(x_1^{(k)}, x_2^{(k-1)}, \ldots\ldots\ldots x_n^{(k-1)}\right)$$

$$\vdots$$

$$x_i^{(k)} = g_i\left(x_1^{(k)}, x_2^{(k)} \ldots\ldots x_{i-1}^{(k)}, x_i^{(k-1)}, \ldots\ldots x_n^{(k-1)}\right)$$

$$\vdots$$

$$x_n^{(k)} = g_n\left(x_1^{(k)}, x_2^{(k)}, \ldots\ldots\ldots\ldots x_{n-1}^{(k)}, x_n^{(k-1)}\right)$$

4. Calculate $e_i^{(k)} = \left| x_i^{(k)} - x_i^{(k-1)} \right| \quad \forall\ i = 1, 2, \ldots n$

5. Calculate maximum error as $e_{max}^{(k)} = \max\left(e_1^{(k)}, e_2^{(k)} \ldots e_n^{(k)}\right)$

6. If $e_{max}^{(k)} < \epsilon$, then the algorithm converges and print the result.
else, go to step 7.

7. Increment $k = k+1$

8. Go to step 3

Usually $\epsilon \to 10^{-6}$