Optimization in Electrical Distribution Systems: Discrete Ascent Optimal Programming

by

Paul A. Dolloff

Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

Approved:

R. Broadwater, Chairman

T. Herdman

S. Rahman

L. Mili

Y. Liu

February, 1996

Blacksburg, Virginia

Optimization in Electrical Distribution Systems: Discrete Ascent Optimal Programming

by

Paul A. Dolloff

Dr. Robert P. Broadwater, Chairman

Bradley Department of Electrical Engineering

(ABSTRACT)

This dissertation presents a new algorithm for optimal power flow in distribution systems. The new algorithm, Discrete Ascent Optimal Programming (DAOP), will converge to the same solution as the Lagrange multiplier approach as demonstrated by example. An intuitive discussion illustrating the path of convergence is presented along with a theorem concerning convergence. Because no partial derivatives, solutions of simultaneous equations, or matrix operations are required, the DAOP algorithm is simple to apply and program. DAOP is especially suited for programming with pointers. Advantages of the new algorithm include its simplicity, ease of incorporating inequality constraints, and the ability to predict the number of steps required to reach a solution.

In addition to optimal power flow, the algorithm, heuristic in nature, can be applied to switch placement design, reconfiguration, and economic dispatch. The basic principles of the algorithm have been used to devise a phase balancing routine which has been implemented in the Distribution

Engineering Workstation (DEWorkstation) software package sponsored by the Electric Power Research Institute (EPRI).

The new algorithm presented in this dissertation works toward a solution by performing a series of calculations within a finite number of steps. At the start of the algorithm, the assumption is made that no power is flowing in the system. Each step adds a discrete unit of load to the system in such a fashion as to minimize loss. As progress toward the solution is made, more and more load is satisfied and the losses in the system continue to increase. The algorithm is terminated when all system load is satisfied. When the algorithm is finished, the sources which should supply each load have been identified along with the amount of power delivered by each source. Discussion will show that the method will converge to a solution that is within the discrete step size of the optimum.

The algorithm can be thought of as an ascent method because the cost (losses) continually increases as more and more load is satisfied. Hence, the name Discrete Ascent Optimal Programming (DAOP) has been given to the algorithm.

The new algorithm uses the topology of the power system such that the entire system is not considered at each step. Therefore, DAOP is not an exhaustive state enumeration scheme. Only those portions of the system containing loads most closely connected (via least loss paths) to the sources are first considered. As loads become supplied during the course of the solution, other loads are considered and supplied until the system is fully loaded.

Dedicated to my wife, Robin N. Dolloff

Acknowledgments

I would like to thank Dr. Broadwater and the Electric Power Research Institute for the opportunity to work on this project. I would also like to thank Dr. Herdman, Dr. Rahman, Dr. Mili, and Dr. Lui for serving on my advisory committee. My appreciation is extended to all of my fellow graduate students in the DEWorkstation Laboratory for their guidance. Special recognition goes to Jeff Thompson and Larry Trussell. Thanks go to Al Sargent and Arkansas Power and Light for providing distribution circuit models on which the switch placement and phase balancing routines were tested. Thanks are also extended to Dr. Broadwater and his family and Kent and Myra Adams for their hospitality. Deepest thanks are extended to my wife, Robin, my parents, Howard and Anne, my brother, Matthew, and my mother-in-law, Louise Schoenemann for their support and patience. Let me not forget to thank my late night companion, Cassie Dawg Dolloff, our dog.

Contents

Figures

Tables

Appendix D Transmission Line Losses

Chapter One

Introduction

1.1 Introduction

This dissertation outlines the research efforts and software implementation of a newly developed optimal power flow algorithm for electrical distribution power systems. In addition to optimal power flow, the algorithm, heuristic in nature, can be applied to switch placement design, reconfiguration, and economic dispatch. The basic principles of the algorithm have been used to devise a phase balancing routine which has been implemented in the Distribution Engineering Workstation (DEWorkstation) software package sponsored by the Electric Power Research Institute (EPRI).

The objective of the optimal power flow problem is to improve power system efficiency by satisfying a set of loads at minimum generation costs and transmission losses while satisfying all constraints. The optimum system state is achieved when all loads are supplied with power at minimum line loss and generation costs. A new algorithm, Discrete Ascent Optimal Programming (DAOP), addresses this issue by determining, for each load, which substation bus or buses should supply power to that load.

1.2 The DAOP Algorithm

The new algorithm presented in this dissertation works toward a solution by performing a series of calculations within a finite number of steps. At the start of the algorithm, the assumption is made that no power is flowing in the system. Each step adds a discrete unit of load to the system in such a fashion as to minimize loss. With each step, the algorithm seeks to answer the question:

Which combination of source and load will result in the next increment of load being supplied at minimum line loss?

As progress toward the solution is made, more and more load is satisfied and the losses in the system continue to increase. The algorithm is terminated when all system load is satisfied. When the algorithm is finished, the sources which should supply each load have been identified along with the amount of power delivered by each source.

The algorithm can be thought of as an ascent method because the cost (losses) continually increases as more and more load is satisfied. Hence, the name Discrete Ascent Optimal Programming (DAOP) has been given to the algorithm.

Loads are modeled as either constant current or constant power. Thus, the discrete steps are either current or power. For constant power loads, constant power increments are used, and for constant current loads, constant current increments are used. For constant power loads, power flow solutions need to be run for each load increment to be evaluated. If the loads are treated as constant current, the calculations are greatly simplified, and are even amenable to spreadsheet solutions.

The new algorithm uses the topology of the power system such that the entire system is not considered at each step. Only those portions of the system containing loads most closely connected (via least loss paths) to the sources are first considered. As loads become supplied during the course of the solution, other loads are considered and supplied until the system is fully loaded. Therefore, the new algorithm is not an exhaustive state enumeration scheme.

Unlike iterative methods such as Gauss-Seidel and Newton-Raphson, DAOP will converge on the optimum without the threat of divergence, with a known degree of accuracy in the solution, and in a finite and determinable number of computational steps. Given the discrete step size and the total system load, the number of steps to reach the solution can be approximated by dividing the total load by the step size. Discussion will show that the method will converge to a solution that is within the discrete step size of the optimum.

DAOP has some similarities to dynamic programming. A fundamental principle of dynamic programming is optimal *sub-paths*. A sub-path is optimal if and only if it lies along the optimal path which extends from the initial system state to the final system state. The DAOP algorithm progresses through a series of *near* optimum points until the final system state is reached. That is, at each step line losses are minimized for the loads considered for that step. The qualifier near is used because the path of convergence lies close to the optimum (within the step size) for each step.

1.3 Dissertation Outline

The remaining chapters of this thesis are organized as follows. Chapter Two provides the results of a literature search on the various topics addressed in this dissertation. Chapter Three describes the proposed algorithm and gives a graphical interpretation of the path to convergence. This chapter outlines the computational mechanics, two rules which are observed, and the sequence of steps for the algorithm. An intuitive discussion followed by a theorem of convergence is presented in Chapter Four. Chapter Five describes how to adapt DAOP to perform switch placement design. The phase balancing variation of the algorithm is defined in Chapter Six along with the DEWorkstation implementation. Chapter Seven describes how to tailor DAOP to the economic dispatch problem in transmission systems. Finally, Chapter Eight gives conclusions concerning the algorithm's performance, lists the contributions of this research effort, and outlines possible future work.

Chapter Two

Literature Review

2.1 Introduction

This chapter contains the results of a literature search conducted in the area of optimization in electrical power systems. The application of the various algorithm types have been applied to optimal power flow, reconfiguration, restoration, and economic dispatch.

Brief descriptions and comparisons between past research efforts and the new algorithm described in this dissertation is presented. The new algorithm has been named Discrete Ascent Optimal Programming (DAOP) and will be described fully in the coming chapters.

2.2 Optimal Power Flow

The majority of feeders at the distribution voltage level are operated radially. Radial feeders, and their loads, receive power from only one point (direction) at any moment in time. However, it is possible to transfer load from one feeder to another by opening and closing switches while keeping the system radial. In so doing, system losses are reduced and load balancing among the feeders can be achieved.

In most cases, Newton based power flow programs developed for transmission networks can solve radially operated systems. However, their computational burden with relatively long solution times can be overcome by radial specific power flow algorithms. Because the radial power flow algorithms can solve many different configurations in a very short period of time, they can be used in automated distribution power operations.

One of the first published papers addressing the optimal power flow (OPF) problem is [2.1]. In this paper, the authors apply a gradient method to optimally adjust the control of parameters of a Newton based power flow algorithm.

Another popular OPF technique is the use of quadratic programming. These efforts are iterative in nature and minimize a quadratic approximation of the classical Lagrangian equation [2.2] [2.3]. Reference [2.4] combines the Newton approach with a quadratic approximation. The DAOP algorithm is similar to the work in [2.2] in that the OPF problem is divided into the real and reactive sub-problems.

Further developments in the OPF problem incorporate linear programming [2.5]. Very much like linear programming, the DAOP algorithm reaches a solution in a piecewise-linear segmentation process when solving the Lagrange objective (cost) function.

A different approach to the OPF problem [2.6] simplifies the constraint equations by eliminating the voltage angle at each bus and uses a variation of the ladder theory power flow solution [2.7].

The reference papers [2.8] and [2.9] examine deficiencies in existing OPF techniques. From [2.8] three concerns are:

1). The equivalents now used for OPF problems can cause large errors in OPF solutions, 2). The methods for handling discrete variables in OPF algorithms produce solutions that have significantly higher costs than optimal solutions,

3). The number of control actions used in solving OPF problems is often too large to be executed on an actual system.

The DAOP algorithm address each of these points as follows. Using DAOP, the Lagrange objective function is not approximated. During the DAOP solution process, a series of sub-objective functions are processed, none of which are approximations. Although DAOP uses a discrete approach, the final solution has been proven to be within the discrete step size. Therefore, the smaller the step size, the greater the accuracy in the final solution. DAOP provides elegant solutions when encountering linear and non-linear constraints. Finally, as addressed in [2.9], the final DAOP solution is not dependent on a feasible starting solution. In fact, the starting point of the algorithm is the same for all systems: completely unloaded.

The intent of the early OPF research was to reduce system losses. Later efforts apply basic OPF theory to energy management systems, namely reactive power flow control. The reactive OPF problem is strongly linked to voltage control because the power factor, which is a function of the voltage angle, determines the amount of reactive power flow. Reference [2.10] applies the Newton method to the reactive OPF while [2.11] and [2.12] use linear programming. Reference [2.13] is a real time implementation of the reactive OPF problem.

2.3 Reconfiguration

A popular topic of optimization in electrical distribution systems is reconfiguration. Reconfiguration is the process of altering the topological structures of distribution feeders by changing the open/closed status of the sectionalizing and tie switches. During normal operating conditions, networks are configured to reduce system losses and to balance loading.

During equipment failure and fault conditions, transferring load, or reconfiguring of the system, is often performed to isolate the problem area while retaining service to as many customers as possible. This type of action is known as restoration.

Before any system reconfiguration is implemented, a power flow study should be performed on each configuration to determine if the system can be operated safely. All feeders of the reconfigured system must operate within their thermal, current, and voltage limits, and the protection scheme should protect against faults.

This section provides a survey of work dealing with switch operation schemes in the area of reconfiguration because publications on true switch placement design and phase balancing algorithms are extremely rare. The switching schemes researched, however innovative, do not provide recommendations for new switch locations. Present load balancing techniques only attempt to move load from one circuit to another. The DAOP extension to phase balancing recommends to which phases single and two phase laterals should connect for improvement in substation imbalance and reduction of losses.

A number of original switching schemes have been devised to optimally reconfigure distribution systems. Heuristic in nature, these methods achieve loss reduction by performing switch exchange operations. The work in [2.14] and [2.15] involves the operation of switch pairs. Methods are devised to determine, for each pair, which switch to open and which to close for a reduction in system losses. The solution schemes in [2.16] and [2.17] start with all system switches closed. A radial system is produced by selecting which switches to open by using an equivalent linear resistive network model. Work in [2.18] is an extension of that in [2.14] by using integer programming techniques. This work also includes a method for capacitor placement. Work in [2.19] is also an extension of the work in [2.14] but converges to a solution independently of the starting solution. These research efforts fall into the greedy search techniques class which only accept search movements that produce immediate improvements.

The two part papers [2.20] and [2.21] use a spanning tree type approach: Given a graph (i.e. nodes of the system), find a spanning tree (radial configuration) such that a desired objective function is minimized while certain system constraints are satisfied. The work in [2.22] sets up a decision tree to represent the various switching operations available. A best first tree searching strategy, based on heuristics, is used to evaluate the various alternatives.

Other approaches such as [2.23] convert the optimization problem of determining the open/closed status of the system switches into an iterative series of continuous quadratic programming sub-problems. The work in [2.24] converts the reconfiguration problem into a quadratic simplex problem while [2.25] and [2.26] combine a transshipment formulation with quadratic costs. These methods create the optimal solution by improving an initial (or current) basic solution.

The work in [2.27] involves the use of an artificial neural network. The researchers adopt the multi-layer feedforward machine developed in [2.28]. The multi-layer perception has the ability of not only handling the analog/binary input for feeder reconfiguration, but also mapping the complex and nonlinear input-output relationship with the hidden layer. The model is trained by the error back propagation algorithm, and then the adjustment process of the interconnecting weights and thresholds is repeated until the appropriate recognition capability is obtained.

2.4 Economic Dispatch

At this point in the literature review, the DAOP optimizing technique was determined to be unique and no references can be found relating DAOP to economic dispatch. Reference [2.30] was used to study, in depth, the Lagrange multiplier approach for solving the economic dispatch problem. In this book, the authors provide a thorough examination of not only the economic dispatch problem but the Lagrange technique as well. Reference [2.31] provides an excellent review of recent advances in the area of economic dispatch.

2.5 Conclusions

After conducting the literature review, it was found that the Discrete Ascent Optimal Programming method for optimization presented in this dissertation is original. All concepts behind DAOP's ascent philosophy are also original. Not only does DAOP provide an optimal power flow technique, but the fundamental properties of DAOP can be applied to switch placement design, phase balancing, and economic dispatch.

2.6 References

[2.1] H. Dommel and W. Tinney, *Optimal Power Flow Solutions*, IEEE Transactions on Power Apparatus and Systems, Vol. PAS-87, No. 10, October 1968, pp. 1866 - 1876.

[2.2] G. Contaxis, C. Delkis, and G. Korres, *Decoupled Optimal Load Flow Using Linear or Quadratic Programming*, IEEE Transactions on Power Systems, Vol. PWRS-1, No. 2, May 1986.

[2.3] C. Lu, S. Chen, and C. Ong, *The Incorporation of HVDC Equations in Optimal Power Flow Methods Using Sequential Quadratic Programming Techniques*, IEEE Transactions on Power Systems, Vol. 3, No. 3, August 1988, pp. 1005 - 1011.

[2.4] G. Maria and J. Findlay, *A Newton Optimal Power Flow Program for Ontario Hydro EMS*, IEEE Transactions on Power Systems, Vol. PWRS-2, No. 3, August 1987, pp. 576 - 581.

[2.5] O. Alsac, J. Bright, M. Prais, and B. Stott, *Further Developments in LP-Based Optimal Power Flow*, IEEE Transactions on Power Systems, Vol. 5, No. 3, August 1990, pp. 697 - 706.

[2.6] C. Renato, *New Method for the Analysis of Distribution Networks*, IEEE Transactions on Power Delivery, Vol. 5, No. 1, January 1990, pp. 391 - 396.

[2.7] W. Kersting and D. Mendive, *An Application of Ladder Theory to the Solution of Three-Phase Radial Load-Flow Problems*, IEEE PAS Winter Meeting, New York, NY, January 1976, Paper No. A 76 044-8.

[2.8] W. Tinney, J. Bright, K. Demaree, and B. Hughes, *Some Deficiencies in Optimal Power Flow*, IEEE Transactions on Power Systems, Vol. 3, No. 2, May 1988, pp. 676 - 681.

[2.9] A. Papalexopoulos, C. Imparato, and F. Wu, *Large-Scale Optimal Power Flow: Effects of Initialization, Decoupling & Discretization*, IEEE Transactions on Power Systems, Vol. 4, No. 2, May 1989, pp. 748 - 754.

[2.10] M. Bjelogrlic, M. Calovic, B. Babic, and P. Ristanovic, *Application of Newton's Optimal Power Flow in Voltage/Reactive Power Control*, IEEE Transactions on Power Systems, Vol. 5, No. 4, November 1990, pp. 1447 - 1453.

[2.11] D. Kirschen and H. Van Meeteren, *MW/Voltage Control in a Linear Programming Based Optimal Power Flow*, IEEE Transactions on Power Systems, Vol. 3, No. 2, May 1988, pp. 481 - 486.

[2.12] G. Opoku, *Optimal Power System Var Planning*, IEEE Transactions on Power Systems, Vol. 5, No. 1, February 1990, pp. 53 - 59.

[2.13] C. Lin, T. Chen, and C. Huang, *A Real-Time Calculation Method for Optimal Reactive Power Compensator*, IEEE Transactions on Power Systems, Vol. 4, No. 2, May 1989, pp. 643 - 652.

[2.14] S. Civanlar, J. Grainger, H. Yin, and S. Lee, *Distribution Feeder Reconfiguration for Loss Reduction*, IEEE Transactions on Power Delivery, Vol. 3, No. 3, July 1988, pp. 1217 - 1223.

[2.15] D. Ross, J. Patton, A. Cohen, and M. Carson, *New Methods for Evaluating Distribution Automation and Control System Benefits*, IEEE Transactions on PAS, Vol. PAS-100, No. 6, June 1981, pp. 2978 - 2986.

[2.16] A. Merlin and H. Back, *Search for a Minimal-Loss Operating Spanning Tree Configuration for an Urban Power Distribution System*, Proceedings of PSCC, Cambridge 1975, paper 1.2/6.

[2.17] S. Goswami and S. Basu, *A New Algorithm for the Reconfiguration of Distribution Feeders for Loss Minimization*, IEEE Transactions on Power Delivery, Vol. 7, No. 3, July 1992, pp. 1484 - 1491.

[2.18] M. Baran and F. Wu, *Network Reconfiguration in Distribution Systems for Loss Reduction and Load Balancing*, IEEE Transactions on Power Delivery, Vol. 4, No. 2, April 1989, pp. 1401 - 1407.

[2.19] D. Shirmohammadi and H. Hong, *Reconfiguration of Electric Distribution Networks for Resistive Line Losses Reduction*, IEEE Transactions on Power Delivery, Vol. 4, No. 2, April 1989, pp. 1492 - 1498.

[2.20] H. Chiang and R. Jean-Jumeau, *Optimal Network Reconfigurations in Distribution Systems: Part 1: A New Formulation and a Solution Methodology*, IEEE Transactions on Power Delivery, Vol. 5, No. 4, November 1990, pp. 1902 - 1909.

[2.21] H. Chiang and R. Jean-Jumeau, *Optimal Network Reconfigurations in Distribution Systems: Part 2: Solution Algorithms and Numerical Results*, IEEE Transactions on Power Delivery, Vol. 5, No. 3, July 1990, pp. 1568 - 1574.

[2.22] T. Taylor and D. Lubkeman, *Implementation of Heuristic Search Strategies for Distribution Feeder Reconfiguration*, IEEE Transactions on Power Delivery, Vol. 5, No. 1, January 1990, pp. 239 - 246.

[2.23] K. Aoki, T. Ichimori, and M. Kanezashi, *Normal State Optimal Load Allocation in Distribution Systems*, IEEE Transactions on Power Delivery, Vol. PWRD-2, No. 1, January 1987, pp. 147 - 155.

[2.24] W. Zangwill, *The Convex Simplex Method*, Management Science, 1967, pp. 221 - 238.

[2.25] A. Orden, *The Transshipment Problem*, Management Science, 1956, pp. 276 - 285.

[2.26] V. Glamocanin, *Optimal Loss Reduction of Distribution Networks*, IEEE Transactions on Power Systems, Vol. 5, No. 3, August 1990, pp. 774 - 782.

[2.27] H. Kim, Y. Ko, and K. Jung, *Artificial Neural-Network Based Feeder Reconfiguration for Loss Reduction in Distribution Systems*, IEEE Transactions on Power Delivery, Vol. 8, No. 3, July 1993, pp. 1356 - 1366.

[2.29] D. Rumelhart, G. Hinton, and R. Williams, *Learning Internal Representations by Error Propagation*, in D. Rumelhart and J. McClelland (eds.) Parallel Distributed Processing: Exploration in the Microstructure of Cognition, Vol. 1, Foundations, MIT Press, 1987.

[2.30] Allen J. Wood and Bruce F. Wollenberg, Power Generation, Operation, and Control, John Wiley and Sons, New York, NY, 1984.

[2.31] B. Chowdhury and S. Rahman, A Review of Recent Advances in Economic Dispatch, IEEE Transactions on Power Systems, Vol. 5, No. 4, November 1990, pp. 1248 - 1254.

Chapter Three

DAOP Description

3.1 Introduction

This chapter gives a graphical interpretation of the Discrete Ascent Optimal Programming (DAOP) algorithm path to convergence which shows how the power system topology is exploited. Definitions of graphical terms are given. The computational mechanics of the algorithm, two rules which are followed, and the solution steps are also given.

The graphical interpretation, which offers a pictorial view of the algorithm's convergence process, is best understood when demonstrated by example. For simplicity, the example uses a DC system. The extension to AC systems is straight forward. In either case, the power system loads will be supplied in discrete increments.

3.2 Definition of Terms

In the description of the solution process, line sections and loads which are not carrying power are drawn with dots. Once a line section carries any power, it is drawn with a solid line. Partially supplied loads are called ending loads and are drawn with dashed arrows. A completely supplied load is drawn with a solid arrow [3.1].

Figure 3.1 illustrates a DC system with three current sources, seven line sections modeled as resistors, and five constant current loads. When DAOP starts, all loads are assumed to be unsupplied and all line sections and loads are drawn with dots as shown in Figure 3.2. During the course of the algorithm, as the loads become supplied, portions of the system are drawn with solid and dashed lines as well as dotted lines as shown in Figure 3.3. In Figure 3.3, two loads are fully supplied, two ending loads are partially supplied, and one load is completely unsupplied.

Portions of the system connected with a solid line are referred to as supplied graphs while the dotted and dashed portion of the system is termed the unsupplied graph. At the start of DAOP, all of the loads are included in the unsupplied graph as shown in Figure 3.2. As the algorithm progresses toward the solution, separate supplied graphs may exist as shown in Figure 3.3. In addition, the supplied graphs merge into one another as ending loads become fully supplied. At convergence, the entire system is included in a single supplied graph as show in Figure 3.1.

3.3 Exploiting System Topology

Figures 3.1 - 3.3 illustrate how DAOP processes the power system topology. At the start, each source is a member of a supplied graph where each supplied graph contains at least one ending load. The first set of ending loads are defined as those loads which are most closely connected via the least loss path to a supplied graph. Other unsupplied loads are not considered until they become an ending load of a supplied graph. This technique allows the algorithm to exploit the system topology by breaking the system into smaller sub-systems. At each step, DAOP considers all subsystems to find the optimal, feasible solution for the next increment of load.

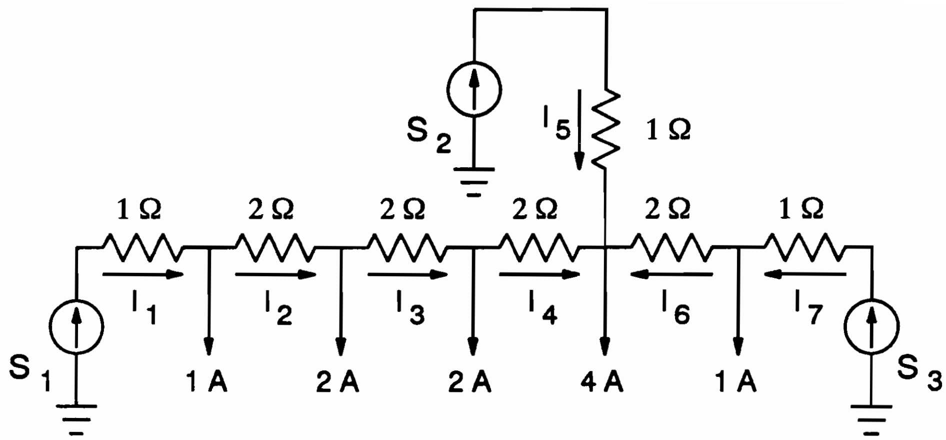

Figure 3.1 Example 3.1: DC System

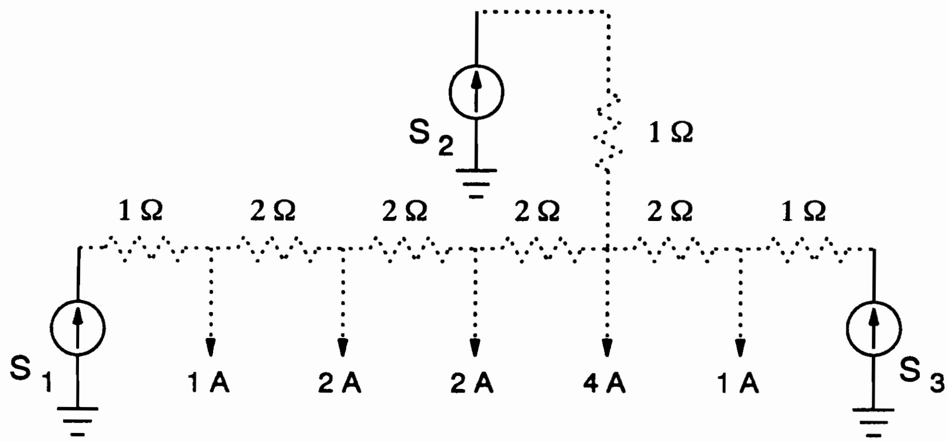

Figure 3.2 Example 3.1 at Start of DAOP

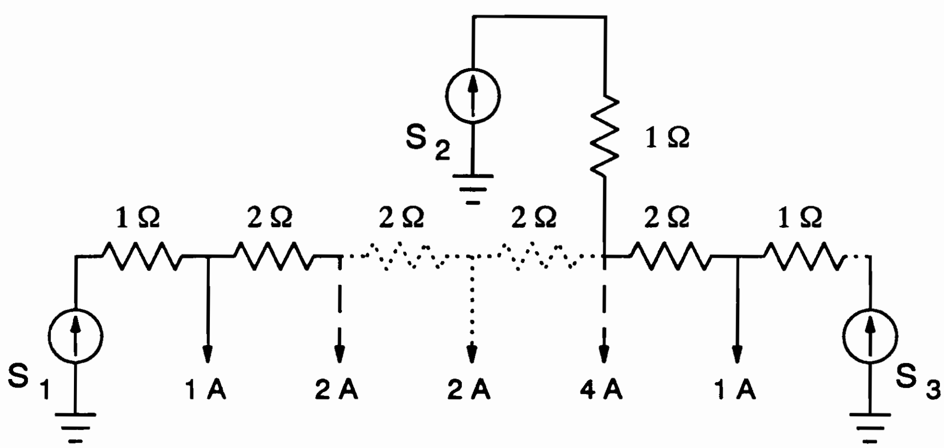

Figure 3.3 Example 3.1 During DAOP Solution Process

3.4 Computational Mechanics

With each step, the DAOP algorithm adds the load increment to the source which produces the minimum line losses. The line losses increase with each supplied load increment, where the calculation of line losses includes the line losses due to the new load increment, the previously supplied loads, and the ending loads. The DAOP algorithm can be thought of as an ascent method because the losses continually increase as more and more load is satisfied.

As the algorithm progresses, the two rules given below are observed:

> Rule 1: All loads that are supplied must be part of a supplied graph with one or more sources.
>
> Rule 2: A load must be supplied from sources that belong to the same supplied graph.

The load increment may be varied from step to step. Load increments are generally selected to be some percentage of the smallest existing ending load (from experience, 20% is reasonable). When the amount needed to fully supply an ending load is smaller than the existing load increment, the load increment may be reduced to the exact amount needed to fully supply that particular ending load. The load increment may be reset when a new ending load is defined. Once an ending load has been completely supplied, adjacent, unsupplied loads, if they exist, become new ending loads.

With each step, the line losses due to supplying the load increment from every source are evaluated. The source and load combination producing the minimum increase in line loss is selected. Only one source is incremented by the given load increment at the conclusion of each step. In the event that two or more sources produce the same minimum increase in line loss, an arbitrary choice of which source to increment is made. In the case of such a tie, the explanation given in Chapter 4 will confirm that the final solution is not affected by the arbitrary choice.

When AC systems are analyzed, the real and reactive components are considered separately. That is, either the real load increment or the reactive load increment that yields the smallest increase in line losses is selected. Note that the real and reactive load increments need not be equal.

The following summarizes the basic steps followed by the DAOP algorithm.

Step 1: Calculate the line loss associated with supplying the next increment of load at each ending load in the system. If the load contains both real and reactive components, the real load increment is considered separately from the reactive load increment.

Step 2: From the results of Step 1, choose the combination of source and ending load which produces the minimum increase in loss.

Step 3: If an ending load is completely supplied as a result of Step 2, the ending load becomes part of a supplied graph. If adjacent loads exist, they become new ending loads.

Step 4: If all loads are fully supplied, the DAOP algorithm has converged; otherwise, return to Step 1.

3.5 Example 3.1: DC System

The minimum line loss problem for the DC system shown in Figure 3.1 is solved using both the Lagrange multiplier approach and the DAOP algorithm. The example illustrates the steps of the DAOP algorithm and compares the results with the solution obtained from the Lagrange multiplier approach.

For this system, the optimal power flow problem can be mathematically expressed by the following loss function and equality constraints:

$$\text{Minimize}\{\, 1I_1^{\,2} + 2I_2^{\,2} + 2I_3^{\,2} + 2I_4^{\,2} + 1I_5^{\,2} + 2I_6^{\,2} + 1I_7^{\,2} \,\} \tag{3.1}$$

Subject to:

$$I_1 - I_2 - 1 = 0 \tag{3.2}$$

$$I_2 - I_3 - 2 = 0 \tag{3.3}$$

$$I_3 - I_4 - 2 = 0 \tag{3.4}$$

$$I_4 + I_5 + I_6 - 4 = 0 \tag{3.5}$$

$$I_6 - I_7 + 1 = 0 \tag{3.6}$$

Loss function (3.1) provides the total system line loss which is the quantity to be minimized. Equations (3.2) - (3.6) represent line flow and load current constraints. Using Lagrange multipliers, loss function (3.1) and equality constraints (3.2) - (3.6) may be combined as follows:

$$\begin{aligned} & F(\, I_1, I_2, I_3, I_4, I_5, I_6, I_7, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 \,) \\ = \quad & 1I_1^{\,2} + 2I_2^{\,2} + 2I_3^{\,2} + 2I_4^{\,2} + 1I_5^{\,2} + 2I_6^{\,2} + 1I_7^{\,2} \\ & + \lambda_1 \,(\, I_1 - I_2 - 1 \,) \\ & + \lambda_2 \,(\, I_2 - I_3 - 2 \,) \\ & + \lambda_3 \,(\, I_3 - I_4 - 2 \,) \\ & + \lambda_4 \,(\, I_4 + I_5 + I_6 - 4 \,) \\ & + \lambda_5 \,(\, I_6 - I_7 + 1 \,) \end{aligned} \tag{3.7}$$

To solve for the minimum, partial derivatives of (3.7) with respect to the currents and Lagrange multipliers are taken and each of the partials is set equal to zero. The resulting system of simultaneous equations may then be solved. Performing these operations will yield the solution in Table 3.1.

Table 3.1 Lagrange Multiplier and DAOP Solutions for Example 3.1

Current	Lagrange Multiplier	DAOP Algorithm
I_1	3.2	3.0
I_2	2.2	2.0
I_3	0.2	0.0
I_4	-1.8	-2.0
I_5	4.6	5.0
I_6	1.2	1.0
I_7	2.2	2.0
Total Loss	55.4	56.0

For the DAOP algorithm, a load increment, Lp, must be selected. Let the load increment be given by

$$Lp = 1 \text{ amp.} \tag{3.8}$$

From Figure 3.1, the total load to be supplied is 10 amps. At each step, 1 amp of load will be supplied, and thus, ten steps are required for convergence.

Table 3.2 organizes the DAOP algorithm solution. At each step the line losses are evaluated by adding the 1 amp load increment to each combination of source and ending load in turn. The S_1, S_2, and S_3 columns form a matrix which indicates the output from each source. This matrix shows how the load increment is added to each source (as indicated by the Source column) in turn. The *Δ Loss* column indicates the additional system loss while the *Total Loss* column gives the total system loss due to the loading condition given in the corresponding row of the source output matrix. The <= symbol under the *Selection* column indicates which source increment yields the minimum total loss for that particular step. The results from the DAOP algorithm are shown in Table 3.1.

This example shows that the solution currents obtained from the DAOP algorithm lie within the increment, Lp, of the solution obtained by the Lagrange multiplier method. For this example, the total line loss computed by the two methods differs by less than 1 watt as shown in Table 3.1.

Table 3.2 DAOP Solution Table for Example 3.1

Step	Source	S_1	S_2	S_3	Δ Loss	Total Loss	Selection
		(Source Output Matrix)					
	S_1	1	0	0	1	1	<=
1	S_2	0	1	0	1	1	
	S_3	0	0	1	1	1	
	S_1	2	0	0	5	6	
2	S_2	1	1	0	1	2	<=
	S_3	1	0	1	1	2	
	S_1	2	1	0	5	7	
3	S_2	1	2	0	3	5	
	S_3	1	1	1	1	3	<=
	S_1	2	1	1	5	8	
4	S_2	1	2	1	3	6	<=
	S_3	1	1	2	5	8	
	S_1	2	2	1	5	11	<=
5	S_2	1	3	1	5	11	
	S_3	1	2	2	5	11	
	S_1	3	2	1	11	22	
6	S_2	2	3	1	5	16	<=
	S_3	2	2	2	5	16	
	S_1	3	3	1	11	27	
7	S_2	2	4	1	7	23	
	S_3	2	3	2	5	21	<=
	S_1	3	3	2	11	32	
8	S_2	2	4	2	9	30	<=
	S_3	2	3	3	13	34	
	S_1	3	4	2	11	41	<=
9	S_2	2	5	2	15	45	
	S_3	2	4	3	17	47	
	S_1	4	4	2	19	60	
10	S_2	3	5	2	15	56	<=
	S_3	3	4	3	17	58	

3.6 References

[3.1] R. Broadwater, P. Dolloff, T. Herdman, R. Karamikhova, and A. Sargent, *Minimum Loss Optimization in Distribution Systems: Discrete Ascent Optimal Programming*, Electric Power Systems Research, Winter 1995.

Chapter Four

Convergence of DAOP

4.1 Introduction

This chapter gives an intuitive discussion of DAOP's convergence path and compares the method with the Lagrange multiplier approach using a DC power system example. Following the example, a theorem of convergence is given.

4.2 Discussion of Convergence

For Example 4.1, consider the two source, constant current, DC system in Figure 4.1. At each step, a loss function can be developed (see Appendix A) in terms of I_2. Figure 4.2 is the series of configurations the system undergoes during the course of the solution. These configurations may be used to develop a series of loss functions, each of which will be referred to as a sub-loss function.

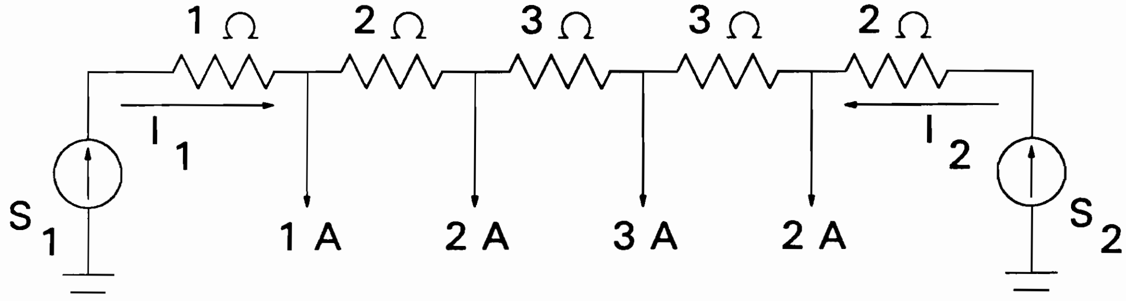

Figure 4.1 Example 4.1: DC System

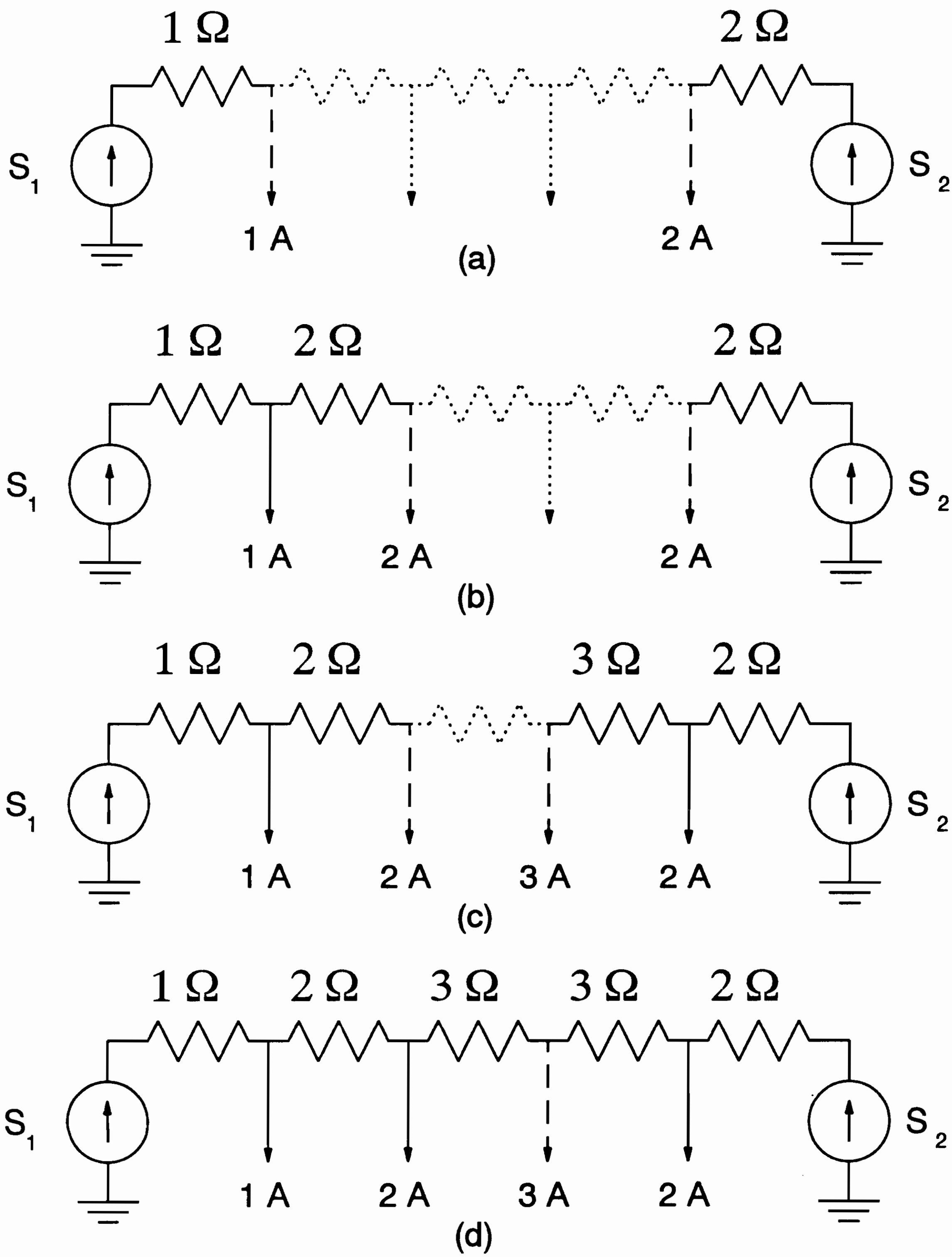

Figure 4.2 Series of Configurations of Example 4.1

The solution path that is followed by the algorithm passes close to the minimum of each of these sub-loss functions. Let F_1 represent the applicable sub-loss function for step 1, F_2 the sub-loss function for step 2 and so forth. The sub-loss functions corresponding to the eight steps for the example DC system are given by (4.1) - (4.8) shown below.

$$F_1(I_2) = 1 - 2I_2 + 3I_2^2 \tag{4.1}$$

$$F_2(I_2) = 6 - 8I_2 + 5I_2^2 \tag{4.2}$$

$$F_3(I_2) = 17 - 14I_2 + 5I_2^2 \tag{4.3}$$

$$F_4(I_2) = 34 - 20I_2 + 5I_2^2 \tag{4.4}$$

$$F_5(I_2) = 69 - 38I_2 + 8I_2^2 \tag{4.5}$$

$$F_6(I_2) = 98 - 44I_2 + 8I_2^2 \tag{4.6}$$

$$F_7(I_2) = 181 - 74I_2 + 11I_2^2 \tag{4.7}$$

$$F_8(I_2) = 249 - 72I_2 + 11I_2^2 \tag{4.8}$$

Note that F_8 is the loss function that would be used by the Lagrange multiplier method. *Hence, the sequence of sub-loss functions converges to the loss function that would be used by the Lagrange multiplier method.* Table 4.1 indicates the calculation step, the sub-loss function that applies for that step, and the sub-figure of Figure 4.2 from which the sub-loss function may be derived.

Table 4.1 Step, Loss Function, and Corresponding Figure Number for Example 4.1

Step	Fi	Figure	Step	Fi	Figure
1	F_1	4.2-A	5	F_5	4.2-C
2	F_2	4.2-B	6	F_6	4.2-D
3	F_3	4.2-B	7	F_7	4.2-D
4	F_4	4.2-B	8	F_8	4.2-D

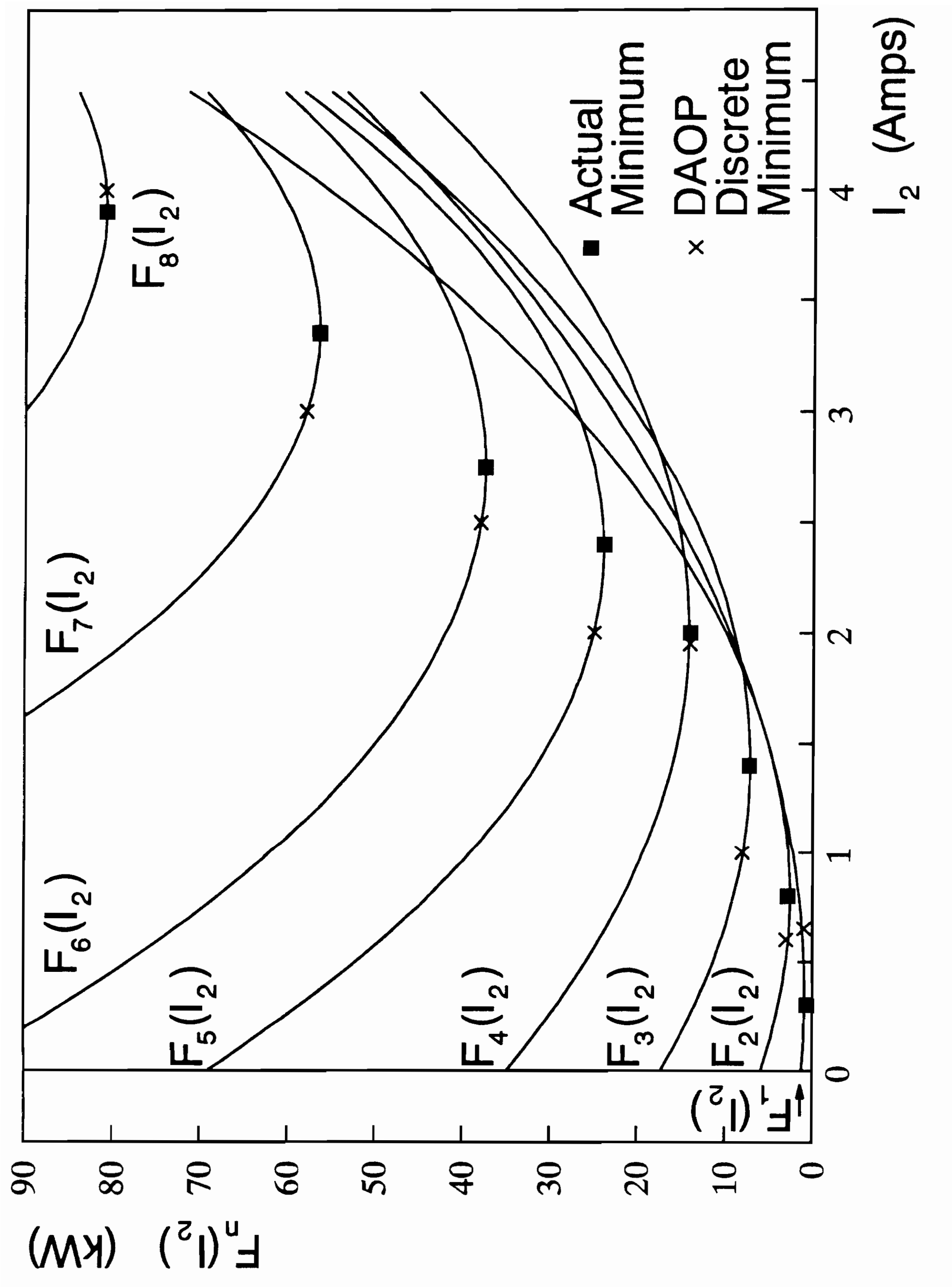

Figure 4.3 Sequence of Sub-Loss Functions Followed by DAOP

Figure 4.3 shows plots of all eight sub-loss functions on a total loss versus I_2 plane. The cross marks indicate the solution path followed by the algorithm while the boxes indicate the actual minimum point for each sub-loss function. As shown, the solution path followed by the algorithm passes close (within the step size of 1) to the minimum points of the sub-loss functions. Note that using a smaller step increases both the accuracy of the solution and the number of computational steps required to reach the solution.

4.3 Convergence Theorem [4.1]

A theorem of convergence for a loss function of two variables subject to no constraints follows. This function can be derived from a three variable (source) system in which the constraints have been used to eliminate one variable from the loss function. In particular, it is illustrated that the scheme will, in a finite number of steps, produce an ordered pair of integers (x, y) that gives an integer approximation to the true point of minimization (x*, y*) such that

$$[x^*] \le x \le [x^*] + Lp \tag{4.9}$$

$$[y^*] \le y \le [y^*] + Lp, \tag{4.10}$$

where the step size $Lp = 1$.

Here $[x^*]$ denotes the greatest integer less than or equal to the real number x^*.

Consider the problem of finding the point (x*, y*) that minimizes the quadratic function

$$f(x, y) = ax^2 + by^2 + cxy - dx - ey. \tag{4.11}$$

Through substitutions, Loss Function (3.1) and equality constraints (3.2) - (3.6) may be placed in the form of (4.11). Since (4.11) is an energy function representing the losses in the system, it is positive definite [4.2].

The problem of finding (x^*, y^*) is equivalent to minimizing the function $F(x, y)$ given by

$$F(x, y) = \tfrac{1}{2}(x, y)\, A(x, y)^T - B(x, y)^T \tag{4.12}$$

where:

$$B = [d,e] \tag{4.13}$$

and A is the positive definite matrix

$$A = \begin{bmatrix} 2a & c \\ c & 2b \end{bmatrix} \tag{4.14}$$

such that $a > 0$, $b > 0$, $c \geq 0$, $d \geq 0$, $e \geq 0$, $2a > c$, $2b > c$, $d \geq c$, and $e > c$.

Because A is positive definite, further examination of (4.12) reveals the loss function to be an ellipse. In addition, $F(x, y)$ has a unique global minimum (x^*, y^*) [4.3] such that

$$\nabla F(x^*, y^*) = (F_x(x^*, y^*), F_y(x^*, y^*)) = (0,0), \tag{4.15}$$

where $F_x(x^*, y^*)$ represents the partial of $F(x, y)$ with respect to x and $F_y(x^*, y^*)$ represents the partial of $F(x, y)$ with respect to y evaluated at the optimal solution.

Let the line L in the xy-plane be defined by

$$L = \{ (x, y):\ F_x(x, y) = F_y(x, y) \}. \tag{4.16}$$

The line L defines the path of equal incremental loss. It follows that $(x^*, y^*) \in L$.

A single contour plot of the function f(x, y) along with the line L is illustrated in Figure 4.4. The shape, rotation, and translation of the ellipse depend on the values of the coefficients a, b, c, d, and e in (4.11).

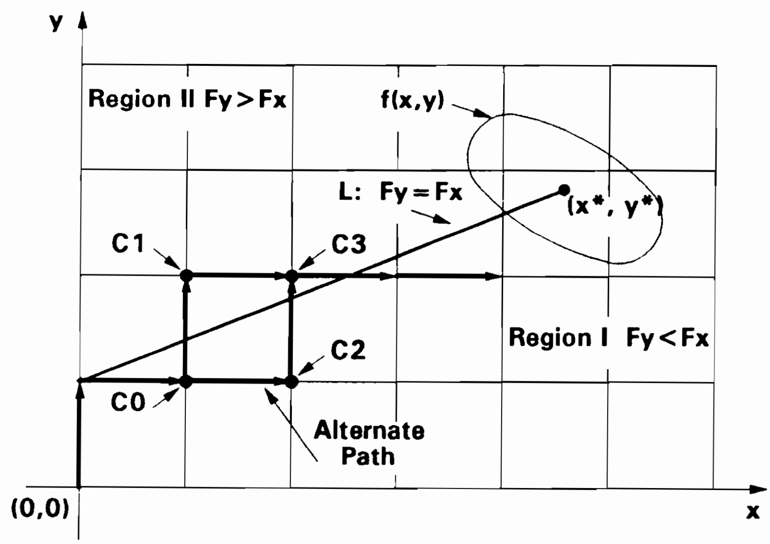

Figure 4.4 Illustration of Solution Path Followed by DAOP

As illustrated in Figure 4.4, the line L divides the xy-plane into two half planes. The lower half plane contains all points (x, y) satisfying $y<mx+b_y$ where m is the slope and b_y is the y-intercept of the line L. It follows that $y<mx+b_y$ if and only if $F_y(x, y) < F_x(x, y)$. On the other hand, $y>mx+b_y$ if and only if $F_y(x, y) > F_x(x, y)$. Consequently, the following two regions are defined in the xy-plane.

$$\text{Region I} = \{ (x, y):\ F_y(x, y) < F_x(x, y) \} \tag{4.17}$$

$$\text{Region II} = \{ (x, y):\ F_y(x, y) > F_x(x, y) \} \tag{4.18}$$

Figure 4.4 also pictures the steps taken by the DAOP algorithm indicated by the series of arrows. This path, stair-step in nature because the load increment can only be added in the positive x or y (I_1 or I_2) direction, closely follows the line of equal incremental loss.

The starting point for the algorithm is (0,0). If (0,0) is in Region I, $F_y(x, y) < F_x(x, y)$, the algorithm takes k steps in the direction (0,1) such that the k+1 step would place (x, y_{k+1}) in Region II. In order to cross into Region II, $F(x, y_{k+1})$ must be less than $F(x_{k+1}, y_k)$. If not, the algorithm remains in Region I and takes n steps in the (1,0) direction until $F(x_{n+1}, y_k) > F(x_n, y_{k+1})$. At this point, the algorithm takes a single step in the (0,1) direction causing the solution path to cross into Region II. Now in region II, $F_y(x, y) > F_x(x, y)$, the algorithm steps in the direction (1,0). As before, the solution path does not cross into Region I until the function is not minimized by staying in Region II. The procedure is repeated until the solution is reached. The solution is within the load increment of the actual optimum given by (x^*, y^*).

The only exception to the above described movement is when equal minimum loss can be obtained from multiple directions. Consider the points C_0, C_1, C_2, and C_3 shown in Figure 4.4. Let $C_0 = (x_j, y_j)$ be a non stopping point on the DAOP path to convergence. Also, let the change in losses from C_0 to C_1 equal the change in losses from C_0 to C_2. Because these changes in loss are equal, the DAOP solution path can move to either C_1 or C_2. Each path continues to C_3 in two steps from the point of split at C_0. The two paths, via either C_1 or C_2, result in the same increase in total system loss. Therefore, when multiple directions yield the same minimum increase in loss, an arbitrary choice of direction can be made.

A theorem concerning the convergence for a function of two variables with no constraints is now presented.

> THEOREM:
> In k steps, where $[x^*] + [y^*] < k < [x^*] + [y^*] + 2$,
> the algorithm will stop and $(x, y) = (x_k, y_k)$ with
> $|x_k - x^*| < 1$, $|y_k - y^*| < 1$.

This result requires that the loss function have a unique minimum (x^*, y^*), a characteristic found in functions arising in unconstrained power optimization problems as given by (4.12). When applied to the DAOP algorithm, the theorem states that the optimum solution will be found in a finite number of steps [4.1].

4.4 References

[4.1] T. R. Burrell, An Alternating Direction Search Algorithm for Low Dimensional Optimization: An Application to Power Flow, Masters Thesis, VPI&SU, Blacksburg, VA. 1993.

[4.2] J. N. Franklin, Matrix Theory, Prentice Hall Inc., Englewood Cliffs, NJ, 1968.

[4.3] Gene Golub, Charles VanLoan, Matrix Computations, Johns Hopkins University Press, Baltimore & London, 1989.

Chapter Five

Switch Placement Design

5.1 Introduction

To make an electrical distribution system operate radially, the power engineer must determine which switches to open and which to close. Switch installations and switching operations are used as control actions to force the actual system to operate in a radial fashion. The optimal radial configuration may not be possible due to the lack of appropriately or poorly placed switching devices.

A switch placement design algorithm would not only indicate the open/close status of all switches in the system but would also indicate where to install new switches to realize the optimal radial solution.

This chapter describes the switch placement application of DAOP for best efficiency of operation of a radial distribution system. For a radially operated system, the DAOP algorithm must not allow individual loads to be supplied from multiple sources, thus placing a constraint on the solution. DAOP's ability to handle this constraint is explained. Finally, a switch placement design for a system on the Arkansas Power and Light distribution system is performed.

5.2 Radial Power Flow Constraints

In order to handle the radial constraint, the DAOP algorithm is run repeatedly. The switch placement design problem starts by allowing the

DAOP algorithm to run in the manner described thus far (normal fashion). When the unconstrained optimum is reached, the loads which are fed from multiple sources (multi-supplied loads) are identified. This unconstrained optimum solution is then used as a starting point for the constrained solution where the multi-supplied loads form the only set of unsupplied loads. All other loads from the unconstrained solution are treated as supplied loads. Thus, at the start of the constrained solution the multi-supplied loads form the set of ending loads to be considered.

Upon completion of the first step of the constrained solution, the source and ending load combination which produces the minimum loss from this step is identified. From here, the decision is made to allow the identified ending load to be entirely supplied by the associated source. For the remainder of the solution process, all paths connected to this load bus can only carry power away from the bus. This constrained second solution has reduced the number of multi-supplied loads by at least one.

This method, heuristic in nature, is repeated until all loads are supplied from a single source. Therefore, the number of times DAOP must be repeated is less than or equal to the number of multi-supplied loads that exist in the unconstrained solution. When the final solution is reached, the open switch locations are identified as those branches which do not carry any power.

5.3 Example 5.1: Switch Placement Design

Consider the system of Figure 5.1 provided by Arkansas Power (AP&L) and Light. For simplicity, the system has been approximated using a DC representation with constant current loads.

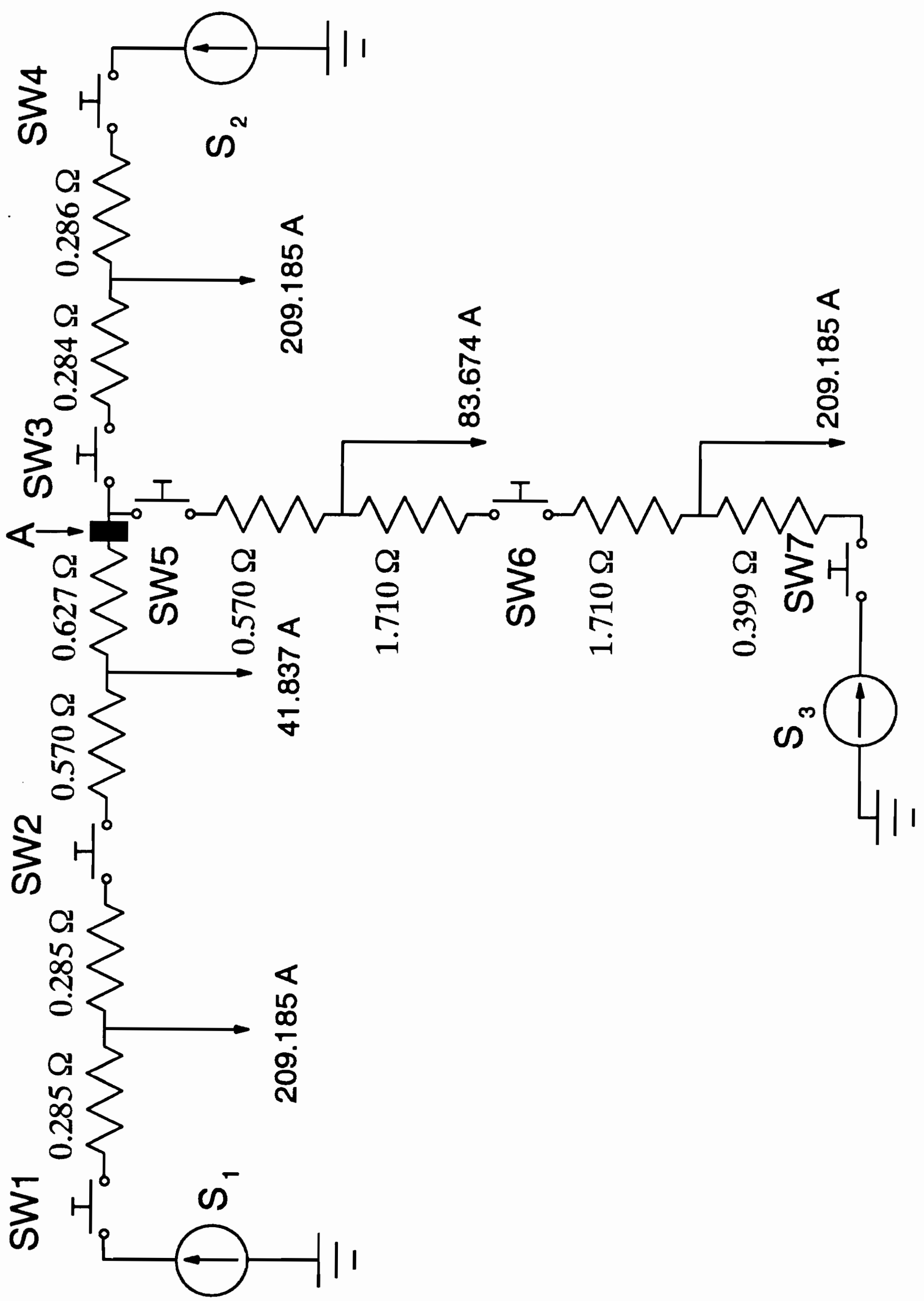

Figure 5.1 AP&L Switch Placement Design System

This particular circuit contains a number of switches which are currently used to produce a radial system. Comparisons between different radial systems achievable using these switches and the proposed DAOP switch locations will be made. Note that the DAOP solution is not an exhaustive search among the possible radial configurations using the existing switches.

To improve performance, a variable load increment will be employed in this example. When the unsupplied amount of an ending load is less than the load increment, Lp will be adjusted (down) to that amount which will exactly supply the ending load in question. When the ending load becomes fully supplied, Lp can be reset to 20% of the smallest existing ending load.

Table 5.1 Switch Placement Design Configurations for Example 5.1

Switch	Configuration 1	Configuration 2	Configuration 3
SW1	Closed	Closed	Closed
SW2	Closed	Closed	Closed
SW3	Open	Open	Closed
SW4	Closed	Closed	Closed
SW5	Open	Closed	Closed
SW6	Closed	Open	Open
SW7	Closed	Closed	Closed
A	N/A	N/A	Open
Total Loss	97,723.4 W	74,418.9 W	60,010.6 W

Switches SW1 - SW7 exist in the system. With these switches, different independent, radial system configurations can be made. Table 5.1 presents three different configurations. Configurations 1 and 2 can be achieved using the existing switches while Configuration 3 is the design result achieved using DAOP. From these results, the minimum line loss is achieved if a new switch is installed at position A in Figure 5.1. Table 5.1 gives the total line loss for each of the three configurations.

5.4 Conclusions

This chapter describes the switch placement design algorithm for electrical distribution systems. The algorithm is an extension of the DAOP algorithm. The routine has been tested on a real circuit in the Arkansas Power and Light distribution system in Hot Springs, Arkansas. Results of these tests have been independently verified by AP&L personnel.

Chapter Six

Phase Balancing Application

6.1 Introduction

To reduce losses in an electrical distribution system, the power engineer will distribute the system load evenly across the three phases. Decisions must be made as to which phases single and two phase laterals are to connect. A phase balancing routine would indicate the phasing of all laterals and compute the system losses and imbalance at the substation. A true design application would allow the engineer to chose the lateral reconnections and present the results.

This chapter describes how the basic principles of the DAOP algorithm have been used to develop a novel approach to phase balancing electrical distribution systems. The new phase balancing routine has been incorporated as a design application in the Distribution Engineering Workstation (DEWorkstation) software package sponsored by the Electric Power Research Institute (EPRI) [6.1]. Two phase balancing designs are given for circuits taken from the Arkansas Power and Light (AP&L) distribution system. A third example, taken from a test system supplied with DEWorkstation, gives before and after plots of the system load as a function of distance from the substation.

6.2 Definition of Terms

The phase balancing application recommends rephasing of single and two phase laterals for improvement in load imbalance at the substation. A lateral is defined as all components with the same phasing connected in series. Phase balancing does not attempt to rephase the three phase portion of the system. In addition, no transposition within a lateral is performed.

Imbalance is a measure of the load distribution across all phases present for a given component. The imbalance is calculated using the formula given in Equation 6.1 [6.2]. The imbalance ranges from 0 to 2 where 0 is perfectly balanced and 2 is perfectly out of balance. Because of this range, the imbalance is normally adjusted to yield a 0% to 100% range as shown.

$$\text{imbalance} = (\max\{\text{deviation}\} / \text{average}) * 100 / 2 \qquad 6.1$$

where:

average = (phase a load + phase b load + phase c load) / sNumPh

sNumPh = number of phases present

deviation = | (phase i - average) |

i = a, b, c

Phase balancing calculations can be based on either power (kW) or complex power magnitude (kVA). For this application, phase imbalance is calculated from line flows which come from a power flow solution. Because DEWorkstation has an open architecture platform, power flow results are readily available [6.3].

6.3 DAOP Modifications for Phase Balancing

Instead of continually defining a set of ending loads as described in Chapter 3, the phase balancing adaptation of DAOP starts by locating all laterals in the circuit, determines the total flow through each, and disconnects them from the rest of the circuit so that only the three phase portion of the circuit (the spine) is connected to the substation. Laterals directly connected to the spine are reconnected one at a time. The lateral with the most heavily loaded phase is reconnected first. The laterals are added in like manner from most to least loaded until all laterals are reconnected to the spine. Next, all remaining laterals (single phase laterals connected to two phase laterals) are reconnected from most to least loaded.

The phase balancing scheme does not add a discrete unit of load to the system with each step but adds the entire load modeled on a single lateral. Therefore, variable step adaptations do not apply to the phase balancing routine; however, further research may reveal alternative methods of determining the ordering of lateral reconnections. Note that the number of computational steps required to reach convergence is simply the number of laterals within the system.

For each lateral reconnection, the phase balancing routine evaluates all possible rephasing options and selects that which produces the minimum imbalance at the substation. Because this is a phase balancing routine, substation imbalance is minimized, not system losses. In most cases, the improvement in the imbalance at the substation results in a reduction of system loss.

6.4 Example 6.1: Phase Balancing Mechanics

Example 6.1 illustrates how the phase balancing routine defines, sets reconnection order, and rephases the laterals in a power system. Figure 6.1 gives the circuit that is used to demonstrate the mechanics of the routine. Phasing and loading (in kW) are indicated beside each line section in the figure. The example uses real power (kW) to calculate imbalance and losses are neglected.

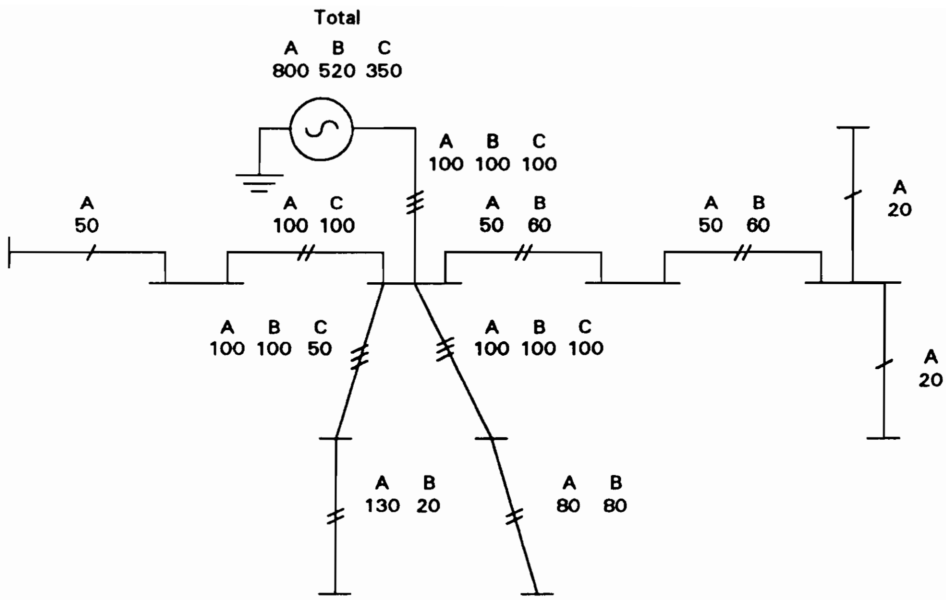

Figure 6.1 System for Example 6.1

Phase balancing locates all laterals in the system, determines the flow through each, and detaches them from the rest of the circuit. Figure 6.2 shows the system with all laterals removed from the system.

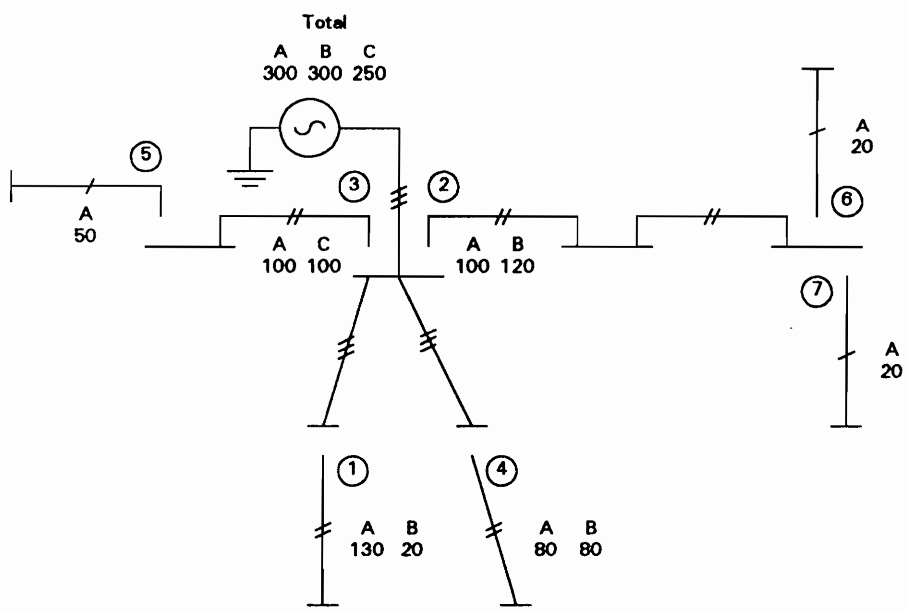

Figure 6.2 Phase Balancing Lateral Location and Reconnection Order

The load at the substation is from the three phase portion of the system. Note that lateral ② is two line sections connected in series. Thus, the load on this lateral is the sum of the load on the two individual line sections.

Phase balancing connects all laterals connected to the three phase portion of the system first. These laterals, ① through ④ define the set of *first level* laterals. Of the first level laterals, phase balancing reconnects that which has the largest loaded phase first. Even though laterals ② through ④ have more total load than ①, ① has the most heavily loaded phase and is, therefore, reconnected first.

Table 6.1 shows all possible reconnections and the resulting imbalance at the substation for lateral ①. The imbalance at the substation before the reconnection of lateral ① is 0.1176 (5.88%). For this lateral, phase balancing recommends moving phase A to C and not moving phase B. Note that multiple rephasing options often give the same imbalance at the substation. In such cases, phase balancing attempts to move as few phases as possible.

Table 6.1 Reconnection Choices for Lateral ① of Example 6.1

Movement	Phase A	Phase B	Phase C	Imbalance	% Imbalance
A->A, B->B	430.00	320.00	250.00	0.2900	14.50
A->A, B->C	430.00	300.00	270.00	0.2900	14.50
A->C, B->B	300.00	320.00	380.00	0.1400	7.00*
A->B, B->A	320.00	430.00	250.00	0.2900	14.50
A->B, B->C	300.00	430.00	270.00	0.2900	14.50
A->C, B->A	320.00	300.00	380.00	0.1400	7.00

* Rephasing Selection

After connecting all first level laterals (① through ④), phase balancing reconnects the remaining laterals referred to as *second level* laterals. Second level laterals, if they exist, are always single phase laterals connected to two phase laterals. For this example, laterals ⑤ through ⑦ are second level laterals.

Table 6.2 shows all possible reconnections and the resulting imbalance at the substation for the most heavily loaded second level lateral, ⑤. Phase balancing recommends leaving lateral ⑤ as phase A. One must be aware that first order laterals, to which second order laterals connect, may have been rephased. For this reason, the rephasing choices for second level laterals may not be intuitive by examining the original system configuration.

Table 6.2 Reconnection Choices for Lateral ⑤ of Example 6.1

Movement	Phase A	Phase B	Phase C	Imbalance	% Imbalance
A->A	570.00	500.00	560.00	0.0798	3.99*
A->B	520.00	500.00	610.00	0.1227	6.14

* Rephasing Selection

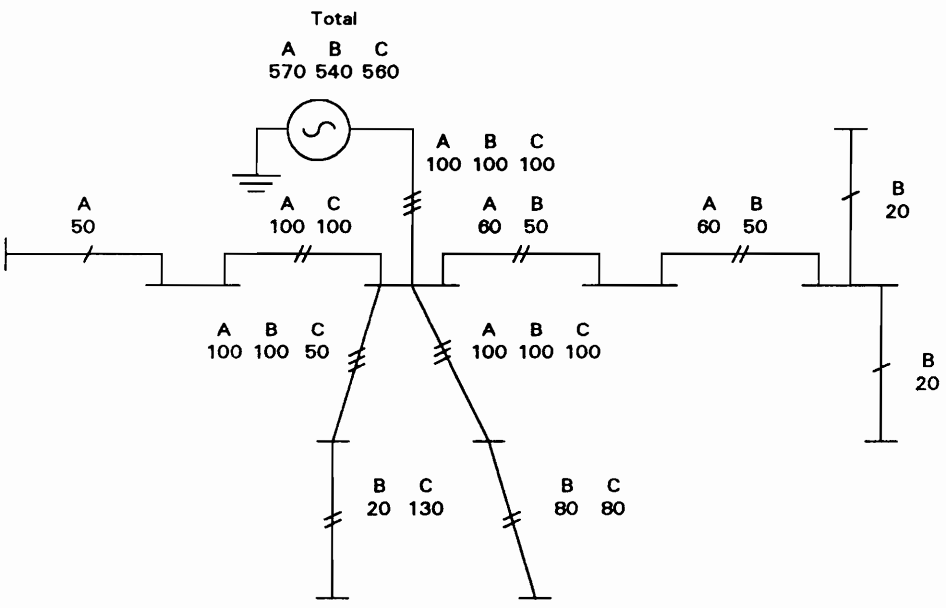

Figure 6.3 Rephased Design System of Example 6.1

Figure 6.3 shows the rephased design system produced by phase balancing. The imbalance at the substation of the original system is 0.4371 (21.86%). After rephasing, the design system has an imbalance of 0.0299 (1.50%), an improvement of 93.16%. Appendix B contains the spreadsheet used to solve Example 6.1.

6.5 DEWorkstation Overview

The Electric Power Research Institute (EPRI) sponsored Distribution Engineering Workstation (DEWorkstation) is a software package which provides the power engineer with an integrated data and applications environment. Because DEWorkstation uses a standardized data scheme (database), engineering application modules access and exchange database data through common data areas. The engineering applications are analysis, design, and operation modules [6.1].

In addition to providing a set of graphical user interface (GUI) functions, DEWorkstation is based on an open architecture platform [6.3]. The open architecture platform allows the engineering applications to invoke and obtain results from other modules. In addition, users can add, delete, and replace the application modules [6.4].

The phase balancing algorithm described in this chapter is one of two design applications provided with DEWorkstation Version 1.0. A design application differs from an analysis application in that it produces results as the user experiments with model parameters. Although the phase balancing module will recommend the rephasing of all laterals, the user can attach (rephase) the laterals in any desired configuration. Figure 6.4 is an example of a dialog box which allows the user to pick the rephasing of a two phase lateral.

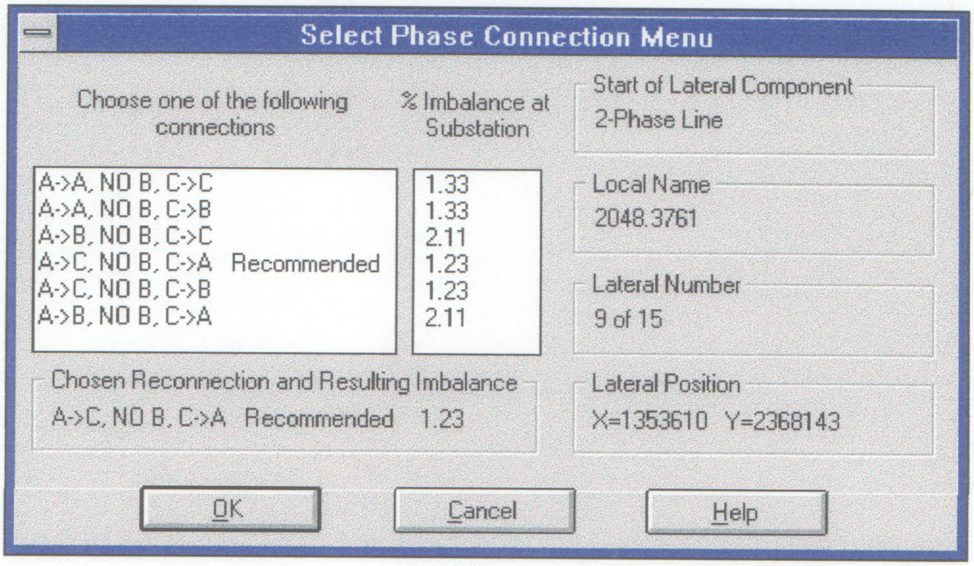

Figure 6.4 Phase Balancing Rephasing Dialog Box

Note that the user has the option of selecting the recommended rephasing or can select any other physically allowable rephasing. Not choosing the recommended rephasing for a particular lateral affects the remaining lateral rephasing recommendations and possibly the choices. This feature allows the user to develop and experiment with multiple designs.

6.6 Example 6.2: Hot Springs North

The DEWorkstation model of the Hot Springs North circuit on the Arkansas Power and Light power system is show in Figure 6.5. The model contains 48 line sections (15 laterals), 4 capacitors, and 2 reclosers. The total system load is 6,822.3 kW and the imbalance at the substation before rephasing is 0.1714 (8.57%). The phase balancing routine produced a design with a substation imbalance of 0.0450 (2.25%) which is an improvement of 73.75%. An output report generated by phase balancing containing all lateral movements can be found in Appendix C.

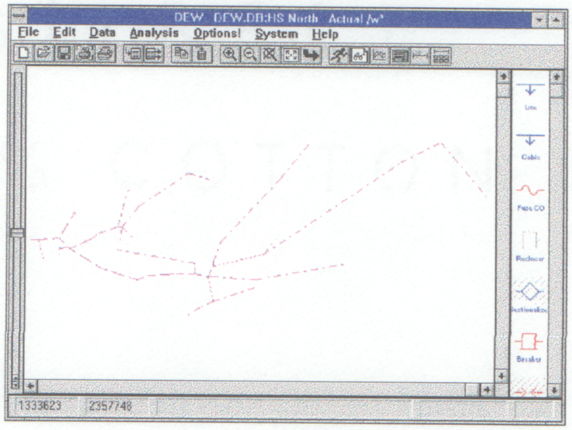

Figure 6.5 AP&L North DEWorkstation Model

6.7 *Example 6.3: Hot Springs Carpenter*

The DEWorkstation model of the Hot Springs Carpenter circuit on the Arkansas Power and Light power system is show in Figure 6.6. The model contains 56 line sections (15 laterals), 8 capacitors, and 1 recloser. The total system load is 12,246.35 kW and the imbalance at the substation before rephasing is 0.1044 (5.22%). The phase balancing routine produced a design with a substation imbalance of 0.0154 (0.77%) which is an improvement of 85.25%. An output report generated by phase balancing containing all lateral movements can be found in Appendix C.

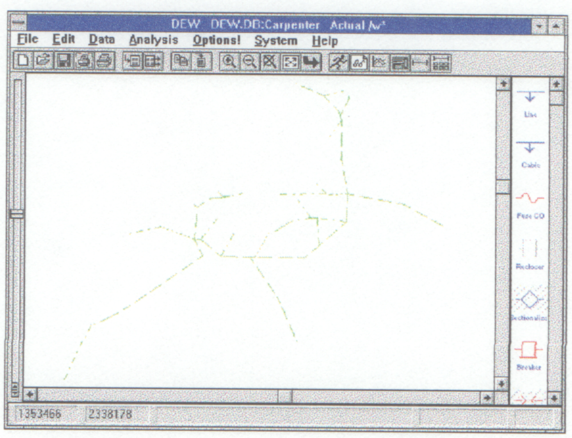

Figure 6.6 AP&L Carpenter DEWorkstation Model

6.8 *Example 6.4: VA Tech Demo System*

The purpose of this example is to create plots of the system load as a function of distance to the substation before and after phase balancing. The DEWorkstation model of the VA Tech Demo System, a fictitious system supplied with the software for demonstration purposes, will be used and is shown in Figure 6.7.

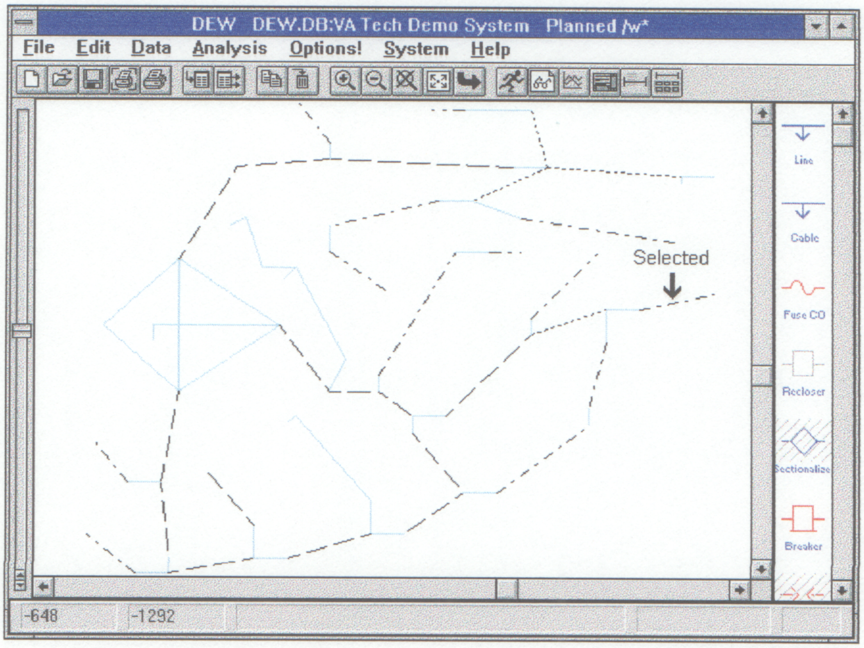

Figure 6.7 VA Tech Demo System DEWorkstation Model

Using DEWorkstation, all application results can be plotted against either time or distance. Plots of data made against distance are not commonplace but can give a great deal of information when appropriate. One such use is plotting loading conditions on a system before and after running phase balancing. When creating a load versus distance plot, the user must pick a component from which the load is to be plotted. From this component, the path from which the component receives power from the substation is traced.

The load is summed from the selected component back to the substation along this path. This data allows a load versus distance to the substation plot to be created.

The arrow on the system model in Figure 6.7 indicates the component from which the load plots are made. Figures 6.8 and 6.9 are plots of the load from this component to the substation before and after phase balancing the system, respectively.

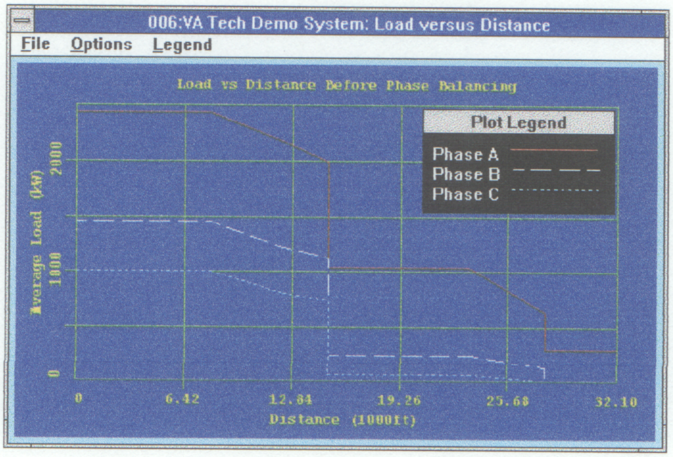

Figure 6.8 Load Versus Distance Before Phase Balancing

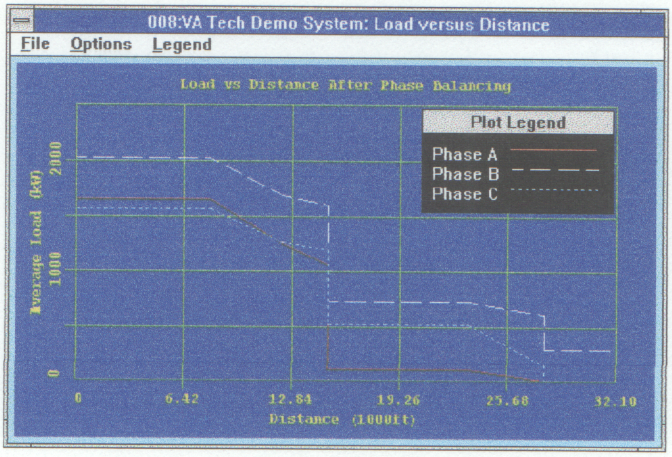

Figure 6.9 Load Versus Distance After Phase Balancing

For comparison purposes, both of these graphs have the same scale on both axes. Note how the phase load curves are more tightly grouped for the rephased circuit. This indicates that the component loads are more evenly distributed across the phases.

6.9 Conclusions

This chapter describes a new phase balancing algorithm for electrical distribution systems. The algorithm uses basic principles of the DAOP algorithm. Implemented as a design module in the EPRI DEWorkstation software package, the routine has been tested on real circuits in the Arkansas Power and Light distribution system in Hot Springs, Arkansas. Results of these tests have been independently verified by AP&L personnel.

6.10 References

[6.1] R. Broadwater, H. Shaalan, P. Dolloff, and H. Ng, *EPRI Distribution Engineering Workstation and Measurement Data*, Proceedings of the 4th International Symposium on Distribution Automation and Demand Side Management, Orlando, FL, January 17-20, 1994.

[6.2] IEEE Power System Engineering, IEEE Recommended Practice for Electric Power Distribution for Industrial Plants (IEEE Red Book), ANSI/IEEE Std 141 - 1993.

[6.3] R. Broadwater, P. Dolloff, M. Ellis, N. Singh, T. Corbin, and H. Ng, *EPRI Distribution Engineering Workstation Design Concepts*, EPRICON Conference Proceedings, Scottsdale, AR, December 9-11, 1992.

[6.4] R. Broadwater, J. Thompson, M. Ellis, H. Ng, N. Singh, and D. Loyd, Application Programmer interface for the EPRI Distribution Engineering Workstation, IEEE Transactions on Power Systems, Vol. 10, No. 1, February 1995, pp. 499 - 505.

Chapter Seven

Economic Dispatch

7.1 Introduction

In this chapter the application of DAOP to the economic dispatch problem is demonstrated. Originally developed for optimal power flow solutions in distribution systems, DAOP must consider system constraints such as generation and transmission limits to solve the economic dispatch problem.

A three part economic dispatch example problem is solved. Part (a) of this example solves the economic dispatch problem considering only generation costs. Part (b) includes generation limits while Part (c) includes transmission line (security) limits. Parts (a) and (b) compare the results from the Lagrange multiplier approach to those of the DAOP algorithm. Discussion of the incorporation of transmission losses into the economic dispatch problem and the DAOP algorithm is given. A second example involving transmission losses is given in Appendix D.

7.2 Economic Dispatch Problem Statement

When operating a power system, not only must the power demands of customers be met, but they should be met while minimizing the cost of supplying the system load. The problem of determining the most economical configuration of the generators to supply the system load is known as *economic dispatch* [7.1].

Each generating unit has an operating cost rate, F_i, defined by the generating unit heat rate and fuel cost among other costs such as the fixed and variable operating and maintenance costs. The total operating cost rate for the entire power system, Ft, is the sum of the cost rates of the individual generating units. Reduced to its simplest form, the only constraint the economic dispatch problem must meet states that total generation for the entire power system, Pt, must equal the total load, Pr. The total generation, Pt, is the sum of the power outputs of the individual generating units, P_i. Therefore, the optimal economic dispatch of the generating units is reached by minimizing Ft while supplying the total system load as shown in (7.1) through (7.4).

$$Pt = P_1 + P_2 + P_3 + \ldots + P_n \tag{7.1}$$

$$Ft(Pt) = F_1(P_1) + F_2(P_2) + F_3(P_3) + \ldots + F_n(P_n) \tag{7.2}$$

$$\text{Minimize } Ft(Pt) \tag{7.3}$$

$$\text{Subject to: } Pr - Pt = 0 \tag{7.4}$$

The minimization problem can be solved using the Lagrange multiplier approach. To form the Lagrange function, £, the constraint equation is multiplied by the Lagrange multiplier, λ, and added to the objective function as shown in (7.5).

$$£ = Ft + \lambda(Pr - Pt) \tag{7.5}$$

To solve for the minimum, partial derivatives of (7.5) with respect to each of the independent variables (P_i) and λ are taken and set equal to zero. The resulting system of simultaneous equations may then be solved. The Lagrange multiplier, λ, represents the incremental cost rate of the generating units.

Using DAOP to solve the economic dispatch problem is very similar to the procedure described in Chapter 3. When DAOP starts, all of the generating units are assumed to be completely unloaded. From this starting point, the algorithm determines the operating point of each generating unit such that the system load is supplied in a least cost fashion by adding the chosen load increment to the unit which produces the minimum cost at each step. Note that the cost is a combination of line losses and generation costs.

7.3 Incorporating Generating Unit Limits

There are additional limiting constraints to consider when formulating the economic dispatch problem, one of which is generating capacity limits. That is, the power output from each generating unit must not exceed the maximum power output permitted by that particular unit. In addition, there may be lower generation limits as well. Generating capacity limits take the form of inequality constraints as shown in (7.6).

$$P_{i\,min} \leq P_i \leq P_{i\,max} \tag{7.6}$$

When considering system inequality constraints on generation levels, the final solution reached by the Lagrange multiplier approach, as presented in the previous section, may include generation levels for the individual units which are outside of the allowable operating range.

When generating limits have been violated, an iterative process must be initiated which includes clamping the offending independent variables at their limits when using the Lagrange multiplier approach to reach a solution. From this point on, not all generating units will operate at the same incremental cost rate. Therefore, incremental cost rates must be defined for each generating unit, given by λ_i. Only those generating units which are operating within their respective operating ranges will operate at the same incremental cost rate, referred to as the system λ. At the conclusion of each iteration, the individual incremental cost rates are compared to the system λ.

The individual incremental cost rates, λ_i , can be used to determine the cost to operate the generating units with respect to each other. The cost to operate those generating units which initially exceeded their maximum operating capacity and have been clamped to their maximum is less than those defining the system λ. For these units, λ_i is less than the system λ. Conversely, the cost to operate those generating units initially operated below their minimum operating capacity and have been forced to their minimum is greater than those defining the system λ. For these units, λ_i is greater than the system λ.

When using the DAOP algorithm, additional computational steps (iterations) are not required when system constraints are encountered. The constraints are never violated during the DAOP solution process thereby guaranteeing that the final answer, if feasible, is within the chosen step size of the optimum. When DAOP encounters a constraint, the algorithm continues to converge on the optimum by avoiding solution paths which violate the constraint. In so doing, additional iterations are not required to adjust intermediate solutions which violate the system constraints. *In fact, the DAOP computational burden may decrease as constraints are met.*

For each step, DAOP must determine if the load increment may be supplied by each individual generating unit without violating the maximum capacity limit of the generating unit. Violation occurs when the output of a unit exceeds its maximum when the load increment is added. In such a case, the output of this generating unit remains at its maximum and is no longer considered a candidate for incrementation. Thus, the final solution does not allow a generating unit to exceed its allowable maximum capacity. Note that the computational burden of DAOP decreases as generating units reach their maximum output. When minimum generation output levels are considered, the DAOP algorithm must be modified. One method is to assume that all generating units are to be dispatched. With this assumption, the initial DAOP starting point is no longer a completely unloaded system. Instead, DAOP starts with every unit set to its minimum output level.

7.4 *Example 7.1: Economic Dispatch*

This example, is divided into three parts in which the first two are solved using both the Lagrange multiplier approach and the DAOP algorithm. Part (a) considers only the cost of generation. Part (b) shows how the two solution methods behave when generation constraints are encountered. Finally, transmission line constraints are imposed in Part (c).

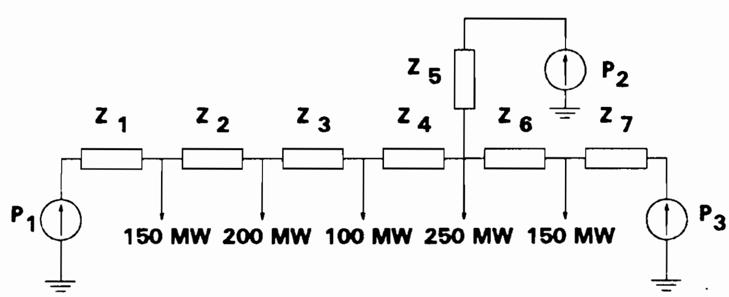

Figure 7.1 Economic Dispatch Test System

Both economic dispatch examples solved in this chapter use the DC system approximation shown in Figure 7.1. The total load to be supplied is 850 MW. All transmission lines are 100 miles in length and have an impedance of 0.03 + j0.3 ohms/mile.

Table 7.1 Generating Unit 1 Specifications

Type of Generation	Coal-Fired Steam
Maximum Output	600 MW
Minimum Output	150 MW
Fuel Cost	1.1 \$/MBtu
Input-Output Curve	$H_1(P_1) = 510.0 + 7.20P_1 + 0.00142P_1^2$ MBtu/Hr

Table 7.2 Generating Unit 2 Specifications

Type of Generation	Oil-Fired Steam
Maximum Output	400 MW
Minimum Output	100 MW
Fuel Cost	1.0 \$/MBtu
Input-Output Curve	$H_2(P_2) = 310.0 + 7.85P_2 + 0.00194P_2^2$ MBtu/Hr

Table 7.3 Generating Unit 3 Specifications

Type of Generation	Oil-Fired Steam
Maximum Output	200 MW
Minimum Output	50 MW
Fuel Cost	1.0 \$/MBtu
Input-Output Curve	$H_3(P_3) = 78.0 + 7.97P_3 + 0.00482P_3^2$ MBtu/Hr

The three generating units taken from [7.1] are described in Tables 7.1 through 7.3 where the functions $H_i(P_i)$ are input-output curves. The input-output curves are multiplied by the fuel cost to give the cost rate curves, $F_i(P_i)$, as shown in (7.7).

$$F_i(P_i) = H_i(P_i) * \text{Fuel Cost} \tag{7.7}$$

From the input-output curves and the fuel costs, the cost rate equations for the three generating units are given by

$$F_1(P_1) = 1.1\ (\ 510.0 + 7.20P_1 + 0.00142\ P_1^2\ ) \tag{7.8}$$

$$F_2(P_2) = 1.0\ (\ 310.0 + 7.85P_2 + 0.00194P_2^2\ ) \tag{7.9}$$

$$F_3(P_3) = 1.0\ (\ 78.0 + 7.97P_3 + 0.00482P_3^2\ )\ . \tag{7.10}$$

Part (a): Cost of Generation

The economic dispatch problem is mathematically expressed by

$$\text{Minimize } F_1(P_1) + F_2(P_2) + F_3(P_3) \tag{7.11}$$

Subject to:

$$P_1 + P_2 + P_3 = 850.0 \tag{7.12}$$

$$150.0 \le P_1 \le 600.0 \tag{7.13}$$

$$100.0 \le P_2 \le 400.0 \tag{7.14}$$

$$50.0 \le P_3 \le 200.0\,. \tag{7.15}$$

Using the Lagrange multiplier approach to solve the problem in (7.11) through (7.15), the Lagrange function is written:

$$\begin{aligned} \pounds = {} & 949.0 + 7.92P_1 + 0.001562P_1^2 \\ & + 7.85P_2 + 0.00194P_2^2 \\ & + 7.97P_3 + 0.00482P_3^2 \\ & + \lambda\,(850.0 - P_1 - P_2 - P_3)\,. \end{aligned} \tag{7.16}$$

Taking the partial derivatives of (7.16), setting them equal to zero, and solving the resulting set of equations yields the results given in Table 7.4. Note that the generation inequality constraints are not violated.

Table 7.4 Lagrange Multiplier and DAOP Solutions for Example 7.1 Part (a)

Generating Unit	Lagrange Multiplier	DAOP Algorithm
P_1	393.17	400
P_2	334.61	325
P_3	122.23	125
Total Cost \$/Hr	8,194.36	8,194.64

For the DAOP algorithm, a load increment, Lp, must be selected. Let the load increment be given by

$$Lp = 25.0 \text{ MW}\,. \tag{7.17}$$

Given the total system load and using a fixed load increment, the number of computational steps to reach the solution may be calculated by dividing the total load by the step size. At each step 25.0 MW of load will be supplied; thus, 34 steps are required to reach the solution. Table 7.5, giving only a few of the 34 steps, organizes the DAOP algorithm solution. The Δ*Cost* column indicates the additional system cost due to supplying the load increment by the corresponding generating unit as indicated under the *Generating Unit* column. The *Total Cost* column gives the total system cost. The <= symbol under the *Selection* column indicates which generating unit output increment yields the minimum total cost for that particular step. The results from the DAOP algorithm are given in Table 7.4.

Table 7.5 DAOP Solution Table for Example 7.1 Part (a)

Step	Generating Unit	Δ Cost $/Hr	Total Cost $/Hr	Selection	Generation Level MW
1	1	1,147.98	1,147.98		25
	2	1,146.46	1,146.46	<=	
	3	1,151.20	1,151.26		
2	1	198.98	1,345.44	<=	50
	2	199.98	1,346.35		
	3	202.26	1,348.72		
3	1	200.93	1,546.37		75
	2	199.89	1,545.33	<=	
	3	202.26	1,547.70		
⋮	⋮	⋮	⋮	⋮	⋮
33	1	228.26	7,968.08		825
	2	226.56	7,966.38	<=	
	3	232.39	7,972.21		
34	1	228.26	8,194.64	<=	850
	2	228.99	8,195.37		
	3	232.39	8,198.77		

Table 7.4 shows that the power outputs from the generating units obtained from the DAOP algorithm lie within the increment Lp of the solution obtained by the Lagrange multiplier method. For this example, the total generation cost computed by the two methods differs by 0.28 $/Hr. The accuracy of the DAOP algorithm is dependent only on the load increment, all else being equal. As the load increment decreases the accuracy of the solution increases. Thus, it can be shown that using a load increment of 0.01 MW, the DAOP solution will equal the Lagrange multiplier solution to two decimal places.

The DAOP algorithm can be defined on a spread sheet. To reach a spread sheet solution, first assign a column to each generating unit. Next, define each cost rate curve on separate columns as a function of the corresponding generating unit column. Finally, define a single column as the sum of the cost rate curve rows. This column is the total system cost. By manually incrementing the generating unit columns, the minimum cost at each step can be identified. To continue to the next step, the appropriate generating unit column is incremented. The process stops once the total system load is met.

Part (b): Generation Constraints

This part of the example shows how the Lagrange multiplier method and DAOP algorithm behave when the generating inequality constraints are encountered and how the DAOP algorithm does not require additional computational steps. Suppose the cost of fuel for generating Unit 1 decreases to 0.9 $/MBtu. The first iteration of the Lagrange multiplier method yields the results given in Table 7.6. Note that generating units 1 and 3 violate their respective generation inequality constraints.

Table 7.6 First Iteration Results of the Lagrange Multiplier for Example 7.1 Part (b)

Generating Unit	Lagrange Multiplier
P_1	704.60
P_2	111.80
P_3	32.60

To reach a feasible solution, Unit 1 is forced to operate at its maximum operating capacity, 600 MW, Unit 3 is forced to operate at its minimum operating capacity, 50 MW, and Unit 2 is computed to be 200 MW. Since Unit 2 is operating within its limits, the incremental cost for Unit 2, λ_2, must now be calculated by setting it equal to the incremental cost of Unit 2 at 200 MW. λ_1 and λ_2 are also calculated for the other two units at their adjusted outputs and compared to λ_2. These values are given below.

$$\lambda_1 = 8.016 \tag{7.18}$$

$$\lambda_2 = 8.626 \tag{7.19}$$

$$\lambda_3 = 8.452 \tag{7.20}$$

Thus, the incremental cost to run Unit 1 is less than Unit 2 which means that Unit 1 is correctly set at its maximum. The incremental cost to run Unit 3 is also less than Unit 2. Because the output of Unit 3 is less than Unit 2, their outputs must be recomputed using the Lagrange multiplier approach. The final solution is given in Table 7.7.

Table 7.7 Lagrange Multiplier and DAOP Solutions for Example 7.1 Part (b)

Generating Unit	Lagrange Multiplier	DAOP Algorithm
P_1	600.00	600
P_2	187.10	175
P_3	62.90	75
Total Cost $/Hr	7,252.11	7,253.10

This part of the example will now be solved using the DAOP algorithm with the load increment given by (7.17). Table 7.8 organizes the DAOP solution and demonstrates a few of the steps.

Note that on step 25, Unit 1 has reached its maximum output. Subsequent steps will no longer attempt to add the load increment to this unit. This allows DAOP to converge on the optimum by avoiding solutions paths which violate the constraint. From this point, the DAOP algorithm will determine how the balance of the system load will be met by the remaining units.

Even though generation constraints have been added to the economic dispatch problem, the number of computational steps to reach the solution remains fixed. In addition, the number of computations per step decreases once the constraint on Unit 1 is encountered.

Table 7.8 DAOP Solution Table for Example 7.1 Part (b)

Step	Generating Unit	Δ Cost $/Hr	Total Cost $/Hr	Selection	Generation Level MW
1	1	1,009.80	1,009.80	<=	25
	2	1,044.46	1,044.46		
	3	1,049.26	1,049.26		
2	1	164.39	1,174.19	<=	50
	2	197.46	1,207.26		
	3	202.26	1,212.06		
⋮	⋮	⋮	⋮	⋮	⋮
25	1	199.54	5,392.54	<=	625
	2	199.89	5,392.89		
	3	202.26	5,395.26		
26	2	199.89	5,592.43	<=	650
	3	202.26	5,594.80		
⋮	⋮	⋮	⋮	⋮	⋮
34	2	214.44	7,253.23		850
	3	214.31	7,253.10	<=	

Again, it can be shown that the DAOP solution will approach the Lagrange multiplier solution more accurately as the load increment is reduced.

Part (c): Transmission Line Constraints

Another type of constraint which may be considered in the economic dispatch problem is transmission line capacities. Transmission line constraints, which indicate the maximum allowable current or power capacity, take the form of inequality constraints as given in (7.21).

$$Ps_i \leq Ps_{i\,max} \tag{7.21}$$

When using DAOP, these constraints are handled in the same manner as the upper generating capacity limits as described earlier. For each step, DAOP must determine if the load increment may be supplied by each individual generating unit without violating the maximum capacity limit of the transmission lines. Violation occurs when the amount of current or power on a transmission line exceeds its carrying capacity when the load increment is added. In such a case, the load increment cannot be added to the selected generating unit. Thus, the final solution does not allow a transmission line to exceed its allowable maximum capacity.

This part of the example incorporates transmission line inequality constraints, Ps_i. With the system as described in Part (a) of this example, let us impose a transmission limit of 375 MW on each transmission line section. The economic dispatch problem is mathematically expressed by

$$\text{Minimize } F_1(P_1) + F_2(P_2) + F_3(P_3) \tag{7.22}$$

Subject to:

$$P_1 + P_2 + P_3 = 850.0 \tag{7.23}$$

$$150.0 \leq P_1 \leq 600.0 \tag{7.24}$$

$$100.0 \leq P_2 \leq 400.0 \tag{7.25}$$

$$50.0 \leq P_3 \leq 200.0 \tag{7.26}$$

$$Ps_i \leq 375.0, \quad i = 1, 7\ . \tag{7.27}$$

The results from the DAOP algorithm using step sizes of 25 MW and 0.01 MW are given in Table 7.9.

Table 7.9 DAOP Solutions Using Two Different Load Increments for Example 7.1 Part (c)

Generating Unit	Lp = 25 MW	Lp = 0.01 MW
P_1	375	375.00
P_2	350	347.56
P_3	125	127.44
Total Cost $/Hr	8,195.37	8,195.33

The transmission limit on line section Z_1 restricts Unit 1 to an output of 375 MW instead of 393.17 MW as in Part (a). Consequentially, this difference in generation must be made up by units 2 and 3. This transmission limit increases the total system cost by 0.97 $/Hr.

Even though transmission line constraints have been added to the economic dispatch problem, the number of computational steps to reach the solution remains fixed. In addition, the number of computations per step decreases once the constraint on line section Z_1 is encountered.

7.5 Incorporating Transmission Line Losses

Another system cost to consider when formulating the economic dispatch problem is transmission line losses. An incremental change in generation causes an incremental change in total system line losses. The inclusion of transmission losses into the economic dispatch problem increases the difficulty in finding a solution using traditional methods. However, preliminary studies using the DAOP algorithm to solve this problem indicate that the computations are straight forward. An economic dispatch problem including transmission line losses is given in Appendix D.

7.6 Conclusions

This chapter describes the application of DAOP to the economic dispatch problem in electrical transmission systems. Discussion and examples have demonstrated how DAOP can incorporate generation costs, generation limits, and transmission line limits. Additionally, the incorporation of transmission losses into the objective function and how DAOP handles this non-linear constraint has been shown.

Examples of the DAOP algorithm have shown that additional computational steps are not required when incorporating system constraints, unlike when using the Lagrange multiplier approach. Furthermore, the computational burden may decrease as constraints are encountered.

For comparison purposes, data for the test circuits has been taken from a text book respected in the area of economic dispatch [7.1]. Results of the routine have been compared and verified against results obtained by the Lagrange multiplier approach.

7.7 References

[7.1] Allen J. Wood and Bruce F. Wollenberg, Power Generation, Operation, and Control, John Wiley and Sons, New York, NY, 1984.

Chapter Eight

Conclusions

8.1 Introduction

This dissertation presents the development work of the Discrete Ascent Optimal Programming (DAOP) method, a new algorithm for optimization in electrical power systems. Originally developed as an optimal power flow technique, the fundamental properties of DAOP have successfully been applied to switch placement design, phase balancing, and economic dispatch.

8.2 Contributions

This dissertation describes the switch placement design algorithm for electrical distribution systems. The switch placement algorithm is an extension of the DAOP algorithm. The routine has been tested on a real circuit in the Arkansas Power and Light (AP&L) distribution system in Hot Springs, Arkansas. Results of these tests have been independently verified by AP&L personnel. This dissertation is not only the first report of applying DAOP to the switch placement design problem but is also the first report of an optimal switch placement design algorithm. Future work can take the switch placement design one step further into the realm of reconfiguration and restoration.

This dissertation describes a new phase balancing algorithm for electrical distribution systems. The algorithm uses basic principles of the DAOP algorithm. Implemented as a design module in the Electric Power Research Institute (EPRI) Distribution Engineering Workstation (DEWorkstation) software package, the routine has been tested on real circuits in the AP&L distribution system in Hot Springs, Arkansas. Results of these tests have been independently verified by AP&L personnel. This dissertation is the first report of applying DAOP to the phase balancing design problem.

This dissertation describes the adaptation of DAOP on the economic dispatch problem in electrical transmission systems. The adaptations of DAOP to include generation costs, generation limits, and transmission line limits into the economic dispatch problem have been demonstrated. DAOP test results have been verified against solutions obtained by the Lagrange multiplier approach. DAOP has been successfully tested on economic dispatch problems which have incorporated transmission line losses, a non-linear constraint. This dissertation is the first report of applying DAOP to the economic dispatch problem.

8.3 Conclusions

Research and testing have proven that the Discrete Ascent Optimal Programming algorithm is well suited to solve a number of optimization problems in power systems. For the unconstrained problem, a proof of convergence has been developed which shows DAOP to converge to the correct solution in a finite and determinable number of steps without the threat of divergence. The proof also shows that the final solution reached by DAOP is within the discrete step size used. Therefore, the degree of accuracy desired can be attained with proper step size selection. Comparisons of DAOP results with those of established solving techniques verify DAOP's convergence to the correct solution. Tests have shown that the computational burden of DAOP is far less than that of the Lagrange multiplier technique when solving the economic dispatch problem with transmission line losses.

Appendix A

Sub-Loss Function Derivations

The sub-loss functions (eqs. (4.1) through (4.8)) can be derived from Figure 4.1 by combining basic network analysis techniques with the circuit topology that applies at each step of the DAOP solution process. For this example, a step size, Lp, of 1 amp is used. When the solution process begins, only 1 amp of load will be supplied. Therefore, only the 1 amp and 2 amp loads drawn with dashed lines in Figure 4.2 (a) will be considered. The sub-loss function can be derived from the following two equations.

$$F_1(I_1, I_2) = (1)(I_1)^2 + (2)(I_2)^2 \tag{A.1}$$

$$I_1 + I_2 = 1 \tag{A.2}$$

Combining Equations (A.1) and (A.2) yields Equation (4.1), repeated below.

$$F_1(I_2) = 1 - 2I_2 + 3I_2^2 \tag{4.1}$$

The derivation of the next sub-loss function depends on the DAOP solution of Equation (4.1). The solution specifies source S_1 to supply 1 amp and source S_2 to supply 0 amps of load. Examining the system topology, the 1 amp load becomes completely supplied and the adjacent 2 amp load becomes a new ending load. Hence, step two of the DAOP solution process uses the system configuration of Figure 4.2 (b). The second sub-loss function can be derived from the following two equations.

$$F_2(I_1, I_2) = (1)(I_1)^2 + (2)(I_1-1)^2 + (2)(I_2)^2 \tag{A.3}$$

$$I_1 + I_2 = 2 \tag{A.4}$$

Combining Equations (A.3) and (A.4) yields Equation (4.2), repeated below.

$$F_2(I_2) = 6 - 8I_2 + 5I_2^2 \tag{4.2}$$

The remaining sub-loss functions can be derived in like manner.

Appendix B

Example 6.1 Spreadsheet Solution

300.00	300.00	250.00							
Phase A	Phase B	Phase C	Dev A	Dev B	Dev C	Max Dev			Selection
430.00	320.00	250.00	96.67	13.33	83.33	96.67	0.2900	14.5000	A->A, B->B
430.00	300.00	270.00	96.67	33.33	63.33	96.67	0.2900	14.5000	A->A, B->C
300.00	320.00	380.00	33.33	13.33	46.67	46.67	0.1400	7.0000	* A->C, B->B
320.00	430.00	250.00	13.33	96.67	83.33	96.67	0.2900	14.5000	A->B, B->A
300.00	430.00	270.00	33.33	96.67	63.33	96.67	0.2900	14.5000	A->B, B->C
320.00	300.00	380.00	13.33	33.33	46.67	46.67	0.1400	7.0000	A->C, B->A
ave =	333.33								

300.00	320.00	380.00							
Phase A	Phase B	Phase C	Dev A	Dev B	Dev C	Max Dev			Selection
400.00	440.00	380.00	6.67	33.33	26.67	33.33	0.0820	4.0984	A->A, B->B
400.00	320.00	500.00	6.67	86.67	93.33	93.33	0.2295	11.4754	A->A, B->C
300.00	440.00	480.00	106.67	33.33	73.33	106.67	0.2623	13.1148	A->C, B->B
420.00	420.00	380.00	13.33	13.33	26.67	26.67	0.0656	3.2787	* A->B, B->A
300.00	420.00	500.00	106.67	13.33	93.33	106.67	0.2623	13.1148	A->B, B->C
420.00	320.00	480.00	13.33	86.67	73.33	86.67	0.2131	10.6557	A->C, B->A
ave =	406.67								

420.00	420.00	380.00							
Phase A	Phase B	Phase C	Dev A	Dev B	Dev C	Max Dev			Selection
520.00	420.00	480.00	46.67	53.33	6.67	53.33	0.1127	5.6338	* A->A, C->C
520.00	520.00	380.00	46.67	46.67	93.33	93.33	0.1972	9.8592	A->A, C->B
420.00	520.00	480.00	53.33	46.67	6.67	53.33	0.1127	5.6338	A->B, C->C
520.00	420.00	480.00	46.67	53.33	6.67	53.33	0.1127	5.6338	A->C, C->A
520.00	520.00	380.00	46.67	46.67	93.33	93.33	0.1972	9.8592	A->B, C->A
420.00	520.00	480.00	53.33	46.67	6.67	53.33	0.1127	5.6338	A->C, C->B
ave =	473.33								

520.00	420.00	480.00								
Phase A	Phase B	Phase C	Dev A	Dev B	Dev C	Max Dev		Selection		
600.00	500.00	480.00	73.33	26.67	46.67	73.33	0.1392	6.9620		A->A, B->B
600.00	420.00	560.00	73.33	106.67	33.33	106.67	0.2025	10.1266		A->A, B->C
520.00	500.00	560.00	6.67	26.67	33.33	33.33	0.0633	3.1646	*	A->C, B->B
600.00	500.00	480.00	73.33	26.67	46.67	73.33	0.1392	6.9620		A->B, B->A
520.00	500.00	560.00	6.67	26.67	33.33	33.33	0.0633	3.1646		A->B, B->C
600.00	420.00	560.00	73.33	106.67	33.33	106.67	0.2025	10.1266		A->C, B->A
ave =	526.67									

520.00	500.00	560.00								
Phase A	Phase B	Phase C	Dev A	Dev B	Dev C	Max Dev		Selection		
570.00	500.00	560.00	26.67	43.33	16.67	43.33	0.0798	3.9877	*	A->A
520.00	500.00	610.00	23.33	43.33	66.67	66.67	0.1227	6.1350		A->C
ave =	543.33									

570.00	500.00	560.00								
Phase A	Phase B	Phase C	Dev A	Dev B	Dev C	Max Dev		Selection		
590.00	500.00	560.00	40.00	50.00	10.00	50.00	0.0909	4.5455		A->A
570.00	520.00	560.00	20.00	30.00	10.00	30.00	0.0545	2.7273	*	A->B
ave =	550.00									

570.00	520.00	560.00								
Phase A	Phase B	Phase C	Dev A	Dev B	Dev C	Max Dev		Selection		
590.00	520.00	560.00	33.33	36.67	3.33	36.67	0.0659	3.2934		A->A
570.00	540.00	560.00	13.33	16.67	3.33	16.67	0.0299	1.4970	*	A->B
ave =	556.67									

Appendix C

Phase Balancing Design Reports

C.1 AP&L Hot Springs North

```
Virginia Tech's Phase Balancing
-------------------------------

Start of circuit information:

Local Name:  Y423.
The Start of Circuit Position:  X=1331913   Y=2372822



System loss information

Before Phase Balancing
     Total kW = 7420.53
     kW Load = 6822.30
     kW Losses = 598.24

After Phase Balancing

NOTE:  THE SAVE THE CIRCUIT OPTION IS NOT AVAILABLE IN
DEWORKSTATION VERSION 1.0.  IN ADDITION, CIRCUIT LOSSES
ARE NOT COMPUTED FOR THE REPHASED CIRCUIT
Imbalance Information at the start of circuit

Before Phase Balancing
Imbalance =    0.17 (   8.57 %)
     At Time:
          Month: January
          Day:   Weekday
          Hour:  12am

After Phase Balancing
Imbalance =    0.05 (   2.25 %)
     At Time:
          Month: January
          Day:   Weekday
          Hour:  12am
```

Maximum Imbalanced Component BEFORE Phase Balancing

The component is a 3-Phase Line
Local Name: 1976.3779.
The Component Position: X=1331863 Y=2372822

Imbalance = 0.17 (8.57 %)
At Time:
Month: January
Day: Weekday
Hour: 12am

Maximum Imbalanced Component AFTER Phase Balancing

The component is a 3-Phase Line
Local Name: 1976.3779.
The Component Position: X=1331863 Y=2372822

Imbalance = 0.05 (2.25 %)
At Time:
Month: January
Day: Weekday
Hour: 12am

Lateral Reconnections

The component at the start of the lateral is a: 2-Phase Line
Local Name: 1964.3757.
Lateral is number 1 of 15
Lateral position is: X=1328293 Y=2367422

The Phase Movements are:
Phase A moved to Phase B
Phase B moved to Phase A
Phase C was not present

The component at the start of the lateral is a: 2-Phase Line
Local Name: 1970.3759.
Lateral is number 2 of 15
Lateral position is: X=1330106 Y=2367889

The Phase Movements are:
Phase A moved to Phase B
Phase B was not present
Phase C did not move

The component at the start of the lateral is a: 1-Phase Line
Local Name: 1998.3768.
Lateral is number 3 of 15
Lateral position is: X=1338563   Y=2369987

The Phase Movements are:
Phase A was not present
Phase B moved to Phase A
Phase C was not present

The component at the start of the lateral is a: 1-Phase Line
Local Name: 1993.3765.
Lateral is number 4 of 15
Lateral position is: X=1337050   Y=2369274

The Phase Movements are:
Phase A was not present
Phase B did not move
Phase C was not present

The component at the start of the lateral is a: 1-Phase Line
Local Name: 2000.3783.
Lateral is number 5 of 15
Lateral position is: X=1339202   Y=2373621

The Phase Movements are:
Phase A was not present
Phase B moved to Phase C
Phase C was not present

The component at the start of the lateral is a: 1-Phase Line
Local Name: 1996.3791.
Lateral is number 6 of 15
Lateral position is: X=1338016   Y=2375573

The Phase Movements are:
Phase A was not present
Phase B moved to Phase C
Phase C was not present

The component at the start of the lateral is a: 1-Phase Line
Local Name: 1992.3762.
Lateral is number 7 of 15
Lateral position is: X=1336741   Y=2368550

The Phase Movements are:
Phase A was not present
Phase B was not present
Phase C moved to Phase A

```
The component at the start of the lateral is a:  1-Phase Line
Local Name:  2032.3767.
Lateral is number 8 of 15
Lateral position is:  X=1348804   Y=2369645

The Phase Movements are:
Phase A was not present
Phase B moved to Phase A
Phase C was not present


The component at the start of the lateral is a:  2-Phase Line
Local Name:  2048.3761.
Lateral is number 9 of 15
Lateral position is:  X=1353610   Y=2368143

The Phase Movements are:
Phase A moved to Phase C
Phase B was not present
Phase C moved to Phase A


The component at the start of the lateral is a:  1-Phase Line
Local Name:  2100.3806.
Lateral is number 10 of 15
Lateral position is:  X=1369374   Y=2378918

The Phase Movements are:
Phase A did not move
Phase B was not present
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  2053.3769.
Lateral is number 11 of 15
Lateral position is:  X=1355135   Y=2370070

The Phase Movements are:
Phase A was not present
Phase B was not present
Phase C did not move


The component at the start of the lateral is a:  2-Phase Line
Local Name:  2029.3738.
Lateral is number 12 of 15
Lateral position is:  X=1347833   Y=2362618

The Phase Movements are:
Phase A did not move
Phase B moved to Phase C
Phase C was not present
```

```
The component at the start of the lateral is a:  1-Phase Line
Local Name:  2045.3745.
Lateral is number 13 of 15
Lateral position is:  X=1352670   Y=2364270

The Phase Movements are:
Phase A moved to Phase C
Phase B was not present
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  2024.3735.
Lateral is number 14 of 15
Lateral position is:  X=1346319   Y=2361904

The Phase Movements are:
Phase A was not present
Phase B moved to Phase A
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  2045.3749.
Lateral is number 15 of 15
Lateral position is:  X=1352579   Y=2365241

The Phase Movements are:
Phase A moved to Phase C
Phase B was not present
Phase C was not present
```

C.2 AP&L Hot Springs Carpenter

```
Virginia Tech's Phase Balancing
-------------------------------

Start of circuit information:

Local Name:  Z405.
The Start of Circuit Position:  X=1331334   Y=2345360



System loss information

Before Phase Balancing
     Total kW = 12773.35
     kW Load = 12246.35
     kW Losses = 527.00

After Phase Balancing

NOTE:  THE SAVE THE CIRCUIT OPTION IS NOT AVAILABLE IN
DEWORKSTATION VERSION 1.0.  IN ADDITION, CIRCUIT LOSSES
ARE NOT COMPUTED FOR THE REPHASED CIRCUIT
Imbalance Information at the start of circuit

Before Phase Balancing
Imbalance =    0.10 (   5.22 %)
     At Time:
          Month: January
          Day:   Weekday
          Hour:  12am

After Phase Balancing
Imbalance =    0.02 (   0.77 %)
     At Time:
          Month: January
          Day:   Weekday
          Hour:  12am

Maximum Imbalanced Component BEFORE Phase Balancing

The component is a 3-Phase Line
Local Name:  1975.3666.
The Component Position:  X=1331284   Y=2345360

Imbalance =    0.10 (   5.22 %)
     At Time:
          Month: January
          Day:   Weekday
          Hour:  12am
```

```
Maximum Imbalanced Component AFTER Phase Balancing

The component is a 3-Phase Line
Local Name:  1975.3666.
The Component Position:  X=1331284   Y=2345360

Imbalance =    0.02 (   0.77 %)
     At Time:
          Month: January
          Day:   Weekday
          Hour:  12am

Lateral Reconnections

The component at the start of the lateral is a:  2-Phase Line
Local Name:  1970.3645.
Lateral is number 1 of 15
Lateral position is:  X=1329824   Y=2340231

The Phase Movements are:
Phase A was not present
Phase B did not move
Phase C did not move


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1963.3652.
Lateral is number 2 of 15
Lateral position is:  X=1327731   Y=2341950

The Phase Movements are:
Phase A was not present
Phase B did not move
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1945.3645.
Lateral is number 3 of 15
Lateral position is:  X=1322285   Y=2340308

The Phase Movements are:
Phase A was not present
Phase B moved to Phase C
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1964.3631.
Lateral is number 4 of 15
Lateral position is:  X=1327980   Y=2336852

The Phase Movements are:
Phase A was not present
Phase B was not present
Phase C moved to Phase A
```

```
The component at the start of the lateral is a:  2-Phase Line
Local Name:  1941.3622.
Lateral is number 5 of 15
Lateral position is:  X=1321021   Y=2334741

The Phase Movements are:
Phase A moved to Phase C
Phase B was not present
Phase C moved to Phase A


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1925.3623.
Lateral is number 6 of 15
Lateral position is:  X=1316198   Y=2335034

The Phase Movements are:
Phase A was not present
Phase B was not present
Phase C moved to Phase A


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1916.3630.
Lateral is number 7 of 15
Lateral position is:  X=1313501   Y=2336761

The Phase Movements are:
Phase A moved to Phase C
Phase B was not present
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1928.3645.
Lateral is number 8 of 15
Lateral position is:  X=1317159   Y=2340362

The Phase Movements are:
Phase A was not present
Phase B was not present
Phase C moved to Phase B


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1923.3643.
Lateral is number 9 of 15
Lateral position is:  X=1315646   Y=2339893

The Phase Movements are:
Phase A was not present
Phase B was not present
Phase C did not move
```

```
The component at the start of the lateral is a:  2-Phase Line
Local Name:  1889.3624.
Lateral is number 10 of 15
Lateral position is:  X=1305342   Y=2335394

The Phase Movements are:
Phase A did not move
Phase B did not move
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1871.3619.
Lateral is number 11 of 15
Lateral position is:  X=1299900   Y=2334241

The Phase Movements are:
Phase A was not present
Phase B moved to Phase A
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1856.3564.
Lateral is number 12 of 15
Lateral position is:  X=1295225   Y=2320948

The Phase Movements are:
Phase A moved to Phase B
Phase B was not present
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1969.3680.
Lateral is number 13 of 15
Lateral position is:  X=1329609   Y=2348725

The Phase Movements are:
Phase A moved to Phase B
Phase B was not present
Phase C was not present


The component at the start of the lateral is a:  1-Phase Line
Local Name:  1978.3705.
Lateral is number 14 of 15
Lateral position is:  X=1332384   Y=2354763

The Phase Movements are:
Phase A was not present
Phase B moved to Phase A
Phase C was not present
```

```
The component at the start of the lateral is a:  1-Phase Line
Local Name:  1976.3708.
Lateral is number 15 of 15
Lateral position is:  X=1331788   Y=2355497

The Phase Movements are:
Phase A was not present
Phase B was not present
Phase C did not move
```

Appendix D

Transmission Line Losses

D.1 Introduction

For this discussion, only real power will be considered. However, the reactive component can readily be incorporated. To solve for transmission losses, the current magnitude in each branch, I_k, must be calculated, squared, and multiplied by the branch impedance, R_k. The losses in each branch are summed together to determine the total transmission losses, Pl, as shown in (D.1).

$$Pl = \Sigma\, I_k^{\,2} * R_k \tag{D.1}$$

When formulating the economic dispatch problem to include transmission line losses, a constant, c, must be used to convert the transmission line losses to dollars. Once the losses are in terms of dollars, the objective function is modified as shown in (D.2). Note that when including transmission line losses, the total generation, Pt, must equal both the total load, Pr, and the losses, Pl. The equality constraint equation is modified to include Pl as shown in (D.3).

$$\text{Minimize: } Ft(Pt) + c * Pl \tag{D.2}$$

$$\text{Subject to: } Pr + Pl - Pt = 0 \tag{D.3}$$

D.2 Example D.1: Transmission Line Losses

This example incorporates transmission line losses into the economic dispatch problem of Example 7.1. The new objective function and new constraints are shown below.

$$\text{Minimize } F_1(P_1) + F_2(P_2) + F_3(P_3) + c * Pl \tag{D.4}$$

Subject to:

$$P_1 + P_2 + P_3 = 850.0 + Pl \tag{D.5}$$

$$150 \le P_1 \le 850.0 \tag{D.6}$$

$$100 \le P_2 \le 600.0 \tag{D.7}$$

$$50 \le P_3 \le 400.0 \tag{D.8}$$

$$Ps_i \le 375.0, \quad i = 1, 7 \tag{D.9}$$

$$Pl = \Sigma\, I^2_k * R_k, \quad k = 1, 7 \tag{D.10}$$

The load increment, Lp, is chosen to be 25 MW as given in (7.17). To convert transmission line losses into dollars, conversion factor, c, in (D.11) is used.

$$c = 20.0 \text{ \$/MW - Hr} \tag{D.11}$$

Table D.1 organizes the DAOP solution by showing a few of the computational steps. The final solution is given in Table D.2.

Table D.1 DAOP Solution Table for Example D.1

Step	Generating Unit	Δ Cost $/Hr	Total Cost $/Hr	Selection	Generation Level MW
1	1	1,147.98	1,148.78		25
	2	1,146.46	1,147.26	<=	
	3	1,151.20	1,152.06		
2	1	198.98	1,347.04	<=	50
	2	199.98	1,350.36		
	3	202.26	1,350.32		
3	1	200.93	1,550.38		75
	2	199.89	1,549.34	<=	
	3	202.26	1,550.10		
⋮	⋮	⋮	⋮	⋮	⋮
33	1	228.26	8,353.42		825
	2	226.56	8,351.62	<=	
	3	232.39	8,372.29		
34	1	228.26	8,609.74	<=	850
	2	228.99	8,614.37		
	3	232.39	8,632.62		

Table D.2 DAOP Solution for Example D.1

Generating Unit	Output
P_1	350
P_2	350
P_3	150
Total Cost $/Hr	8,609.74

Vita

Paul A. Dolloff was born at Andrews Air Force Base in Camp Springs, Maryland on November 22, 1963. In 1982, Dr. Dolloff graduated from McGavock High School in Nashville, Tennessee. During High School, Dr. Dolloff continued his study of the violin and performed with a variety of professional groups. Accepting a full music scholarship, he earned the degree of Bachelor of Science in Electrical Engineering from Tennessee Technological University in June 1987. In January 1995 and January 1996 he earned the degrees of Masters of Science and Doctor of Philosophy in Electrical Engineering from Virginia Polytechnic Institute and State University in Blacksburg, Virginia. In addition, Dr. Dolloff has spent five summer semesters at the University of the South in Sewanee, Tennessee studying the violin and viola. Currently, Dr. Dolloff and his wife, Robin an Environmental Engineer with Springs Inc., have a small horse farm in Rock Hill, SC.