1

Introduction

In today's deregulated environment, the nation's electric power network is being forced to operate in a manner for which it was not intentionally designed. Therefore, system analysis is very important to predict and continually update the operating status of the network. This includes estimating the current power flows and bus voltages (power flow analysis and state estimation), determining the stability limits of the system (continuation power flow, numerical integration for transient stability, and eigenvalue analysis), and minimizing costs (optimal power flow). This book provides an introductory study of the various computational methods that form the basis of many analytical studies in power systems and other engineering and science fields. This book provides the analytical background of the algorithms used in numerous commercial packages. By understanding the theory behind many of the algorithms, the reader/user can better use the software and make more informed decisions (i.e., choice of integration method and step size in simulation packages).

Due to the sheer size of the power grid, hand-based calculations are nearly impossible, and computers offer the only truly viable means for system analysis. The power industry is one of the largest users of computer technology and one of the first industries to embrace the potential of computer analysis when mainframes first became available. Although the first algorithms for power system analysis were developed in the 1940s, it wasn't until the 1960s that computer usage became widespread within the power industry. Many of the analytical techniques and algorithms used today for the simulation and analysis of large systems were originally developed for power system applications.

As power systems increasingly operate under stressed conditions, computer simulation will play a large role in control and security assessment. Commercial packages routinely fail or give erroneous results when used to simulate stressed systems. Understanding of the underlying numerical algorithms is imperative to correctly interpret the results of commercial packages. For example, will the system really exhibit the simulated behavior or is the simulation simply an artifact of a numerical inaccuracy? The educated user can make better judgments about how to compensate for numerical shortcomings in such packages, either by better choice of simulation parameters or by posing the problem in a more numerically tractable manner. This book will provide the background for a number of widely used numerical algorithms that underlie many commercial packages for power system analysis and design.

This book is intended to be used as a text in conjunction with a semester-long graduate level course in computational algorithms. While the majority of examples in this text are based on power system applications, the theory is presented in a general manner so as to be applicable to a wide range of engineering systems. Although some knowledge of power system engineering may be required to fully appreciate the subtleties of some of the illustrations, such knowledge is not a prerequisite for understanding the algorithms themselves. The text and examples are used to provide an introduction to a wide range of numerical methods without being an exhaustive reference. Many of the algorithms presented in this book have been the subject of numerous modifications and are still the object of on-going research. As this text is intended to provide a foundation, many of these new advances are not explicitly covered, but are rather given as references for the interested reader. The examples in this text are intended to be simple and thorough enough to be reproduced easily. Most "real world" problems are much larger in size and scope, but the methodologies presented in this text should sufficiently prepare the reader to cope with any difficulties he/she may encounter.

Most of the examples in this text were produced using code written in MATLAB®. Although this was the platform used by the author, in practice, any computer language may be used for implementation. There is no practical reason for a preference for any particular platform or language.