Journal of Energy

VOLUME 66 | 2017

journal homepage: http://journalofenergy.com/

MARKO PIKUTIĆ[1] GORAN GRDENIĆ[2] MARKO DELIMAR[2]

marko.pikutic@relatio.hr goran.grdenic@fer.hr marko.delimar@fer.hr

[1] relatio ES Adria d.o.o.

[2] University of Zagreb Faculty of Electrical Engineering and Computing

COMPARISON OF RESULTS AND CALCULATION SPEEDS OF VARIOUS POWER SYSTEM POWER FLOW METHODS

SUMMARY

The theoretical part describes basic power flow methods Gauss-Seidel and Newton-Raphson in their practical forms for solving a load flow problem. In practical part, IEEE test 24, 48 and 72 node networks are used to compare basic methods in terms of calculation speed: on execution of one iteration, entire calculation and on given accuracy influence. Also is analyzed optimal acceleration factor for Gauss-Seidel method and convergences of methods. On the end, final conclusions are obtained after analyzing comparison results.

Keywords: calculation speed, iteration, Load Flow, Gauss-Seidel, Newton-Raphson

1. INTRODUCTION

Power flow or often called load flow calculation is one of the most important calculations in the power system analysis, and the basic calculation for determining the state of the power system. Equations describing power system stationary state form the system of nonlinear equations. Therefore, to determine the state of the system iterative mathematical methods must be applied. Problems that occur during calculations of power flow can be divided into the problem of selecting the most efficient iterative mathematical calculation method and the problem of efficient software program execution of these numerical operations over a limited period. Mostly used methods for solving a power flow problem are Gauss-Seidel and Newton-Raphson method. Number of university textbooks describe and analyze these methods for power flow solution [1-5] and one of the aim of this article is to succinctly and concisely present this methods, so it can serve as additional working material for power system students.

Important features of these methods are number of necessary iterations or convergence rate to the correct solution, and the credibility of the final result (system of nonlinear equations has multiple solutions). Methods are applied on IEEE standard test networks with 24, 48 and 72 nodes [6, 7] and compared with respect to the necessary time for one iteration step (which depends directly on the number of operations performed in iteration), overall calculation time (which along with time for one iteration step also depends on number of iteration steps), accuracy condition and convergence rate.

2. GAUSS – SEIDEL METOD FOR LOAD FLOW CALCULATION

A method for solving nonlinear equations by Gauss-Seidel iteration procedure is also known as a sequential shift method. This method is in fact a complement to the method for solving the linear equations developed by Gauss. The Gauss method calculates all the unknown variables of the equations in the iteration k, and then with these new solutions goes to $k+1$ iteration. Gauss-Seidel's calculate the unknown variables in the iterative step k with all the calculated variables to that point in that step k and the others from the iterative step $k-1$.

Applying Gauss-Seidel method for the solution of power system equations, however, faces certain difficulties. Every node in the network is described by four quantities: voltage magnitude, voltage phase angle, active and reactive power. In every node two of these quantities are known, and two are unknown – and based on these – classification of nodes is obtained:

- load nodes or PQ nodes (active and reactive power are known variables)
- generator or PV nodes (active power and voltage magnitude are known variables)
- reference node (voltage magnitude and phase angle are known variables)

There are two variants of Gauss-Seidel method for solving load flow problem: variant using power system impedance matrix and variant using power system admittance matrix. In the sequence, more used variant with admittance matrix (abbreviated GSY) is described and later applied for solving load flow problem.

Voltage in node i in kth iteration step $\bar{V}_i^{(k)}$ is calculated according to equation:

$$\bar{V}_i^{(k)} = \frac{1}{\bar{Y}_{ii}}\left(\frac{P_i^{sch} - jQ_i^{sch}}{\bar{V}_i^{(k-1)^*}} - \sum_{j=1}^{i-1}\bar{Y}_{ii}\bar{V}_j^{(k)} - \sum_{j=i+1}^{n}\bar{Y}_{ii}\bar{V}_j^{(k-1)}\right), \quad \forall i \neq ref \tag{1}$$

where P_i^{sch}, and Q_i^{sch} are scheduled active and reactive power in specific network nodes, and $\bar{Y}_{ii}$, $\bar{Y}_{ij}$ are the elements of network admittance matrix Y. These voltages are calculated in every network node, except referent node. At the beginning of calculation procedure all voltages are set to initial values equal to 1 p.u. – also called flat start. At the end of every iteration step, accuracy of obtained solution is checked, i.e. difference of voltages in two consecutive iteration steps must be smaller than in advanced given accuracy condition ε:

$$|\bar{V}_i^{(k)} - \bar{V}_i^{(k-1)}| < \varepsilon, \qquad \forall i \neq ref \tag{2}$$

Here, needs to be emphasized that this is simplified description without PV nodes in the grid.

A Gauss-Seidel method for load flow calculation is mostly long-term procedure and many iterations steps are necessary for achieving set accuracy which is typically in the range from 10^{-3} to 10^{-5}. It converges with geometrical speed. An algorithm can be accelerated by using acceleration factor α between two iteration steps for calculating node voltages according to expression:

$$\bar{V}_{i,acc}^{(k)} = \bar{V}_i^{(k-1)} + \alpha \cdot \left(\bar{V}_i^{(k)} - \bar{V}_i^{(k-1)}\right) \tag{3}$$

Instead of using voltage $\bar{V}_i^{(k)}$ in the kth iteration step, accelerated voltage $\bar{V}_{i,acc}^{(k)}$ is used. Attention is necessary for the selection of acceleration factor. Choosing to high value can lead to divergence of solution. An optimal acceleration factor depends upon network configuration and grid's operating point. Typical values are in the range from 1.3 to 1.8 [3].

3. NEWTON-RAPHSON METHOD FOR LOAD FLOW CALCULATION

The second method which is used for solving the system of nonlinear equations is Newton-Raphson method which is based on differential calculus. Starting point for load flow calculation using Newton-Raphson method are expressions for active and reactive power in grid nodes:

$$P_i^{(k)} = \left|V_i^{(k)}\right| \sum_{j=1}^{n} \left|V_j^{(k)}\right| \left|Y_{ij}\right| \cos\left(\delta_i^{(k)} - \delta_j^{(k)} - \theta_{ij}\right), \forall i \; e \; PV, PQ \tag{4.1}$$

$$Q_i^{(k)} = \left|V_i^{(k)}\right| \sum_{j=1}^{n} \left|V_j^{(k)}\right| \left|Y_{ij}\right| \sin\left(\delta_i^{(k)} - \delta_j^{(k)} - \theta_{ij}\right), \forall i \; e \; PQ \tag{4.2}$$

where δ is a phase angle of node voltage and θ is a phase angle of the corresponding element of network admittance matrix which is expressed in polar coordinates. In every iteration step, using node voltages, active power is calculated for all load and generator nodes, which is altogether *PV+PQ* or *n-1* equations, where *n* is a number of network nodes. Reactive power is calculated only in load nodes, which is altogether *PQ* or *n-1-g* equations, where *g* is a number of PV nodes.

Thus calculated active and reactive powers are compared with scheduled or known values:

$$|P_i^{(k)} - P_i^{sch}| < \varepsilon, \qquad \forall i \; e \; PV, PQ \tag{5.1}$$

$$|Q_i^{(k)} - Q_i^{sch}| < \varepsilon, \qquad \forall i \; e \; PQ \tag{5.2}$$

An iterative procedure is finished when given accuracy condition is satisfied. If accuracy condition is not satisfied, it is necessary to approach to calculation of voltages in the next iteration. This is done by using Jacobian matrix or often called just Jacobian.

Jacobian is a matrix of the first partial derivatives of given expressions for active and reactive power in network nodes (5.1 and 5.2):

$$\begin{bmatrix} \Delta P_1 \\ \vdots \\ \Delta P_{n-1} \\ \Delta Q_1 \\ \vdots \\ \Delta Q_{n-1-g} \end{bmatrix} = \begin{bmatrix} \frac{\partial P_1}{\partial \delta_1} & \cdots & \frac{\partial P_1}{\partial \delta_{n-1}} & \frac{\partial P_1}{\partial |V_1|} & \cdots & \frac{\partial P_1}{\partial |V_{n-1-g}|} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_{n-1}}{\partial \delta_1} & \cdots & \frac{\partial P_{n-1}}{\partial \delta_{n-1}} & \frac{\partial P_{n-1}}{\partial |V_1|} & \cdots & \frac{\partial P_{n-1}}{\partial |V_{n-1-g}|} \\ \frac{\partial Q_1}{\partial \delta_1} & \cdots & \frac{\partial Q_1}{\partial \delta_{n-1}} & \frac{\partial Q_1}{\partial |V_1|} & \cdots & \frac{\partial Q_1}{\partial |V_{n-1-g}|} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Q_{n-1-g}}{\partial \delta_1} & \cdots & \frac{\partial Q_{n-1-g}}{\partial \delta_{n-1}} & \frac{\partial Q_{n-1-g}}{\partial |V_1|} & \cdots & \frac{\partial Q_{n-1-g}}{\partial |V_{n-1-g}|} \end{bmatrix} \times \begin{bmatrix} \Delta \delta_1 \\ \vdots \\ \Delta \delta_{n-1} \\ \Delta |V_1| \\ \vdots \\ \Delta |V_{n-1-g}| \end{bmatrix} \tag{6}$$

Jacobian matrix in power flow calculation is usually divided into four submatrices:

$$[J] = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} = \begin{bmatrix} \left(\frac{\partial P}{\partial \delta}\right) & \left(\frac{\partial P}{\partial V}\right) \\ \left(\frac{\partial Q}{\partial \delta}\right) & \left(\frac{\partial Q}{\partial V}\right) \end{bmatrix} \tag{7}$$

so equation (6) in abbreviated form can be written as:

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \times \begin{bmatrix} \Delta \delta \\ \Delta V \end{bmatrix} \tag{8}$$

Changes of voltage phase angles and magnitudes in the *k*th iteration are therefore calculated as:

$$\begin{bmatrix} \Delta \delta_{n-1}^{(k)} \\ \Delta |V|_{n-1-g}^{(k)} \end{bmatrix} = \begin{bmatrix} J_1^{(k)} & J_2^{(k)} \\ J_3^{(k)} & J_4^{(k)} \end{bmatrix}^{-1} \times \begin{bmatrix} \Delta P_{n-1}^{(k)} \\ \Delta Q_{n-1-g}^{(k)} \end{bmatrix} \tag{9}$$

Finally, voltage phase angles and magnitudes in the next iteration are simply calculated:

$$|V_i^{(k+1)}| = |V_i^{(k)}| + \Delta|V|_i^{(k)} \quad \forall i \; e \; PV, PQ \tag{10.1}$$

$$\delta_i^{(k+1)} = \delta_i^{(k)} + \Delta|\delta|_i^{(k)} \quad \forall i \; e \; PQ \tag{10.2}$$

Jacobian matrix and its inverse needs to be calculated in every iteration step what is for large networks (with several hundred nodes) time consuming process which requires large memory allocation. To combat this drawbacks, simplifications of Newton-Raphson method are used.

3.1 Fast Newton-Raphson method

In this variant of Newton-Raphson method, Jacobian matrix is only calculated in the first iteration step. These leads to the obvious advantage of time savings in every subsequent iteration step. On the other side, this can lead to the increased number of iteration steps, increased overall time or even divergence of the method, but in most cases will not. Jacobian matrix can also be calculated afterwards, after several iteration steps, which will decrease divergence probability.

3.2 Decoupled Newton-Raphson method

An influence of small changes of voltage magnitude on nodal active power is negligible, as also small changes of voltage phase angles on nodal reactive power. Jacobian matrix can therefore be simplified on the following way:

$$[J] = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} = \begin{bmatrix} \left(\frac{\partial P}{\partial \delta}\right) & 0 \\ 0 & \left(\frac{\partial Q}{\partial V}\right) \end{bmatrix} \tag{11}$$

Submatrices J_2 and J_3 are zero matrices. Corrections of voltage phase angles and magnitudes can now be calculated in two separate matrix equations:

$$\left[\Delta\delta_{n-1}^{(k)}\right] = [J_1]^{-1} \times \left[\Delta P_{n-1}^{(k)}\right] \tag{12.1}$$

$$\left[\Delta|V|_{n-1-gen}^{(k)}\right] = [J_4]^{-1} \times \left[\Delta Q_{n-1-gen}^{(k)}\right] \tag{12.2}$$

Main advantage of this method is lower memory allocation during calculation [2]. Dimension of matrix which needs to be inverted is halved, but inversion of two matrices must be done instead.

3.3 Fast decoupled Newton-Raphson method

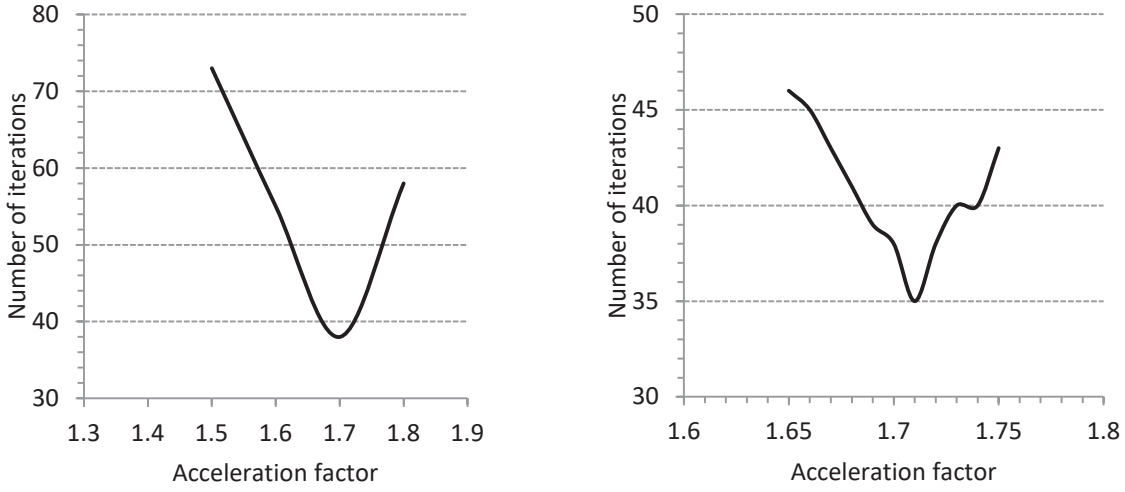
Fast decoupled Newton-Raphson method is simply a combination of two previously described methods. Jacobian matrix is only calculated in the first iteration step, and instead of calculating full Jacobian matrix only J_1 and J_4 submatrices are calculated.

4. COMPARISON OF METHODS FOR LOAD FLOW CALCULATIONS

Load flow calculations were executed on IEEE test network with 24 nodes [10]. Networks with 48 and 72 nodes are made by expansion of IEEE 24 network according to [11].

4.1 An optimal acceleration factor for Gauss-Seidel method

Acceleration factors were varied in Gauss-Seidel method to obtain the minimum number of iterations for given voltage accuracy $\varepsilon=10^{-5}$ p.u. Thus attained acceleration factors were used in subsequent calculations and comparisons. On Figure 1a are presented results for IEEE 24 grid with x-axis resolution of 0.1 for which optimal acceleration factor is equal to 1.7 and on Figure 1b x-axis resolution is 0.01 and thus obtained acceleration factor equals to 1.71. For acceleration factor 1.7 number of iteration steps is 38, while for the acceleration factor 1.71 further reduces to 35. For needs of other calculations, acceleration factor 1.7 for IEEE 24 grid is used. An optimal acceleration factor for 48 and 72 node networks acceleration factor 1.8 is obtained and further used.



(a) (b)

Figure 1: Optimal acceleration factor for 24 node network with given voltage accuracy 10^{-5} [p.u.] ***(a)****: Number of iterations for acceleration factor resolution of 0.1* ***(b)****: Number of iterations for acceleration factor resolution of 0.01*

4.2 Time comparison of one iteration step

Time of one iteration step is an average time of one iteration, obtained as overall time divided by the number of iterations. In the average calculation does not enter zero iteration because in data analysis was shown data it differs significantly from time execution of other iterations. Final iteration is not included also because involves additional operations. Time comparison of one iteration step is shown on

Figure 2 for all three networks and for Gauss-Seidel and different variants of Newton-Raphson method.

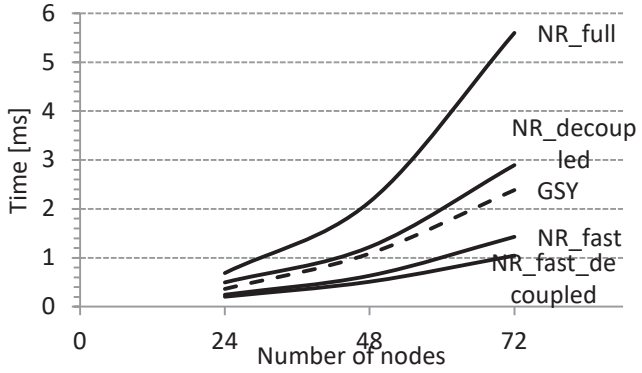


Figure 2: Time comparison of one iteration step for Gauss-Seidel and different variants of Newton-Raphson method

From the figure it can be seen, obviously, that execution time of one iteration rises with number of nodes and more importantly that time execution for Gauss-Seidel method is smaller than for the full Newton-Raphson method. With increasing number of nodes this difference becomes greater. Reason for this lies in the calculation of inverse of Jacobian matrix what requires more time. Calculation of one iteration of decoupled Newton-Raphson method is obviously smaller than for full Newton-Raphson method (instead of calculation of inverse of n-dimension matrix, calculation of two n/2-dimension matrices is necessary), but still greater than for Gauss-Seidel method. Smaller times of one iteration step in relation to Gauss-Seidel method have Fast and Fast-decoupled Newton-Raphson method. However, it needs to be emphasized, that in these cases, necessary Jacobian is only calculated in the zero iteration which does not enter into average calculation. Therefore, time execution with number of nodes rises with slower rate for these two methods.

4.3 Time comparison of overall calculation time

In Gauss-Seidel method optimal acceleration factor determined in subsection 4.1 was used. Voltage ε_U and power accuracies ε_{PQ} were chosen differently in order to make methods comparable. According to [4] voltage accuracy $\varepsilon_U=10^{-5}$ p.u. for Gauss-Seidel method corresponds to power accuracy $\varepsilon_{PQ}=10^{-3}$ p.u. for Newton-Raphson method.

From Figure 3 it can be seen main reason why in power flow calculations Newton-Raphson method is mostly used. Although time of one iteration is smaller for Gauss-Seidel method, overall time is much greater because of necessary number of iteration steps to reach required accuracy. This is especially emphasized in the grids with higher number of nodes where number of iterations for Gauss-Seidel method increases, while for the Newton-Raphson method remains nearly the same. Overall execution time for Fast and Fast-Decoupled Newton-Raphson is the least, but interestingly overall execution time for Decoupled Newton-Raphson is greater than for full Newton-Raphson method. A cause again lies in the greater number of

iterations. Nevertheless, decoupled Newton-Raphson has some other advantages, such as necessary memory demand.

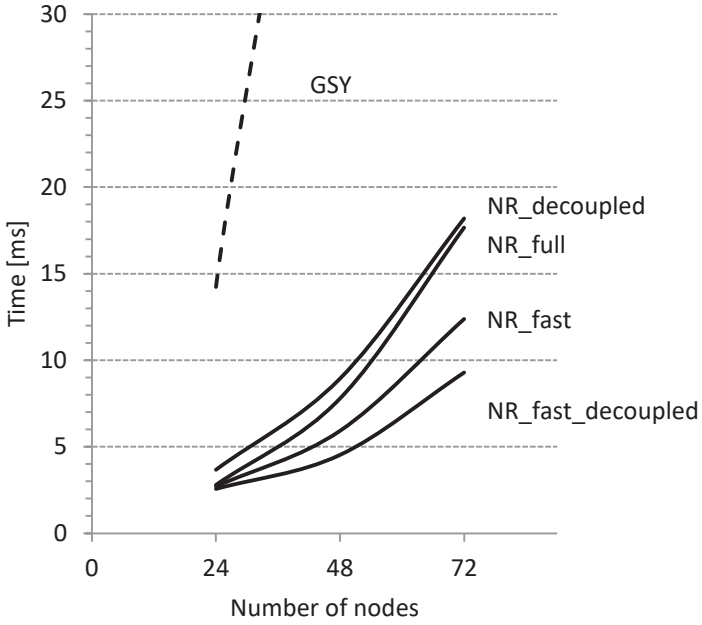


Figure 3: Time comparison of overall calculation time for Gauss-Seidel and different variants of Newton-Raphson method

4.4 Convergence comparison

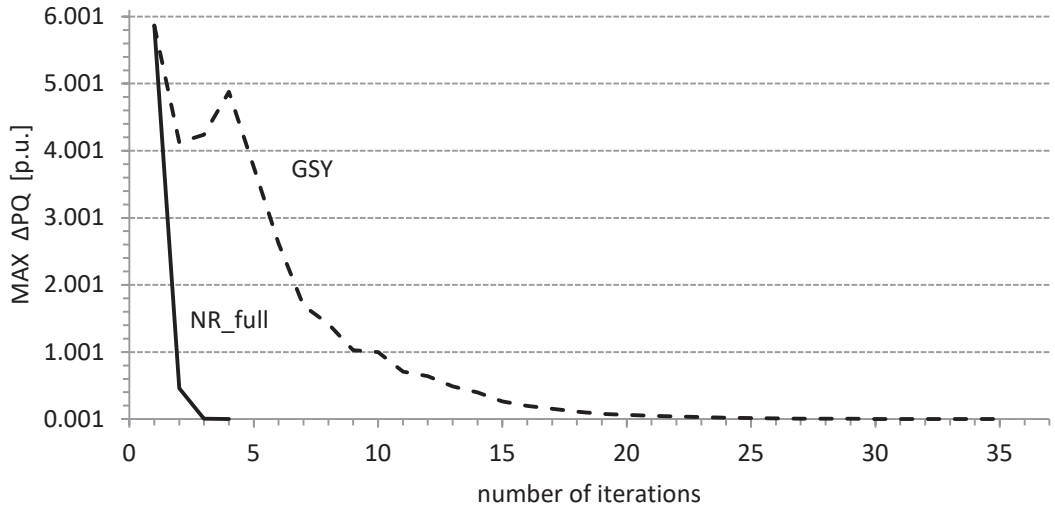


Figure 4: Convergence comparison for Gauss-Seidel and Newton-Raphson method

Comparison of convergence was performed using same power accuracy condition equal to 10^{-3} p.u. This is somewhat different than in previous subsection where for Gauss-Seidel equivalent voltage accuracy was used. Iterations were executed in Gauss-Seidel method until power accuracy was satisfied. Thus, two

more calculations – of active and reactive power – were performed in each iteration step in Gauss-Seidel method. Given accuracy for Newton-Raphson is obtained in four iterations, and for Gauss-Seidel in 35 iterations. On Figure 4 quadratic convergence of Newton-Raphson and geometric convergence of Gauss-Seidel can be observed. It is interesting to notice that Gauss-Seidel at some point starts to diverge, but soon after takes the right direction. One of the possible reasons for this is using acceleration factors. While approaching correct solution, acceleration factors could run away voltage iterations from the correct solution.

4.5 Calculation time dependency on given accuracy

For both methods calculations for given power accuracy in the range from 10^{-1} to 10^{-4} p.u. were conducted on network with 24 nodes.

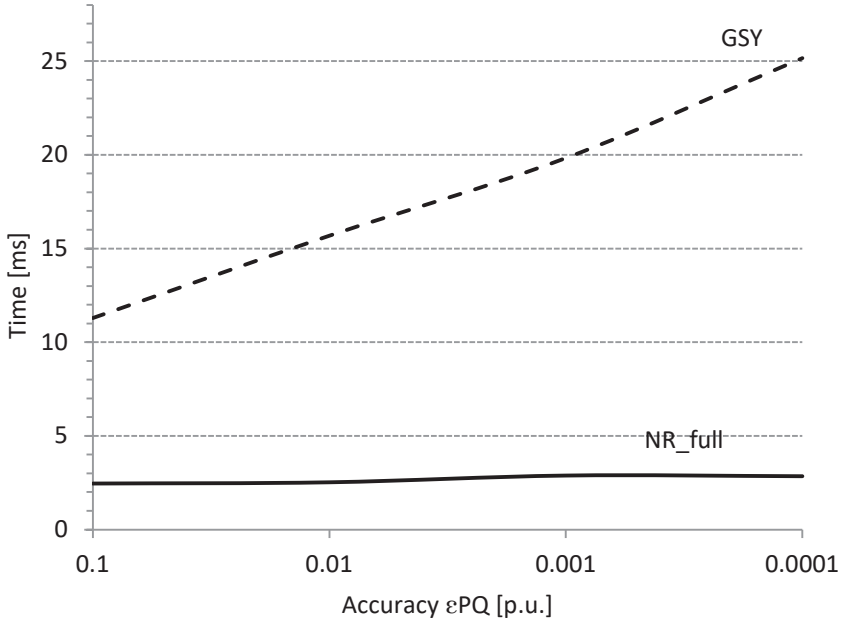


Figure 5: Calculation time dependency on given accuracy

From the Figure 5 it can be seen that overall calculation time for Gauss-Seidel method increases linearly with increasing given accuracy, while for the Newton-Raphson method it does not have large influence. With increasing accuracy Gauss-Seidel method requires greater number of iterations, i.e. converges more slowly which increases overall computation time. On the other side, Newton-Raphson reaches greater accuracy very fast.

4.6 Case of voltage deviation from the allowed limits

According to grid rules which are valid for certain power system, normal system operation state presumes node voltages inside allowable range. In case of a disturbance in the system, e.g. big load switching on or off, line outage, generator outage and similar, voltages may not be any more in allowable limits. Higher deviations appear in case of a fault, most often short-circuit occurrence, when voltage deviation of faulted node become incomparable to other voltages

(theoretically, in case of a direct short-circuit, node voltage becomes zero). This also has great impact on power flow calculation. A short-circuit was simulated on the IEEE 24 grid by changing corresponding diagonal element of admittance matrix. Conditions on short-circuit location are very low impedance, or respectively, very high admittance.

Short-circuit was simulated in node 20 and results of calculations are shown in Figure 6. A Newton-Raphson method has not reached final solution (it diverged – mark NR: U_{20}=1 p.u. on the Figure 6), while the Gauss-Seidel has converged to the solution. Improvements can be made during initialization of voltage variables by setting voltage of faulted node closer to zero. When setting faulted node voltage to 0.1 p.u., which corresponds to value obtained by Gauss-Seidel method, Newton-Raphson also converged to the correct solution (solid line on Figure 6). It can be concluded that Gauss-Seidel method converges better to solution when initial variable values are further from the correct solution. Therefore, often in power flow calculations, Gauss-Seidel and Newton-Raphson are used together: Gauss-Seidel in the first several iterations – to ensure convergence of iteration process, and Newton-Raphson to the end of calculation – to achieve greater speed.

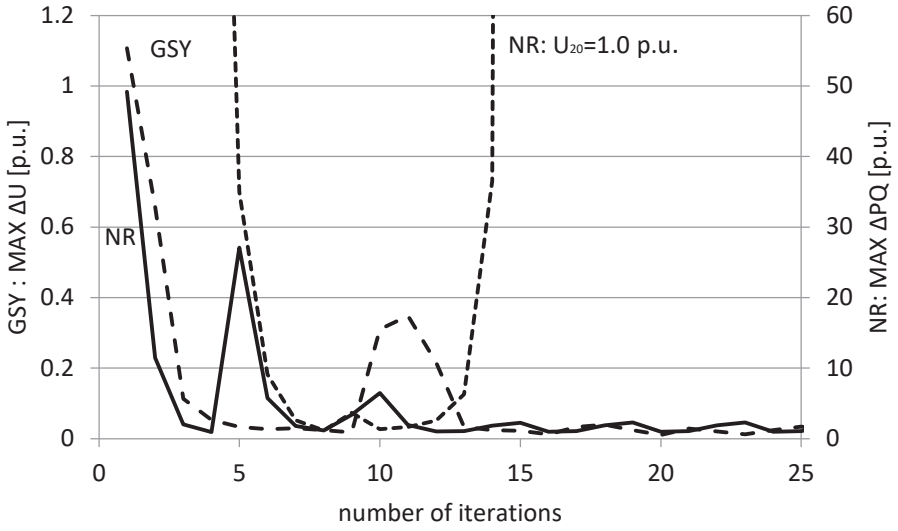


Figure 6: Convergence of a Gauss-Seidel and Newton-Raphson method in case of a short-circuit occurrence

5. CONCLUSION

For networks with small number of nodes, iteration and total calculation times are in comparable domain, and to make specific decision about which method to use can depend upon other things. Decision can be made by the impact of the method on the requirements on the hardware and software support. An advantage of Gauss-Seidel method is its simpler performance, as well as the lower requirement in terms of memory space allocation for execution. A time of one iteration step is also faster in the Gauss-Seidel method, but it will require a greater number of iterations to reach final solution due to geometric convergence. On the other side, Newton-Raphson method converges squarely in the total time of the calculation,

and hence overall computation time is reduced. However, in specific cases, like short-circuit simulations, Gauss-Seidel nevertheless benefits when initial conditions are far from finite solution. On contrary, a Newton-Raphson method is highly dependent on initial input conditions. Simplified Newton-Raphson methods can also achieve faster times of total calculation, and by increasing the number of nodes, they have a significant advantage over the Gauss-Seidel method.

In real-world power management, the choice of method will depend on the specificity of the problem being analyzed, as well as on the importance and responsibility behind the results of the method. Today, the importance of accuracy and speed of achieving results plays a major role in the emergence of new, dynamic participants in the power grid. Smart grids, electricity market, energy efficiency systems are just some of the areas where it is important to operate and make decisions with the results obtained.

REFERENCES

[1] M. Ožegović and K. Ožegović, *Električne energetske mreže IV*. FESB, Split, 1999.

[2] D. P. Kothari and I. J. Nagrath, *Modern Power System Analysis (Fourth Edition)*. Tata McGraw-Hill Education Pvt. Ltd, 2011.

[3] J. J. Grainger and W. D. Stevenson Jr, *Power System Analysis*. McGraw-Hill Book Co.-Singapore, 1994.

[4] H. Saadat, *Power System Analysis*. McGraw-Hill Book Co.-USA, 1999.

[5] G. W. Stagg and A. H. El-Abiad, *Computer methods in power system analysis*, McGraw-Hill Inc, 1968.

[6] Reliability Test System Task Force, "IEEE RELIABILITY TEST SYSTEM", *IEEE Trans. on Power Apparatus and Systems*, Vol. PAS-98, No. 6 Nov. /Dec. 1979.

[7] Reliability Test System Task Force, "The IEEE Reliability Test System – 1996", *IEEE Trans. on Power Systems*, Vol. 14, No. 3, August 1999.