DINCON 2019 ID: 1954
XIV CONFERÊNCIA BRASILEIRA DE DINÂMICA, CONTROLE E APLICAÇÕES
25 de novembro a 27 de novembro de 2019 – São Carlos/SP

Application and Comparison of Numerical Optimization Methods applied to Power System State Estimation

Julio Augusto Druzina Massignan[1]
João Bosco Augusto London Junior
Escola de Engenharia de São Carlos, USP, São Carlos, SP
Rafael Bernardo Zanetti Cirino
Universidade Federal de Lavras, Lavras, MG

Abstract. Power System State Estimation (PSSE) is one of the main tools for energy management systems, responsible for gathering information of the network and evaluating the real time operating condition of the system. This work presents the application and comparison of nonlinear optimization methods and the classical algorithm based on the Gauss-Newton Method, applied to PSSE. Simulations were made using the IEEE benchmarking test systems and performance of the methods were evaluated in terms of computational efficiency, numerical precision and implementation. The main goal of this work is to evaluate the nonlinear optimization algorithms applied to PSSE and also to reduce the distance from optimization researchers to power systems engineers in the context of power grids monitoring.

Keywords. Power Systems, State Estimation, Optimization, Inference, Measurements.

1 Introduction

Since its conception in the early 70's, PSSE has consolidated its presence in transmission systems operation. It is responsible for the real time monitoring of the power system, providing reliable information about its operational condition. This is accomplished by obtaining the state variables of the power system [1]. PSSE is based on a set of redundant real time measurements obtained from the SCADA system, and the nonlinear measurement model that relates those measurements with the state variables.

Essentially PSSE is formulated as an optmization problem in the sense of minimization of the measurement model residue [1]. The large scale of power systems (it can have from thousands to hundred of thousands state variables) and the need of fast calculation (since it is a real time application) are key aspects that a state estimator must be able to deal with. Numerical aspects are also of major importance in PSSE, such as sparsity and ill-conditioning problems. This has driven researches related to nonlinear optmization and for more efficient numerical methods applied to PSSE [2–4].

[1]julio.massignan@usp.br

2 PSSE Formulation and Solution

Given a power system with m measurements and n state variables, the state estimation problem is formulated using the following measurement model [1]:

$$z = h(x) + \varepsilon \tag{1}$$

where $z \in R^m$ is the measurement vector with the observed values obtained from the SCADA system; $x \in R^n$ is the state variables vector composed of the voltage magnitude and voltage angle at each node of the power system; $h(x) : R^n \rightarrow R^m$ is the set of nonlinear equations that relates the measurements with the state variables; and $\varepsilon \in R^m$ is the error vector assumed as independent normally distributed random variables, with zero mean and known covariance matrix W^{-1}.

The measurement model equations are given accordingly to the type of electrical quantity being measured, and are typically nonlinear functions related to power flows [1]. The state variables are defined as the complex nodal voltages at each node of the system $x = [V, \theta]'$. The active and reactive power flow (p_{km} and q_{km}) trough a branch of the power system, i.e. from node k to node m, can be expressed by:

$$p_{km}(x) = g_{km}.V_k^2 - V_k.V_m.(g_{km}.\cos(\theta_k - \theta_m) + b_{km}.\sin(\theta_k - \theta_m)) \tag{2}$$

$$q_{km} = -(b_{km} + b_{km}^{sh}).V_k^2 - V_k.V_m.(g_{km}.\sin(\theta_k - \theta_m) + b_{km}.\cos(\theta_k - \theta_m)) \tag{3}$$

where, g_{km} and b_{km} are the real and imaginary parts of the series admittance and b_{km}^{sh} the parallel admittance of branch $k - m$ (network electrical parameters); θ_k and θ_m the voltage angle of nodes k and m (state variables); and V_k and V_m the voltage magnitude of nodes k and m (state variables). Power injections are expressed as a sum of power flows.

Given the statistical nature of the model, the joint probability function is:

$$f(\varepsilon|x) = f(\varepsilon_1|x, ..., \varepsilon_m|x) = \prod_{i=1}^{m} f(\varepsilon_i|x) = \prod_{i=1}^{m} \frac{1}{2\pi\sigma_i^2} \exp^{-\frac{(z_i - h_i(x))^2}{2\sigma_i^2}} \tag{4}$$

where σ_i is the standard deviation of the measurement i related to the precision of the measurement devices. The maximum likelihood estimator for the state vector x can be found by the following unconstrained optimization problem:

$$\min J(x) = \frac{1}{2} \sum_{i=1}^{m} \frac{(z_i - h_i(x))^2}{\sigma_i^2} = \frac{1}{2}[z - h(x)]'.W.[z - h(x)] \tag{5}$$

The gradient vector of the above minimization problem is given by:

$$\nabla J(x) = -H(x)'.W.[z - h(x)] \tag{6}$$

where $H(x)$ is the Jacobian matrix of the nonlinear model $h(x)$.

And the Hessian matrix of the above minimization problem is given by:

$$\nabla^2 J(x) = H(x)'.W.H(x) - \sum_{i=1}^{m} \frac{(z_i - h_i(x))}{\sigma_i^2}.\frac{\partial^2 h_i(x)}{\partial x^2} \tag{7}$$

The classical approach for solving the state estimation problem, i.e, finding the optimal value of x, is based on the linearization of $h(x)$ in equation (5) and using the iterative Gauss-Newton method [1]. The linearization of $h(x)$ is obtained trough the first order term of its Taylor expansion around a point x^k. The optimal value of x is then calculated by finding the optimal solution of (5) trough the first-order optimality condition:

$$H(x^k)'.W.H(x^k).\Delta x = H(x^k)'.W.[z - (h(x^k)] \tag{8}$$

where $\Delta x = (x^{k+1} - x^k)$ is the update value of x for the next iteration k and $G(x) = H(x^k)'.W.H(x^k)$ is known as the Gain matrix of the state estimator. The solution of the above equation is generally obtained trough a direct method for linear systems, such as LU or QR factorization.

The second-order optimality condition in this case is given by $H(x^k)'.W.H(x^k) > 0$. If the system is observable, then $H(x)$ is full rank [5]. Also the covariance matrix is always definite positive. Thus, a solution of (8) always satisfies the second order optimality condition. The method is iterative in the sense that equation (8) is calculated sequentially. Convergence is achieved when two successive values of x have its difference less than a pre-specified tolerance $\|\Delta x\|_\infty \leq tol$

3 Numerical Optimization Methods

3.1 Linear Search Methods

Linear Search Methods starts at a point x_0 and for a given descent direction p_0 performs a linear search, i.e., the next point x_1 will be given by (9) for some scalar α_k.

$$x_{k+1} = x_k + \alpha_k p_k \tag{9}$$

The Inexact Linear Search Method starts with an initial step α_0 and verifies if it has an sufficient decrease, given the the Armijo condition as shown in expression (10) [6].

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f_k^\top p_k \tag{10}$$

Where c_1 is the Armijo's constant, strictly positive and usually small. This condition assures that not only the step will provide a point with a lower objective function value, but also assures that this difference will be at least $c_1 \alpha \nabla f_k^\top p_k$ smaller, i.e., the algorithm will not drift around a vicinity of x_k. The Armijo's condition provides a test method for a given value of α. But if a given α_0 fails the test, a new α_1 should be provided. Since the direction p_k is descending at least for some small value of α this condition is satisfied.

Gradient Method:

The Gradient Method is a first-order Local Search Method. The choosen direction for this method is $p_k = -\nabla f(x_k)$, which is a descent direction, as shown in (11).

$$\nabla f(x_k)^\top p_k = -\nabla f(x_k)^\top \nabla f(x_k) = -\|\nabla f(x_k)\|^2 < 0 \tag{11}$$

Modified Newton Method:

The Newton Method is a second-order Local Search Method, characterized by the choice of direction given by (12), where $\nabla^2 f(x_k)$ is the Hessian Matrix. This direction, unlike the Gradient direction, could in fact not be descending. If $\nabla f(x_k)^\top \nabla^2 f(x_k) \nabla f(x_k) < 0$ for a given x_k this is a descent direction. Therefore, if the Hessian is positive define, (12) holds for every x_k.

$$p_k = -\nabla^2 f(x_k) \nabla f(x_k) \tag{12}$$

The Modified Newton Method uses the enhancement provided by the second-order information for the Newton Step, trying to avoid the problems related to some points which the direction is not descent. Some options are available: one could apply some transformation on the Hessian Matrix to make it sufficiently positive define; or use a positive define approximation of the Hessian Matrix, instead of the Hessian itself. For this method the direction is given by $p_k = -B\nabla f(x_k)$, where B is a hessian approximation.

3.2 Trust Region Methods

Trust Region Methods, like the Linear Search Methods, produces a decreasing sequence $[f(x_k)]_{k\to\infty}$ that stops when reaches an inflection point of f. Both methods rely on the quadratic approximation model m_k of the function f in the vicinity of x_k. However, the Trust Region Methods defines a region where m_k is considered to be a good approximation, usually this is an open ball of radius Δ_k. The problem can be formulated as (13).

$$\begin{aligned} \min \qquad & m_k(p) = f(x_k) + \nabla f(x_k)^\top p + \frac{1}{2} p^\top B_k p \\ & \text{subject to:} \qquad \|p\| \leq \Delta_k \end{aligned} \tag{13}$$

Where B_k is a symmetrical matrix. It is necessary to highlight that the step size and direction are calculated simultaneously by this method [6].

Cauchy Step:

The Cauchy Step for the Trust Region method is based on a linear approximation of the quadratic model to obtain a direction p_k^S. This direction is the maximum scalar of the gradient that does not violates the radius restriction. Then the quadratic model is solved for $\tau_k > 0$ in that direction. If ${p_k^S}^\top B_k p_k^S > 0$, then the function m_k is a convex quadratic. In that case τ_k is either the convex quadratic unconstrained solution or it is equal to 1, which occurs first. Finally the Cauchy Step p_k^C will be given as (14).

$$p_k^C = -\tau_k \frac{\Delta_k}{\|\nabla f(x_k)\|} \nabla f(x_k) \tag{14}$$

Where τ_k is given as (15).

$$\tau_k = \begin{cases} 1, & \text{if } {p_k^S}^\top B_k p_k^S \leq 0 \\ \min\{1, \frac{\|\nabla f(x_k)\|^3}{\Delta_k \nabla f(x_k)^\top B_k \nabla f(x_k)}\}, & \text{otherwise} \end{cases} \tag{15}$$

Dogleg Step:

For the Dogleg Step the matrix B_k needs to be positive define. With the unconstrained solution for (13) and taking the first order expansion of f as the approximation model, two directions are obtained to build the Dogleg Step, namely p_k^B and p_k^U. In case the unconstrained solution for the quadratic approximation is within the trust region ($\|p_k^B\| \leq \Delta_k$), then this step is used. Otherwise, if the solution for the linear approximation is also outside the trust region, the the direction will be $\frac{\Delta_k}{\|p_k^U\|} p_k^U$, which is the greatest scalar in the p_k^U direction that is within the trust region. Finally, in case the linear approximation solution p_k^U is within the trust region and the quadratic solution p_k^B is outside, then the Dogleg Step will be a combination of p_k^U and p_k^B. Geometrically, the Dogleg Step will step the full p_k^U step and then combine it with the p_k^B direction with the maximum scalar that keeps the full step inside the trust region. The Dogleg Step p_k^D is given as (16).

$$p_k^D = \begin{cases} p_k^B, & \text{if } \|p_k^B\| \leq \Delta_k \\ \frac{\Delta_k}{\|p_k^U\|} p_k^U, & \text{if } \|p_k^B\| > \Delta_k \text{ and } \|p_k^U\| \geq \Delta_k \\ p_k^U + \tau(p_k^B - p_k^U), & \text{otherwise} \end{cases} \tag{16}$$

Where τ is the smallest solution for the given quadratic polynomial (17).

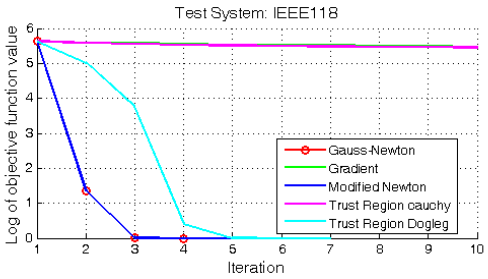
$$\|p_k^B - p_k^U\|^2 \tau^2 + \langle 2p_k^U, p_k^B - p_k^U \rangle \tau + \|p_k^U\|^2 - \Delta_k^2 = 0 \tag{17}$$

4 Simulation Results and Discussion

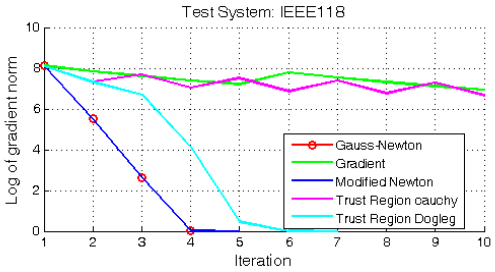
The test systems used in this work are the IEEE benchmarking energy transmission systems: IEEE14, IEEE30, IEEE57 and IEEE118 with all possible voltage magnitudes, power flows and injections measurements. All simulations were run in a i7-4500U 1800 GHz with 8GB RAM. All algorithms were implemented in C and a limit of 100 iterations for each was fixed for analysis. A tolerance of 10^{-5} was used for all test cases. For the Gradient Method the initial step was set in 10 and for the Newton Method it was 1. The Armijo's constant c_1 was set as 10^{-3}. All methods use as initial value of x the *flat start*, i.e. all voltage magnitude of the system are set to 1.0 pu and 0 degrees.

Initially the convergence of each method is under investigation, by evaluating the value of the objective function f and the gradient norm $\|\nabla f\|$ for all test systems for each iteration. Neither the Gradient Method nor the Trust Region with Cauchy Step converges under 100 iterations. Take note that those approaches behave like a first-order method, while the Newton and Dogleg Step behave like a second-order method. Figure 1 shows the results for function value and gradient norm on IEEE118.

Similar convergence characteristics were obtained for the other test systems. Note that the Gauss-Newton reaches convergence in 4 iterations, while Modified Newton reaches in



((a)) Log of Function Value



((b)) Log of Gradient Norm

Figura 1: Performance of each method on IEEE118 Test Power System

5 iterations. The next algorithm to converge was the Trust Region with Dogleg Step, taking 7 iterations. It should be highlighted that for the IEEE118 the Gauss-Newton and Modified Newton differ on the number of iterations to converge, what is related to the convergence criterion of each method. The first uses the norm of subsequent updates of x, while the second has its convergence based on the norm of the gradient.

In Table 1 some performance metrics are shown for each algorithm for all four test system. The number of iterations (#It), objective function evaluation (#F), gradient evaluation (#G), hessian evaluation (#H) and time in seconds are shown.

Tabela 1: Performance Evaluation Results

Method		Gauss-Newton	Gradient*	Mod. Newton	TR Cauchy*	TR Dogleg
IEEE14	#It	4	100	4	100	5
	#F	4	2011	10	200	10
	#G	4	100	4	200	10
	#H	4	0	4	100	4
	Time	<0,001	0,062	<0,001	0,016	<0,001
IEEE30	#It	4	100	4	100	5
	#F	4	2265	10	200	10
	#G	4	100	4	200	10
	#H	4	0	4	100	4
	Time	0,094	0,36	0,078	2,281	0,094
IEEE57	#It	4	100	4	100	7
	#F	4	2373	10	200	14
	#G	4	100	4	200	14
	#H	4	0	4	100	6
	Time	0,656	0,672	0,482	14,939	0,922
IEEE118	#It	4	100	5	100	7
	#F	4	2776	13	200	14
	#G	4	100	5	200	14
	#H	4	0	4	100	6
	Time	13,673	2,36	14,25	339,465	20,925

Some values have a visible discrepancy between methods, like the #F which is used

twenty fold by the Gradient Method when compared to the Modified Newton or the Trust Region with Dogleg Step. The reason behind those high number is explained by the backtracking algorithm for the Gradient Method. Although the number of evaluation of the Hessian Approximation could indicate a high use of second order information, it's not that direct. Taking into account the number of evaluations by the Trust Region with Cauchy Step is equal for each iteration to the Trust Region with Dogleg Step, the first behaves like the Gradient Method in terms of convergence. This is due to lack of extrapolation of this information beyond the Trust Region Radius, hence offering a crude linearization in a small vicinity of a given point.

5 Conclusion

This work presented approaches of nonlinear optimization for the PSSE problem, tested on IEEE benchmark test power systems. Two distinct Numerical Optimization frameworks were used (Linear Search and Trust Region) and for each one two methods were implemented, one with first order and another with second order information. The classical method, Gauss-Newton, is used for comparison. It is shown that the incorporation of second order information highly improves the convergence rate of the method, with a marginal increasing on the CPU process time. We hope to make improvements with special treatment for sparsity and algorithm optimization in a future work.

The authors acknowledge CAPES, CNPq and FAPESP (grant no. 2016/19646-6) for the finacial support.

Referências

[1] A. Abur and A. G. Expósito, *Power System State Estimation: Theory and Implementation*. Nova York: CRC Press, 2004.

[2] A. S. Costa, R. Salgado, and P. Haas, "Globally convergent state estimation based on givens rotations," in *2007 iREP Symposium - Bulk Power System Dynamics and Control - VII. Revitalizing Operational Reliability*, pp. 1–9, Aug 2007.

[3] S. Pajic and K. A. Clements, "Power system state estimation via globally convergent methods," *IEEE Transactions on Power Systems*, vol. 20, pp. 1683–1689, Nov 2005.

[4] H. Zhu and G. B. Giannakis, "Power system nonlinear state estimation using distributed semidefinite programming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 1039–1050, Dec 2014.

[5] J. B. A. London, S. A. R. Piereti, R. A. S. Benedito, and N. G. Bretas, "Redundancy and observability analysis of conventional and pmu measurements," *IEEE Transactions on Power Systems*, vol. 24, pp. 1629–1630, Aug 2009.

[6] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.