Electrical Power and Energy Systems 27 (2005) 614–619

ELECTRICAL POWER & ENERGY SYSTEMS

www.elsevier.com/locate/ijepes

Slack bus treatment in load flow solutions with uncertain nodal powers

Aleksandar Dimitrovski *, Kevin Tomsovic

School of EECS, Washington State University, Pullman, WA 99164, USA

Abstract

This paper addresses the problem introduced by the slack bus in load flow solutions with uncertain nodal powers. While balancing powers in the system the slack bus will also absorb all uncertainty. The results obtained are of no practical interest unless realistic constraints are imposed on slack power production/consumption. Two methods of dealing with these constraints are investigated suitable for implementation within the recently developed boundary load flow.

Keywords: Fuzzy sets; Load flow analysis; Power system planning; Slack bus; Uncertainty.

1. Introduction

The most common formulation of the load flow problem requires all input variables (PQ at loads, PV at generators) to be specified as deterministic ('crisp') values. Each set of specified values corresponds to one system state, which is deemed representative for some set of system conditions. Thus, when the input conditions are uncertain, as is predominantly the case in planning, there is a need for numerous scenarios to be analyzed. A load flow approach that could directly incorporate uncertainty into the solution process has been long recognized as useful. The results from such analysis would be expected to give solutions over the range of the uncertainties, i.e. solutions that are sets of values or regions instead of single operating points.

To date, two families of uncertain load flow algorithms have evolved. The first one is the probabilistic load flow (PLF), which considers loads and generations as random variables with some probability distributions (e.g. [1–4]). The results of the load flow, i.e. voltages, power flows, and so on, are also random variables with resultant probability distributions obtained using probabilistic techniques. The second is the fuzzy load flow family of algorithms, where input variables are represented as fuzzy numbers (e. g. [5–7]). Fuzzy numbers are described by possibility distributions and can be considered to be intervals with indistinct boundaries. The results obtained are also fuzzy numbers with resultant possibility distributions. The authors have recently extended these concepts to the so-called boundary load flow [8,9].

Both families of uncertain load flow algorithms use the same definition of the problem as the traditional deterministic approach. That is, load buses are defined as PQ buses, generator buses as PV buses, and one bus is assumed to be a slack bus to balance the active and reactive power in the system. The 'slack' bus (or 'swing' bus) is defined as Vθ bus. While this definition of the load flow problem is appropriate for a deterministic solution (although it may still be helpful to define a distributed 'slack' among several buses), it has an inherent drawback when dealing with uncertain input variables: the slack bus must absorb all uncertainties arising from the solution and thus, will have the widest nodal power possibility (probability) distributions in the system. If even moderate amounts of uncertainty are allowed in a large system, the resulting distributions will frequently contain values well beyond the generating margins of the slack generator.

This problem has been neglected so far in the literature except for the case of a linearized fuzzy DC load flow [7]. In that work, three approaches, conceptually the same, use an iterative corrective procedure in order to satisfy constraints imposed on the slack bus. Recently, the authors have developed a methodology that enables an accurate solution from a non-linear AC fuzzy load flow [8]. It follows the concept of boundary load flow (BLF) solutions, where solutions are based on an optimization procedure for implicitly defined vector functions. Numerical results obtained from test systems have shown the feasibility of this approach, but they also have shown the problems associated with the inappropriate definition of the slack bus.

This paper extends the previous work and investigates different ways of incorporating the constraints imposed on the

* Corresponding author. Tel.: +1 509 335 8001; fax: +1 509 335 3818.
E-mail address: dimitrov@wsu.edu (A. Dimitrovski).

0142-0615/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijepes.2005.08.015

slack bus in the framework of boundary load flow solutions. Two methods of dealing with this problem are considered: (1) slack bus to PV bus and PV bus to slack bus conversion, and (2) distributed slack bus modeling. The results obtained from different test systems as well as the specifics in different approaches are discussed and compared.

2. Boundary load flow solutions

The BLF was presented for the first time in [2] within the context of PLF. In that paper, an approximate solution for the ranges of values for state and output variables, given the ranges of values of input variables from their probability distributions, was found. The ranges of variables were then used to determine multiple points of linearization for the load flow equations in order to improve the accuracy of the PLF solutions, particularly for the tail regions of the probability distributions.

The authors have developed a methodology, where an accurate solution for a non-statistical interval load flow is obtainable [8]. In the following, a brief explanation of this methodology is given.

The load flow problem is defined by two sets of nonlinear equations:

$$\mathbf{Y} = \mathbf{g}(\mathbf{X}) \tag{1}$$

and

$$\mathbf{Z} = \mathbf{h}(\mathbf{X}), \tag{2}$$

where:

X is the vector of unknown state variables (voltage magnitudes and angles at PQ buses; and voltage angles and reactive power outputs at PV buses),
Y is the vector of predefined input variables (real and reactive injected nodal powers at PQ buses; and voltage magnitudes and real power outputs at PV buses),
Z is the vector of unknown output variables (real and reactive power flows in the network elements), and
g, **h** are the load flow vector functions.

The boundary values are the extreme points found by allowing the inputs to vary over their range. In our notation, we want to find the extreme values for the elements of **X** and **Z** implicitly expressed in (1) and (2), in terms of the elements of **Y** which, in turn, are constrained. Thus, finding the boundary values in a load flow problem is a process of locating the constrained extrema of implicitly defined vector functions of vector arguments.

Because **X** cannot be explicitly expressed in terms of **Y**, the solution of the system of equations (1) is found by an iterative process. Given an initial trial solution, $\mathbf{X}'$, the error is calculated as:

$$\Delta\mathbf{Y} = \mathbf{Y} - \mathbf{Y}' = \mathbf{Y} - g(\mathbf{X}'). \tag{3}$$

If a Newton–Raphson (N–R) based scheme is used, (1) is linearized around $\mathbf{X}'$ and an update for the new solution is found as:

$$\Delta\mathbf{X} = \mathbf{K}\cdot\Delta\mathbf{Y}, \tag{4}$$

where **K** is the inverse of the Jacobian of **g** evaluated at $\mathbf{X}'$. The element K_{ij} of this matrix is the partial derivative of X_i with respect to Y_j. Similarly, if we linearize (2) and substitute for $\Delta\mathbf{X}$ from (4) we will obtain:

$$\Delta\mathbf{Z} = \mathbf{S}\cdot\Delta\mathbf{X} = \mathbf{L}\cdot\Delta\mathbf{Y}, \tag{5}$$

where **S** is the Jacobian of **h** at the given point of linearization. The matrix $\mathbf{L}=\mathbf{S}\cdot\mathbf{K}$ is a sensitivity coefficient matrix and the element L_{ij} is the partial derivative of Z_i with respect to Y_j.

Each row of **K** and **L** represents the gradient vector of the corresponding state and output variable X_i and Z_i, respectively. Similar to derivative based optimization procedures, by iteratively following the direction of the gradient, extreme points (possibly local) of the state or output variable can be found.

Only the signs of the partial derivatives that comprise the gradient are used in the solution since our experience has shown that the values of the partials are not useful for efficiently determining the updates. Further, a procedure is needed to maintain feasibility of the solution, i.e. ensure the input variables remain within the constraints for all iterations. The iterative procedure is reviewed in the following.

Suppose that the *minimum* value of X_i is sought. If K_{ij} is positive (negative), then decrease (increase) the value of Y_j by some fixed step size. After repeating for all Y_j we obtain a new point of **Y** from which a new **X** from (1) can be found. From this new point, the above steps are repeated until one of the following is true for all input variables:

- the partial derivative is positive and the associated variable is at a minimum;
- the partial derivative is negative and the associated variable is at a maximum;
- the partial derivative is zero.

If the final condition does not hold for any variable, then the solution is clearly a local constrained extremum. Because of the nonlinearity of (1) and (2), this point may not be the only extrema. In practice, we have found the physical nature of the load flow problem leads to either a unique solution or a relatively small number of extrema.

When one or more of the partial derivates are zero, the solution point lies somewhere on the boundary surface. Such a point is either a local constrained extremum or a saddle point. Though it is unlikely that by preceding in a downhill direction one will end up trapped in a local maximum or a saddle point, theoretically such a possibility exists. Here, previous values of X_i are recorded at each step and if X_i fails to decrease, then the step length is modified.

Finally, in the special case when all the partial derivatives are zero, a solution cannot be obtained due to the singularity of the Jacobian. Such a point typically indicates infeasibility of the load flow and a loading limit for the system considered. A singularity of the Jacobian may also occur even if not all of the partial derivatives are zero. In such cases, the ranges of values of the input variables are too great and one must repeat the calculations with reduced variations for some or all of the