# DYNAMIC STABILITY ANALYSIS OF LARGE-SCALE POWER SYSTEMS USING A FAST INVESTIGATION METHOD

by

MING-TSE KUO

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2006

ACKNOWLEDGEMENTS

Sincere appreciation is expressed to Dr. Wei-Jen Lee, Dr. Raymond R. Shoults, Dr. William E. Dillon, Dr. Kai Yeung, and Dr. Jay Rosenberger for their guidance as members of the author's graduate committee.

The author would like to express particular gratitude to his supervising professor, Dr. Wei-Jen Lee for his guidance and unceasing encouragement. In spite of his busy schedule, Dr. Lee finds time to help and support his students. It was indeed a pleasure and an honor to have worked with him. His dedication to power systems education has to be greatly admired.

The author would also want to express appreciation to all members of Energy Systems Research Center (ESRC), the University of Texas at Arlington, for all their valuable assistances, discussions, and enjoyable association. Their careful review, helpful suggestions and discussions in this dissertation are sincerely appreciated.

The author's parents and sister were the primary source of inspiration during his education. I am very grateful for their love, encouragement and continuous support.

November 19, 2006

ABSTRACT

DYNAMIC STABILITY ANALYSIS OF LARGE-SCALE POWER SYSTEMS USING A FAST INVESTIGATION METHOD

Publication No. _____

Ming-Tse Kuo, Ph.D.

The University of Texas at Arlington, 2006

Supervising Professor: Wei-Jen Lee

Following power market deregulation and the proposed standard market design, the activities of cross boundary power transactions are increased significantly. Traditional company based stability analysis can not satisfy today's operating environment. However, simulating the entire interconnection power system may exceed the capacity of existing commercial stability analysis software. Fortunately, most of the stability problems are started from the local level. One will be able to avoid most of the system wide stability issues by mitigating the problems in the early stage. Based on

eigenvector decompositions of the Jacobian, this dissertation proposes a new algorithm for performing the dynamic stability problem for a large interconnected power system. Using Python and MATLAB to build an integrated simulation environment around PSS/E not only overcomes the large-scale power system problem but also shortens the simulation time. The proposed method has been verified on the Southwest Power Pool (SPP) system, which has close to 30,000 buses. The simulation results on both the proposed approach and the entire system were almost identical.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

## 1.1 Background of Power Market Deregulation and Emerging Problems

Since an increase in the number of commodities in the market improved the operation efficiency through free competition, the price of goods was further decreased. Some traditional government owned utilities such as telecommunication and transportation began to be opened to private enterprises. The change in the power system from the traditional vertically integrated utility to the new competitive structure was considered. Almost all the Electricity Supply Industries (ESI) in the world are currently in the process of being restructured to a competitive environment. In the United States, the Energy Policy Act of 1992 was originally formed to implement the idea of deregulation of power industry. In 1996, the Federal Energy Regulatory Commission (FERC) issued the Order No. 888 to initiate the processes of deregulation by requiring all public utilities to provide open access to transmission facilities. In order No. 888, FERC also recommended establishing Independent System Operators (ISOs) to monitor the reliability of the power system and coordinate the supply of electricity in each region. Order No. 889 initiated the Open Access Same-Time Information System (OASIS) and Standards of Conduct, which required utilities to open information about their available transmission capacity, price, and access to transmission services non-

discriminatorily. This ensures that transmission owners do not have an unfair advantage over other transmission users.

Additionally, the historical structure of power system whereby a vertically integrated utility that owned generating plants, high voltage transmission system, and distribution lines as well as provided all electric services was required to be changed. Obligatorily, all vertically integrated utilities in the restructured regions needed to separate their assets and services into generation, delivery services including transmission and distribution, and retail sales. Therefore, some regions are managed by an Independent System Operator (ISO) or a Regional Transmission Organization (RTO) which was established by FERC in Order No. 2000. The main roles of the ISOs and RTOs are to perform transmission planning, ensure wholesale power grid reliability as well as equal access to the grid, and economically balance electricity demand and supply.

Under the ISO structure, it is clear that the power industry has become more competitive and more regionalized, and all power system facilities will be compelled to operate closer to their limits. As a result, the security level of the system is declining and therefore electrical engineers must frequently analyze the power system stability to prevent a system collapse.

In 2002, after forming the regional ISOs, FERC further issued a notice of the proposed rulemaking on standard market design (SMD) which established spot markets for energy, operating reserves, and transmission service as a bid-based, security-constrained economic dispatch with localtional marginal prices (LMPs) [1]. After the

implementation of the SMD, it can be expected that ISOs will become more competitive and that there will be increased sharing of electricity among them since a unified rule will be used for all market operations. An analysis of the power system will not be confined to a regional system but will involve the systematic analysis of all the interconnected systems.

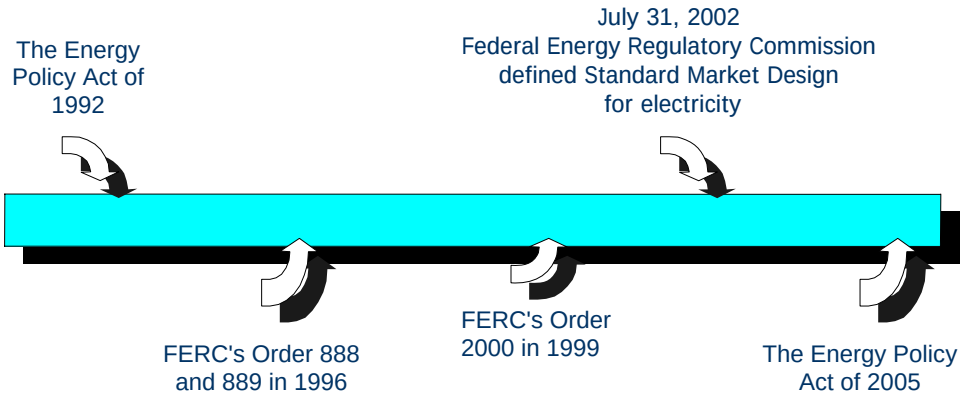



Figure 1.1 Milestones of the power market deregulation in US

The changes taking place in the electric industry have altered many of the traditional mechanisms, incentives, and responsibilities for maintaining reliability to the point that the voluntary system of compliance with reliability standards is no longer adequate. In response to these changes, NERC has promoted the development of a new mandatory system of reliability standards. On August 8, 2005, President Bush signed into law the Energy Policy Act of 2005, which authorizes the creation of an electric reliability organization (ERO) with the statutory authority to enforce compliance with reliability standards among all market participants. [42] This is a significant change for maintaining reliability from voluntary to responsibilities. The effective maintenance of

the national system is a vital subject. The history of the power market deregulation in the US is shown in Figure 1.1.

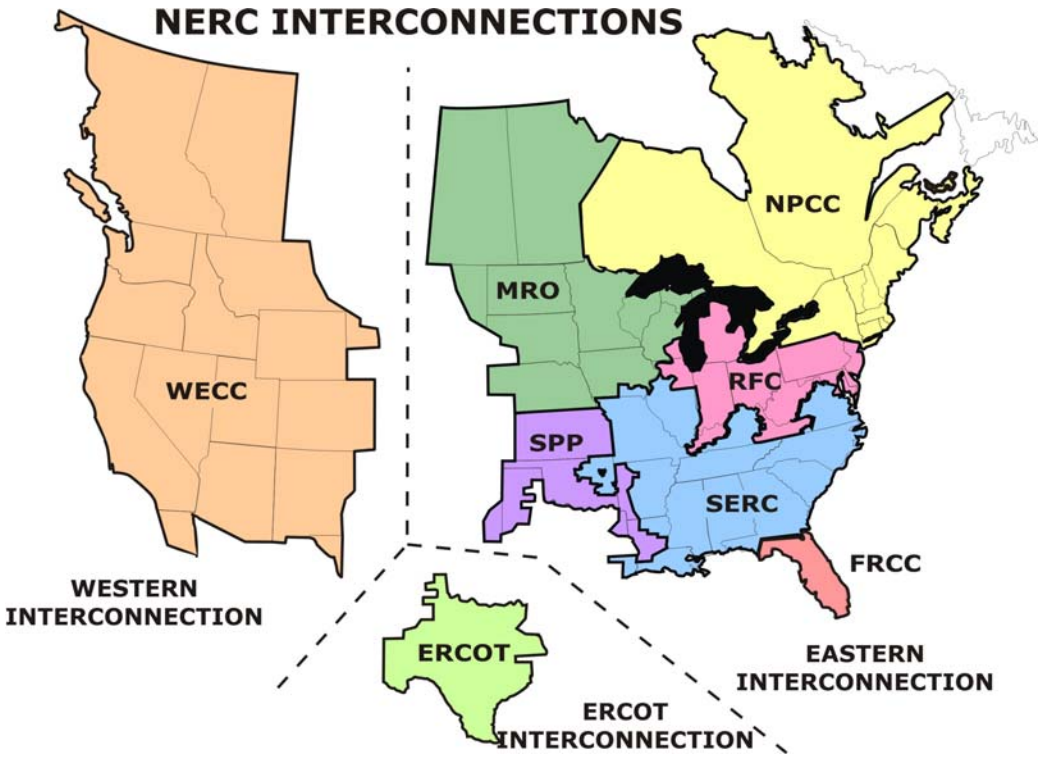



Figure 1.2 North American Electric Reliability Council (NERC) in 2006 [2]

As of November 2006, there are eight Regional Reliability Councils in North America, as shown in Figure 1.2. It is clear that the western systems have been integrated; in fact, the boundary within the eastern systems is disappearing rapidly, resulting in a rapid change in the entire power flow. Due to its unpredictability for power flow, the traditional off-line stability simulation is not suitable for the present requirement. It is necessary to establish the capability of near real-time simulation. Current approaches and algorithms can not support this requirement. Therefore, the development of tools and/or algorithms for the speedy investigation of a large-scale power system is an urgent requirement.

## 1.2 Chronology of Major Events in Power System Stability in US

Transient stability is the ability of a power system to maintain synchronism when subjected to a sudden change in generations, loads, or system characteristics. If this large disturbance creates an unbalance between the supply and demand, a new operating state is reached. The system is stable if the oscillations after the system perturbation are damped toward a new quiescent operating condition keeping in synchronism within interconnected system. [6]

In traditional power system analysis, transient stability is a complex subject that has challenged electrical engineers for many years. The stability of power systems was first recognized as an important issue in 1920. Results of the first laboratory tests on miniature systems were reported in 1924; the first field tests of the stability on a practical power system were conducted in 1925. Early stability problems were identified when remote hydroelectric generating stations were feeding into metropolitan load centers over long distance transmission networks. The analysis algorithm and the models used were dictated by developments in the art of computation and the stability theory of dynamic systems. The theoretical work carried out in the 1920s and early 1930s laid the foundation for the industry's basic understanding of the power system stability phenomena. In the early 1950s, electronic analog computers were used for analyzing of special problems requiring detailed modeling of the synchronous machine, excitation system, and speed governor. When engineers understood more stability phenomena, power systems are accordingly huger and more complicated in the same

time. In the 1960s, most of the power systems in the United States and Canada were joined as part of one of the three large interconnected systems. [3]

While interconnections result in operating economy and increase reliability through mutual assistance, they have also increased complexity of stability problems. A series of blackouts, as shown in Figure 1.3, once again brought into focus the problem of stability. The Northeast Blackout of 1965 was a significant event to make people aware of the importance of power system stability. [4] Although numerous research results were reported, they were still unable to prevent a recurrence and the blackout of 1977 affected New York City again. After deregulation, the scale of blackout is easier to expand because the power flow among each area is increased. The Northeast Blackout of 2003, the largest in North American history, not only resulted in a loss of six billion USD but also affected an estimated ten million people in the Canadian province of Ontario and 40 million people in eight U.S. states [5].

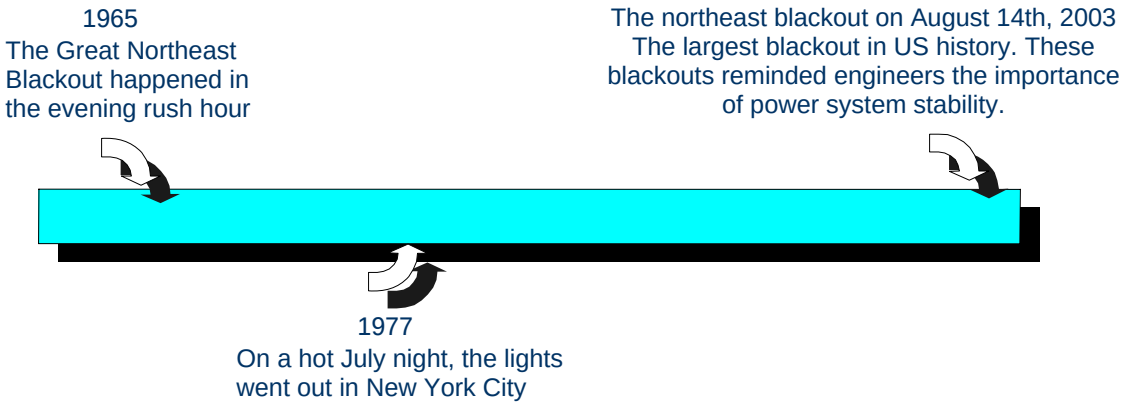



Figure 1.3 Some large blackouts in North American history

This blackout event was triggered by several local faults. From the system record, the system failures started from 1:31 PM. Following by several other incidents,

it has developed into an uncontrollable cascade after 3:46 PM. If a real-time tool can detect the possibility of a system blackout and dispatch a correction measure right after the initial failures, the blackout could have been prevented.

## 1.3 Objectives and Approaches

In the past, to address the problem of stability analysis in a large-scale power system, two types of strategies were proposed: applying a parallel computing algorithm to accelerate the speed of simulation or developing dynamic equivalency techniques to simplify the circuit model.

Parallel software for cluster systems on high-performance computers has been widely used in every field. Moreover, research on parallel algorithms and their application in transient stability analysis has been well developed over the last 16 years [7]. The focus of this development lies in how to divide a large power system grid into several smaller blocks whose data are sent to different computers and computed separately at the same time. Thus, the computational load of each block must be the same, while the system partitioning and communication between blocks must be minimal.

Although parallel computing can accelerate the speed of simulation, there are some disadvantages. To apply the parallel software for cluster systems in dynamic stability is a complex procedure both in software development and hardware architecture. Therefore, it is difficult to apply parallel computing widely in power system simulation [8]. On the other hand, although the parallel computing algorithm will attempt to assign the same computational load to each computer, this goal is

difficult to achieve because of the asymmetric structure of the power system and generator aggregation; therefore, users need to wait for the simulation result of the last block.

Dynamic equivalent approach is based on the observation that oscillations in local areas seldom affect other areas for a traditional monopolistic electric power system. Therefore, one will be able to obtain reasonable simulation results by keeping full circuit model of the local power system while using dynamic equivalents for the system outside the local area to reduce the computational burden. Constructing a dynamic equivalent for a power system requires partitioning the system into coherent areas, aggregating the coherent areas, and aggregating the coherent generators and their control devices [9].

For dynamic equivalency approaches, although the simplified equivalent circuit can speed up the simulation, it faces some challenges after deregulation. For example, there were 10 areas in the NERC in 2005, as shown in Figure 1.4; however, there were only 8 areas left in 2006, as shown in Figure 1.2. Because the boundaries of local areas disappear quickly after power market deregulation, the area being studied may have to cover multiple local boundaries. Especially after SMD, the cross boundary transactions will increase because the electricity price can be calculated in a unified method. Hence, there is a need to develop a more appropriate algorithm to define the local area. On the other hand, if faults occur near the boundary, the simulation result won't be not accurate because of the nearby equivalent circuit instead of the original circuit. Moreover, if

multi-faults occur in different areas, it is difficult to handle with both parallel computing and dynamic equivalency.

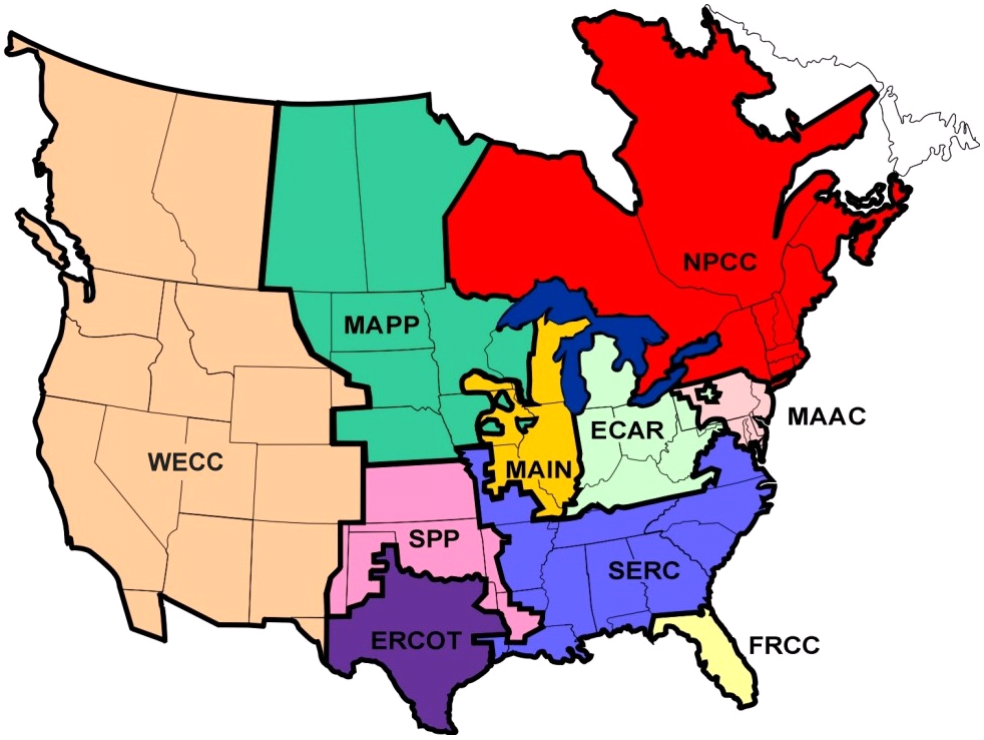



Figure 1.4 North American Electric Reliability Council (NERC) in 2005 [10]

Because parallel computing involves the simulation of entire systems in order to provide a complete solution, the extent to which its simulation time can be reduced is limited. For dynamic equivalency, new equivalent circuit must be established for the simulation of each different area. After deregulation, the entire power system pattern will rapidly change; therefore, the abovementioned two methods are insufficient and a new method must be developed. This new method should be able to rapidly analyze the changing systems and has the capability to evaluate the power system stability near real time. Except implement the above-mentioned objectives, this dissertation also include multi-faults simulation in order to simulate the blackout situation. With the fast

response of the simulation, it should be possible to promptly determine the likelihood of a blackout in order to predict its onset and prevent a chain reaction.

The basic concept to attain the goal is that each fault initially has only a local impact. Its influence is limited in range. If multi-faults occur in the system, a new boundary will be chosen according to their ranges and therefore it is not necessary to simulate the entire system; thus, the simulation result is sufficiently accurate to analyze a blackout. The essence of the proposed algorithm is the application of graph theory for partitioning the power system. Further, clearly define and aggregate a local area from a large-scale power system for dynamic stability simulation. Finally, develop a software workbench that provides a graphic user interface by integrating the commercial stability software and the proposed algorithm. The ability of the proposed method to fast analyze multi-faults transient stability in a large power system is verified by studying the Southwest Power Pool system.

## 1.4 Synopsis of Chapters

The material in this dissertation is organized as follows:

Chapter 1 introduces the general background of the power market deregulation, chronology of events in power system stability, and illustrates the importance, motivation and objective of this dissertation.

Chapter 2 introduces the history and current development of the spectral graph theory. The definitions and theorems that link the relationships among graphs, matrices and algebra are stated. There are two main methods of spectral partitioning, median cut spectral bisection and spectral graph partitioning. Two methods would be discussed in

detail and compared the difference. The last part will explain why spectral graph partitioning is suitable to apply in power system.

Chapter 3 describes the proposed partitioning algorithm in power system. Because of different power system structures, the power flow may be solved by experience. To understand the relationship between the Jacobian matrix in power flow and the Laplacian matrix is the key to apply graph partitioning in power system. After forming pre-partitioning sub-areas, the sensitivity areas will aggregate parts of sub-areas to build new sub-systems which will which will simulate the transient stability in PSS/E.

Chapter 4 tests the proposed method in the Thailand and SPP systems. The main objective of this chapter is to examine whether the proposed algorithm can perform properly in contemporary systems included data transform between IEEE and PSS/E format, partitioning algorithm in graph theory, setting sensitivity areas and selection of final areas. To test different cases in single fault and multi-faults evaluated the algorithm performance.

Chapter 5 simply explains all programming codes which have been integrated with the graphical user interface in the Python program and gives a manual for the developed software. At the last part, user can use the double check boundary method to double check the results.

Chapter 6 states the summary and conclusion of this dissertation and discusses the opportunity for further research.

# CHAPTER 2

# SPECTRAL GRAPH THEORY

## 2.1 Introduction

Spectral graph theory has a long history in mathematics. It introduces eigenvalues and their use in finding a clue to the secret of the graph. In the beginning, spectral graph theory utilized the matrix theory and linear algebra to analyze the adjacency matrices. Then, the study has been expanded to include graphical characters and the relationship between characteristic polynomial and adjacency matrices or the relation between the Laplacian matrix and eigenvalues or eigenvectors. Because a non-directed finite graph has a symmetrical adjacency matrix, it has the real numbers of eigenvalues and completely orthogonal eigenvectors. Because of its regular and symmetrical, the algebraic method works very efficiently with the graph. Furthermore, after Fiedler found that some eigenvalues are related to the algebraic connectivity of a graph, many algebraic views about spectral graph theory emerged in large numbers. Thereafter, guiding the geometric views into spectral graph theory allowed its development to flourish. [13]

Graph partition is a part of spectral graph theory and has extensive applications. In the graph theory, the main research is a graphical partitioning based upon the eigenvalues and eigenvectors of the Laplacian matrix of a graph. Because no unified

method describes a graph, the Laplacian matrix will be different as the node labeling differs, although its spectra are constant in the graph, so that its sub-graphs will be the same after decomposition.

This dissertation has applied the "edge-cut of the graph" to divide a graph into two unconnected parts by a set of edges. The standard minimum cut algorithm will be used to minimize the edge-cut. Although it may cause an unbalanced separation if, for instance, it cuts a part of a graph which only includes a few nodes, the recursive method will be used to overcome this deficiency and reach the subdivided goal.

Because the development of spectral graph theory is very fast and the ranges of spectral graph theory and graph partition are very extensive, this chapter will only introduce some parts that correlate with this dissertation enough for the reader to understand the following chapters. This chapter focuses on how spectrum represents the fundamental property of a graph and how to utilize the skills of geometry and algebra to change a graph into a mathematical model. The next chapter will discuss in detail how to apply spectral graph theory to a power system via eigenvalues and how to make both come together in symbiotic harmony.

## 2.2 Basic Graph Theory

Spectral partitioning is a part of the graph theory. In the beginning, readers are required to be familiar with some definitions and properties of the graph theory.

A graph, G=(N,E), consists of a vertex set (N) and edge set (E), in the non-directional graph. The edge set is a group of non-order vertices. In this dissertation, it is assumed that all graphs are non-directional and finite.

Since each transmission line transmits a different amount of power, a power system will be mapped to a weighted graph. A weighted graph is a graph that has a value $w_i$ weighting in each vertex $v_i$, and a non-zero weight $w_{ij}$, multiplied by each edge $(v_i, v_j)$. If the edge does not exist, its weight is regarded as zero. If the graph is un-weighted, the graph can be regarded as having the weight of the special case of 1.

Because the power grids link to each other, the corresponding graph is connected. A graph G is connected if the graph has a path from any point to another point in the graph -- otherwise the graph is disconnected.

Suppose N={1,…,n}. Let G=(N,E) be an undirected, un-weighted graph without self edges (i,i) or multiple edges from one node to another. The most natural matrix with the graph G is its adjacency matrix $A_G$. The adjacency matrix A of G is a $|N| \times |N|$ matrix. The adjacency matrix of a simple graph is a matrix with rows and columns labeled by graph vertices, with a 1 or 0 in position according to whether they are adjacent or not. Its terms can be represented as follows [11]:

$$a_{i,j} = \begin{cases} 1 & \text{if } (i, j) \in E \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

From the above definition, the following conclusion can be deduced. For a simple graph with no self-loops, the adjacency matrix must have 0s on the diagonal. For an undirected graph, the adjacency matrix is symmetric.

## 2.3 Eigenvalues and the Laplacian of a Graph

The Laplacian matrix $\mathcal{L}(G)$ of a graph G is a $|N|\times|N|$ symmetric matrix, with one row and column for each node. It is defined as follows [12].

$$[\mathcal{L}(G)](i,j)=\begin{cases}-1 & \text{if i}\neq\text{j and } (i,j)\in E\\ d_i & \text{if } i=j, \text{and}\\ 0 & \text{otherwise,}\end{cases} \tag{2.2}$$

for i,j=1,…,N, where $d_i$ is the vertex degree of node i.

A normalized version of the Laplacian matrix, denoted $\mathcal{L}$, is similarly defined by [13].

$$[\mathcal{L}(G)](i,j)=\begin{cases}-\dfrac{1}{\sqrt{d_i d_j}} & \text{if i and j are adjacent}\\ 1 & \text{if } i=j, \text{and } d_i\neq 0\\ 0 & \text{otherwise,}\end{cases} \tag{2.3}$$

A simple example to explain the definition of the Laplacian matrix is shown in Figure 2.1.

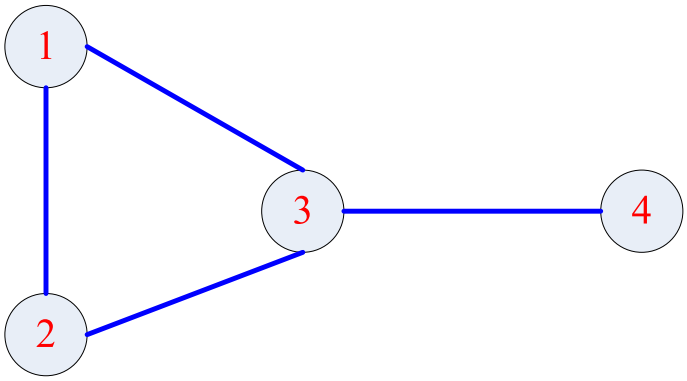



Figure 2.1 A simple graph of four nodes

This is a simple graph of four nodes. Its adjacency matrix is

$$A(G) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.4}$$

According to the definition, its Laplacian matrix is

$$\mathcal{L}(G) = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix} \tag{2.5}$$

Each edge will multiply by weight function w in the program. First, A(G) must be set up by the definition of the adjacency matrix. Then, the following equation is used to calculate $\mathcal{L}$ in order to simplify the programming [14].

$$\mathcal{L} = D - A = diag[sum(A)] - A \tag{2.6}$$

Where D is a $|N| \times |N|$ diagonal matrix. Its diagonal terms are

$$D(i,i) = d(i) = \sum_{i \sim j} w(i,j) \ , \ \forall i \in N \tag{2.7}$$

Where, i~j means that i and j are adjacent and d(i) denotes the degree of the vetex i. The degree represents how many lines connect to this vertex.

Some examples are provided to observe eigenvalues and eigenvectors of $A(G)$ and $\mathcal{L}(G)$ . Although there are no obvious relationships among eigenvalues, eigenvectors, and the graph G, readers can understand some basic characters by following these simple examples.

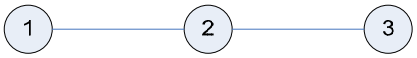



Figure 2.2 The simplest graph of three nodes

At the beginning, observe the simplest graph of three nodes as shown in figure 2.2. The vertex set of this graph is N={1,2,3} and edge set is E={(1,2),(2,3)}. Its eigenvalue and eigenvector are shown in table 2.1.

Table 2.1 Eigenvalues and eigenvectors of three node graph

| Eigenvalues | Corresponding eigenvector |
|---|---|
| 0 | (1,1,1) |
| 1 | (-1,0,1) |
| 3 | (1,-2,1) |

Through observing spectra of this graph, its minimum eigenvalue is 0 and the corresponding eigenvector is the all 1s vector. Other eigenvalues are the non-negative real number. In fact, all of Laplacian matrices have similar characteristics. In the geometric view, every eigenvector is a function that will be correspondent to vertices. In the other words, it appoints a real number in every corresponding vertex and this corresponding relation can further bring to partition a graph.

While partitioning, a graph G=(N,E) can be separated by two methods. First method is to cut part of subset Es from edge set E in order to partition graph G into two disconnected sub-graphs $G_1$ and $G_2$, with nodes $N_1$ and $N_2$ respectively in which $N_1 \neq \phi \neq N_2$, $N_1 \bigcup N_2 = N$ and $N_1 \bigcap N_2 = \phi$. The edges in Es connect nodes in $N_1$ to nodes in $N_2$. Since removing Es disconnects G, Es is called an edge separator. The

other partitioning method is to look for a vertex separator. A vertex separator is a subset Ns in vertices N. Cutting Ns and its connected edges from a graph also yields two disconnected sub-graphs $G_1$ and $G_2$. In other words, $N = N_1 \bigcup N_s \bigcup N_2$ and all three subsets of N are disjoint. The idea is illustrated in figure 2.3. Green and blue branches are the edge separator. Red nodes are the vertex separator. Cutting red nodes and their connected edges will obtain two disjointed sub-graphs.

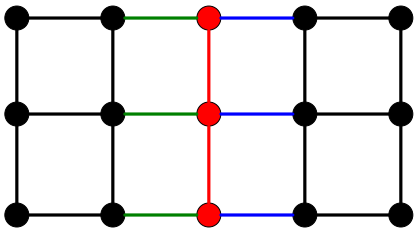


Figure 2.3 Edge separators and vertex separators

In power systems, buses will be represented as nodes in the graph and transmission lines will be treated as edges. Therefore, partitioning a system will be transferred into cutting a graph. Since cut nodes and edges at the same time will have significant impact on the system integrity, a method only to cut edges will be suitable for systematic partitioning for a power system.

## 2.4 Spectral Partitioning

Spectral partitioning started from Fiedler's research on 1970s. After SIAM journal on matrix analysis and applications published "Partitioning sparse matrices with eigenvectors of graphs" by A. Pothen, H. Simon, and K.–P. Liou in 1990, this technique is popular with applications. [12]

The spectral partition mainly studies the second smallest eigenvalue $\lambda_2[\mathcal{L}(G)]$ of a graph. This smallest non-zero eigenvalue is also called the Fiedler value, and the corresponding eigenvector called Fiedler vector, used for revealing the contribution of Fiedler's research. Because Fiedler found for a connected graph, its minimum eigenvalue is 0 and other eigenvalues are positive real numbers, this graph is a connected graph if Fiedler value of this graph is not 0. This property can be defined as follows:

Definition: (M. Fiedler, "Algebraic Connectivity of Graphs", Czech. Math. J. 23:298-305,1973). $\lambda_2[\mathcal{L}(G)]$ is called the algebraic connectivity of G [15].

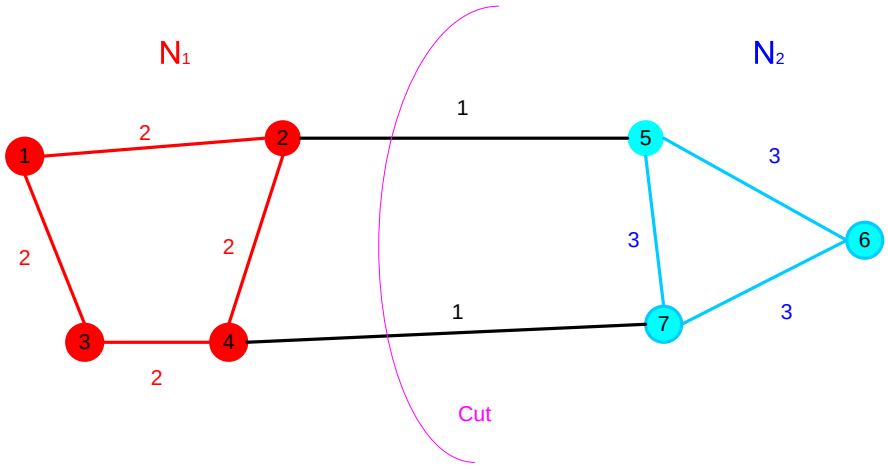
As shown in Table 2.1, the Fiedler value of the graph with three nodes is 1 and its Fiedler vector is (-1,0,1). Fiedler value represents whether a connected graph can be cut and Fiedler vector represents how to divide vertices into two groups, $N_1$ and $N_2$. It is to find the minimum edges have been cut during the partition in most cases. Let

$$Cut(N_1, N_2) = E(N_1, N_2) = \sum_{i \in N_1, j \in N_2} w_{ij} \tag{2.8}$$

It represented a group of edges whose vertices lie on opposite sides of the cut. Figure 2.4 is used to explain the definition of edge cut. Two kinds of commonly used edge cut methods, spectral graph partitioning and median cut spectral bisection, will be applied.

*2.4.1. Spectral Graph Partitioning*

It is clear in the cut definition. The decomposition $N = (N_1, N_2)$ may not be uniquely determined by the cut C. However, it is easy to prove the following assertion:




$$Cut(N_1, N_2) = 2$$

Figure 2.4 Example of edge cut

Corollary 1: Let C be a cut in a graph G. If there is a decomposition $N = (N_1, N_2)$ of the vertex set N of G corresponding to C such that both corresponding banks are connected then the decomposition of N corresponding to C is unique.

Corollary 2: Let G be a valuated connected graph with vertices 1,2,…,n, let $y = (y_i)$ be a characteristic valuation of G. If c is a number such that $0 \le c < \max y_i$ and $c \ne y_i$ for all i then the set of all those edges (i,k) of G for which $y_i < c < y_k$ forms a cut C of G. If $N_1 = \{k \in N \mid y_k > c\}$ and $N_2 = \{k \in N \mid y_k < c\}$ then $N = (N_1, N_2)$ is a decomposition of N corresponding to C and the bank $G(N_2)$ is connected.

Corollary 3: Let G be a valuated connected graph with vertices 1,2,…,n, let $y = (y_i)$ be a characteristic valuation.

If $y_i \ne 0$ for all $i \in N$ the set of all alternating edges, i.e. edges (i,k) for which $y_i y_k < 0$, forms a cut C of G such that both banks of G are connected.

By corollary 1, the decomposition and the banks are uniquely determined in this case, if

$$N_1 = \{i \in N \mid y_i > 0\} \text{ and } N_2 = \{i \in N \mid y_i < 0\} \tag{2.9}$$

Then $N = (N_1, N_2)$ is the decomposition corresponding to C [16].

Based on the above theory, both $N_1$ and $N_2$ are connected if Fiedler vector $y_i$ is divided into two groups greater or smaller than 0 in the equation (2.9), $N_1$ and $N_2$. The procedure of decomposition and the relationship with an optimal minimum cut will be stated as follows:

First, list the optimization function.

$$\begin{aligned} &\text{Objective}: \text{Min } Cut(N_1, N_2) \\ &\text{No constraint} \end{aligned} \tag{2.10}$$

In order to find the minimum cut, first let a bi-partition $(N_1, N_2)$ as a vector

$$p_i = \begin{cases} +1 \text{ if } n_i \in \mathrm{N}_1 \\ -1 \text{ if } n_i \in \mathrm{N}_2 \end{cases} \tag{2.11}$$

We can minimize the cut of the partition by finding a non-trivial vector p that minimizes the function.

$$f(p) = \sum_{i,j \in N} w_{ij} (p_i - p_j)^2 = p^T \mathcal{L}\, p \tag{2.12}$$

The Rayleigh theorem shows the minimum value for f(p) is given by the second smallest eigenvalue of the Laplacian $\mathcal{L}$ and the optimal solution for p is given by the corresponding eigenvector $\lambda_2$, referred as the Fiedler vector [17].

Thus, the proposed cutting procedure is optimized and guarantees that the partitioning result will be connected. Since an islanding situation will not occur if the subsystems are connected, this is a desired situation in the power flow analysis. This method will be adopted and the flow chart is shown in Figure 2.5.

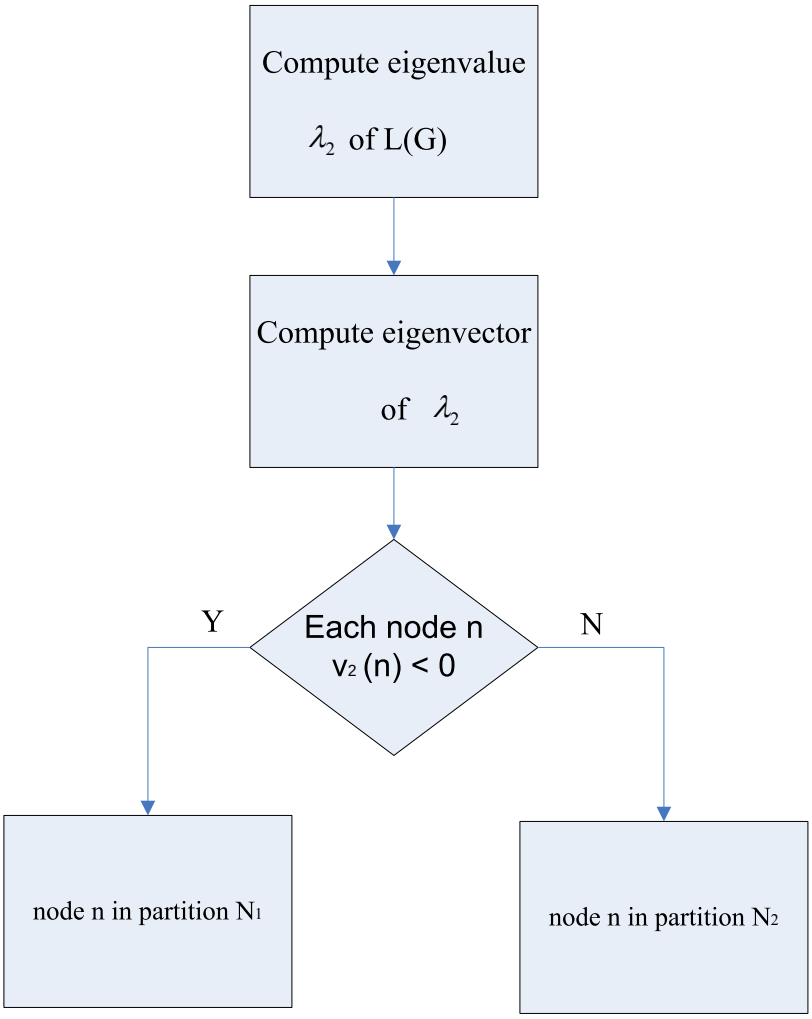



Figure 2.5 Flowchart of Spectral bisection of a graph

Finally, figure 2.4 is used as an example to explain the whole partitioning process.

Step 1. Pre-processing: Build Laplacian matrix $\mathcal{L}$ of the graph.

$$\mathcal{L}(G)=\begin{bmatrix} 4 & -2 & -2 & 0 & 0 & 0 & 0 \\ -2 & 5 & 0 & -2 & -1 & 0 & 0 \\ -2 & 0 & 4 & -2 & 0 & 0 & 0 \\ 0 & -2 & -2 & 5 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 & 7 & -3 & -3 \\ 0 & 0 & 0 & 0 & -3 & 6 & -3 \\ 0 & 0 & 0 & -1 & -3 & -3 & 7 \end{bmatrix} \tag{2.13}$$

Step 2. Decomposition: Find Fielder value $\lambda_2[\mathcal{L}(G)]$ and Fielder vector $v_2(n)$ of the Laplacian matrix $\mathcal{L}$.

$$\lambda_2[\mathcal{L}(G)]=0.9671 \tag{2.14}$$

$$v_2(n) = \begin{matrix} [ & \mathrm{n}_1 & \mathrm{n}_2 & \mathrm{n}_3 & \mathrm{n}_4 & \mathrm{n}_5 & \mathrm{n}_6 & \mathrm{n}_7 & ] \\ & 0.42 & 0.22 & 0.42 & 0.22 & -0.4 & -0.48 & -0.4 & \end{matrix} \tag{2.15}$$

Step 3. Grouping: Split at 0 by dividing the sorted vector in two groups.

$$\text{Positive points : } \mathrm{N}_1 = \begin{matrix} [ & \mathrm{n}_1 & \mathrm{n}_2 & \mathrm{n}_3 & \mathrm{n}_4 & ] \\ & 0.42 & 0.22 & 0.42 & 0.22 & \end{matrix} \tag{2.16}$$

$$\text{Negative points : } \mathrm{N}_2 = \begin{matrix} [ & \mathrm{n}_5 & \mathrm{n}_6 & \mathrm{n}_7 & ] \\ & -0.4 & -0.48 & -0.4 & \end{matrix} \tag{2.17}$$

After calculating $N_1$ and $N_2$, the edge cut is shown in figure 2.4.

*2.4.2. Median Cut Spectral Bisection*

In order to balance the computational loading of the computer, the application for graph partitioning usually used the following optimization functions.

$$\begin{aligned} &\text{Objective : Min } Cut(N_1, N_2) \\ &\text{Constraint : equal sizes } N_1 = N_2 \end{aligned} \tag{2.18}$$

This partitioning method is called the median cut spectral bisection. It only needs to change $v_2(n) < 0$ to $v_2(n) <$ median of $v_2$. Figure 2.6 is used to illustrate the difference of these two methods.

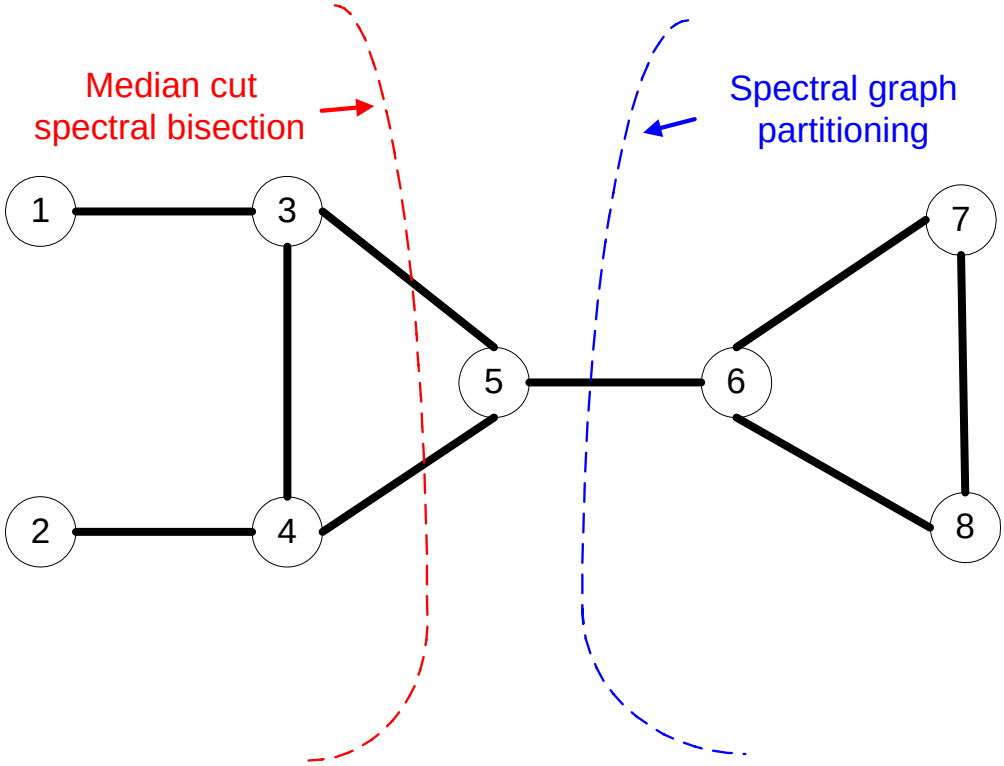



Figure 2.6 Comparison of two partition methods

Let every branch has the same weight in figure 2.6 and eight points will be divided into two parts. For the median cut spectral bisection method, it will force the graph to be divided into equal parts of areas. Every part has 4 nodes and 2 branches are cut. For the spectral graph partitioning method, only one branch is cut since it cuts the weakest link. From this example, one can see that the branches increase after cutting because median cut has added one more constraint. Besides, this method does not

guarantee the graph is connected after cutting. Therefore, the median cut spectral bisection method is unsuitable to apply in the power systems.

The similar extreme examples of figure 2.7 are often mentioned in the literature to show the deficiency of the spectral graph partitioning. This dissertation overcomes this issue by using the recursive function to subdivide the graph.

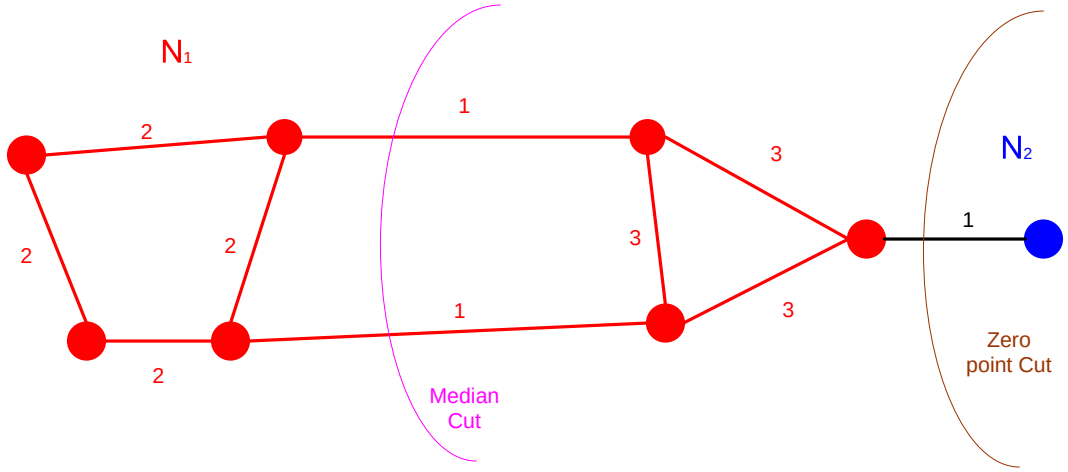



Figure 2.7 Degenerate case of minimum edge cut

# CHAPTER 3

# ALGORITHM FOR POWER SYSTEM PARTITIONING

## 3.1 Introduction

Power plant capacity in the United States expanded at an unprecedented rate between 1999 and 2003 [25]. In order to standardize this new trend and promote the development of power market, in the United State, the Federal Energy Regulatory Commission (FERC) issued the Order No. 888 in 1996 to initiate the processes of deregulation by requiring all public utilities to provide open access to transmission facilities. Three years later, FERC issued the Order No. 2000 in 1999 to request all transmission owning entities placing their transmission facilities under the control of Regional Transmission Organizations (RTOs). At the first stage, FERC has defined ten RTOs to cover the whole country. The initiation of Standard Market Design (SMD) was also set off in the year 2002. [26] Following SMD, the whole power grids in North America will be gradually becoming a single market.

Although interconnections result in operating economy and increase reliability through mutual assistance, they have also increased complexity of stability problems. The northeast blackout of August 14, 2003 brought again the problem of stability. [27] The engineers need to face the problem of solving and analyzing stability in the large-scale power system. Unfortunately, no commercial analysis software is available to

solve the large-scale power system such as in North America. Even though this problem can be solved by software and hardware in the future, calculation time will be very long. Fortunately, most of the stability problems start from local area. If unattended, it may spread out and become a system wide problem. In the course of a blackout development if the real-time simulation can detect the dangerous development, it will help engineers to avoid the systems fall into the uncontrollable situation so this dissertation proposed a new method to quickly solve the stability problem and reduce the computational time by partitioning the large-scale power system.

## 3.2 Power Flow Solution

The interconnected power grids have transmitted electricity for more than a century. Accordingly, the power systems have developed into large-scale systems. In order to have better understanding of the systems, power engineers normally study the characteristics of the system through computer simulation. The most common power system network simulation is that of power flow calculation. It is described in the following paragraph:

"In all buses within the power system, describe the electricity consumption in the loads and the power produced from generators in each power plant. Then, determine the power flow of every line and transformer in the interconnecting network."

The power system is designed to serve its loads and consider the optimization of capital and operating costs. The power system must be operated within the capacity of transmission lines and transformers and all bus voltages must stay within acceptable range. At the same time, the real and reactive power output of the generators must be

maintained within their ratings. For the power flow calculation, the following are the basic input parameters [18]:

1. Transmission line impedances and charging admittances.
2. Transformer impedances and tap ratios.
3. Admittances of shunt-connected devices such as static capacitors and reactors.
4. Load-power consumption at each bus of the system.
5. Real-power output of each generator or generating plant.
6. Either voltage magnitude at each generator bus or reactive power output of each generating plant.
7. Maximum and minimum reactive power output capability of each generating plant.

And the following quantities need to be solved:

1. The magnitude of the voltage at every bus where this is not specified in the input data.
2. The phase of the voltage at every bus.
3. The reactive power output of each plant for which it is not specified.
4. The real power, reactive power, and current flow in each transmission line and transformer.

Power flow calculation is a question of network solution. The network of transmission lines and transformers is expressed by the following linear algebraic equation:

$$I_n = Y_{nn} V_n \tag{3.1}$$

Where,

$I_n$: Vector of positive-sequence currents flowing into the network at its nodes (buses).

$V_n$: Vector of positive-sequence voltages at the network nodes (buses).

$Y_{nn}$: Network admittance matrix.

If $I_n$ or $V_n$ is known, the power flow calculation is straightforward. If both $I_n$ and $V_n$ are unknown, the programming work for power flow entails designing a series of test procedures for $I_n$ and $V_n$ in order to satisfy equation (3.1) and the load and generation condition in input data. Once $V_n$ is determined, the flows of transmission lines and transformers can be obtained from component equations.

Power flow question is nonlinear and its solution must be solved by the iterative process. A simple but efficient iterative method is described as follows:

1. Make an initial estimate of the voltage at each bus.
2. Build up an estimated current inflow vector, $I_n$, at each bus from a boundary condition such as

$$P_k + jQ_k = v_k i_k^* \tag{3.2}$$

where:

$P_k + jQ_k$: Net load and generation demand at bus k.

$v_k$: Present estimate of voltage at bus k.

3. Use equation (3.2) to obtain a new estimated bus voltage vector, $v_n$.
4. Return to Step 2 and repeat the procedure until it converges within an unchanging tolerance of $v_n$.

The first algorithm that power engineers used to solve the power flow question of the large-scale system is the Gauss-Seidel method. However, the Gauss-Seidel method resulted in a relatively bad convergence character. Thus the Newton method was developed to improve the convergence problem. In its wake, there have been a lot of theses and papers that further proposed algorithms to solve power flow. In summary, there are essentially five commonly used AC power flow iteration methods among industry:

1. Gauss-Seidel iteration.
2. Modified Gauss-Seidel iteration suitable for series capacitors.
3. Fully coupled Newton-Raphson iteration.
4. Decoupled Newton-Raphson iteration.
5. Fixed slope Decoupled Newton-Raphson iteration.

Based on whether or not the solution's convergence is determined by the network and load distribution, each method has its advantages and disadvantages. Combining more than one method is usually necessary to improve either convergence speed or accuracy in solving the power flow problems. A general rule of thumb to choose power flow solution methods are shown as follows:

1. The Gauss-Seidel methods are generally tolerant of power system operating conditions involving poor voltage distributions and difficulties with

generator reactive power allocation, but do not converge well in situations where real power transfers are close to the limits of the system.

2. The Newton-Raphson methods are generally tolerant of power system situations in which there are difficulties in transferring real power, but are prone to failure if there are difficulties in the allocation of generator reactive power output or if the solution has a particularly bad voltage magnitude profile.
3. The Gauss-Seidel methods are quite tolerant of poor starting voltage estimates, but converge slowly as the voltage estimate gets close to the true solution.
4. The Newton-Raphson methods are prone to failure if given a poor starting voltage estimate, but are usually superior to the Gauss-Seidel methods once the voltage solution has been brought close to the true solution.

The optimal combination of methods for every power system model can be determined by testing them yourself. For a new and unclear power flow case, the following procedure is suggested.

1. Initialize all voltages to either unity amplitude, or to scheduled amplitude if given, and initialize all phase angles to zero. (This step is referred to as a flat start.)
2. Execute Gauss-Seidel iterations until the adjustments to the voltage estimates decrease to, say, 0.01 or 0.005 per unit in both real and imaginary parts.

3. Switch to Newton-Raphson iterations until either the problem is converged, or the reactive power output estimates for generators show signs of failure to converge.
4. Switch back to Gauss-Seidel iterations if the Newton-Raphson method does not settle down to a smooth convergence within 8 to 10 iterations.

The whole solving procedure can be modified by the experience of each special problem. Particularly, when you know that the previous solution is similar to the estimation of a new expected solution, then the initial Gauss-Seidel iterations and flat-start steps must be stopped and you must switch to other methods.

The power system's stability can be simulated after solving the power flow. Although an automatic solving program will be designed, readers also need to follow the above procedure, step by step, to experiment until the convergent solution is reached.

## 3.3 Applying Partitioning Algorithm in Jacobian Matrix

When a power system operated under a steady load condition, the steady-state solution is called power flow. When this system is perturbed, causing the readjustment of the voltage angles of the synchronous machines, the dynamic behavior during this time is system stability problem. The power flow solution is the initial condition of stability problem and the relationship between voltage angles and power flow will describe in this section. The key point is the relationship between the Jacobian matrix in power flow and the Laplacian matrix mentioned in the previous chapter. In the end of this section, the application of the partitioning algorithm for the Jacobian matrix will be discussed.

The original problem is when electric utilities or areas have a very large power flow exchange, simulating the entire power system is either impossible or extremely time consuming because of the maximum bus limitation in the commercial software. Therefore, the first step is to appropriately subdivide the large-scale power system by converting it into a graph partition problem. The object of this transformation is to solve the dynamic stability problem by finding the deviation of power flow.

First, the basic mathematical expression of the Jacobian matrix must be derived. The matrix in equation (3.1) needs to be transferred into general form for every single term. For bus k, the $k_{th}$ equation in equation (3.1) is

$$I_k = \sum_{n=1}^{N} Y_{kn} V_n \tag{3.3}$$

Using equation (3.3) in equation (3.2),

$$P_k + jQ_k = V_k \left[ \sum_{n=1}^{N} Y_{kn} V_n \right]^* \qquad k = 1,2,...,\mathrm{N} \tag{3.4}$$

With the following notation,

$$\begin{aligned} V_n &= V_n e^{\mathrm{j}\delta_n} \\ Y_{kn} &= Y_{kn} e^{\mathrm{j}\theta_{kn}} \qquad k,n = 1,2,...,\mathrm{N} \end{aligned} \tag{3.5}$$

Equation (3.4) becomes

$$P_k + jQ_k = V_k \sum_{n=1}^{N} Y_{kn} V_n e^{j(\delta_k - \delta_n - \theta_{kn})} \tag{3.6}$$

Separating the real and imaginary parts of equation (3.6), one can obtain the following two equations,

$$P_k = V_k \sum_{n=1}^{N} Y_{kn} V_n \cos(\delta_k - \delta_n - \theta_{kn})$$
$$Q_k = V_k \sum_{n=1}^{N} Y_{kn} V_n \sin(\delta_k - \delta_n - \theta_{kn}) \qquad k = 1,2,...,\mathrm{N} \tag{3.7}$$

Equations 3.7 are analogous to the nonlinear equation $y = f(x)$. We define the x, y, and f vectors for the power-flow problem as

$$x = \begin{bmatrix} \delta \\ V \end{bmatrix} = \begin{bmatrix} \delta_2 \\ \vdots \\ \delta_N \\ V_2 \\ \vdots \\ V_N \end{bmatrix}; \qquad y = \begin{bmatrix} P \\ Q \end{bmatrix} = \begin{bmatrix} P_2 \\ \vdots \\ P_N \\ Q_2 \\ \vdots \\ Q_N \end{bmatrix};$$
$$f(x) = \begin{bmatrix} P(x) \\ Q(x) \end{bmatrix} = \begin{bmatrix} P_2(x) \\ \vdots \\ P_N(x) \\ Q_2(x) \\ \vdots \\ Q_N(x) \end{bmatrix} \tag{3.8}$$

Where all V, P, and Q are in per-unit and $\delta$ terms are in radians. Since they are normally set to 0 and 1, the variables $\delta_1$ and $V_1$ of the swing bus are omitted. Equations 3.6 then have the following form:

$$y_k = P_k = P_k(x) = V_k \sum_{n=1}^{N} Y_{kn} V_n \cos(\delta_k - \delta_n - \theta_{kn})$$
$$y_{k+N} = Q_k = Q_k(x) = V_k \sum_{n=1}^{N} Y_{kn} V_n \sin(\delta_k - \delta_n - \theta_{kn}) \qquad k = 2,3,...,\mathrm{N} \tag{3.9}$$

The Jacobian matrix of power flow has the form

$$J = \begin{array}{c} \begin{matrix} \multicolumn{3}{c}{J_1} & \multicolumn{3}{c}{J_2} \end{matrix} \\ \left[\begin{array}{ccc|ccc} \frac{\partial P_2}{\partial \delta_2} & \cdots & \frac{\partial P_2}{\partial \delta_N} & \frac{\partial P_2}{\partial V_2} & \cdots & \frac{\partial P_2}{\partial V_N} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial P_N}{\partial \delta_2} & \cdots & \frac{\partial P_N}{\partial \delta_N} & \frac{\partial P_N}{\partial V_2} & \cdots & \frac{\partial P_N}{\partial V_N} \\ \hline \frac{\partial Q_2}{\partial \delta_2} & \cdots & \frac{\partial Q_2}{\partial \delta_N} & \frac{\partial Q_2}{\partial V_2} & \cdots & \frac{\partial Q_2}{\partial V_N} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial Q_N}{\partial \delta_2} & \cdots & \frac{\partial Q_N}{\partial \delta_N} & \frac{\partial Q_N}{\partial V_2} & \cdots & \frac{\partial Q_N}{\partial V_N} \end{array}\right] \\ \begin{matrix} \multicolumn{3}{c}{J_3} & \multicolumn{3}{c}{J_4} \end{matrix} \end{array} \tag{3.10}$$

After obtaining the Jacobian matrix, the next significant step is to adjust and extract Jacobian matrix according to the purposes of the applications. The power flow Jacobian must be calculated from the initial condition using equation (3.10) simplified as the following matrix:

$$J = \begin{bmatrix} \frac{\partial \mathbf{P^N}}{\partial \boldsymbol{\delta}} & \frac{\partial \mathbf{P^N}}{\partial \mathbf{V_L}} \\ \frac{\partial \mathbf{Q^N}}{\partial \boldsymbol{\delta}} & \frac{\partial \mathbf{Q^N} - \mathbf{Q^I}}{\partial \mathbf{V_L}} \end{bmatrix} \tag{3.11}$$

In addition to the traditional Jacobian matrix solution [19], another technique can be used to account for the power system losses. For the purposes of this proposed application, this derivation [20] is deduced and modified during the final phase of the development. The overbar of the form"$\vec{\ }$" is used to indicate a complex value quality. $\Re^n$ is a vector with n real numbers. Similarly, $C^n$ is a vector with n complex numbers and $C^{n\times n}$ is a matrix with $n \times n$ complex numbers. Let

$\delta \in \Re^n, \delta :=$ vector of bus voltage phase angles relative to an arbitrary synchronous reference frame of frequency $\omega_0$ (no reference angle is deleted);

$V \in \Re^n, V :=$ vector of variable bus voltage magnitudes;

$\vec{V} \in C^n, \vec{V} :=$ vector of complex bus voltages; note that $\vec{V} = V.*\exp(j\delta)$

$\overrightarrow{Y} \in C^{n \times n}, \overrightarrow{Y} :=$ full bus admittance matrix (reference bus rows and column not eliminated);

$\vec{I} \in C^n, \vec{I} :=$ vector of currents absorbed into network at each bus; note that $\vec{I} = \vec{Y} * \vec{V}$

$\vec{S} \in C^n, \vec{S} :=$ vector of complex powers absorbed into network at each bus;

note that $\vec{S} = \vec{V}.*conj(\vec{I}) = conj(\vec{I}).*(\vec{V})$

$$\frac{\partial \vec{V}}{\partial \delta} = \frac{\partial \{V.*\exp(j\delta)\}}{\partial \delta} = j*\text{diag}\{\vec{V}\} \tag{3.12}$$

$$\frac{\partial \vec{I}}{\partial \delta} = \vec{Y} * \frac{\partial \{V.*\exp(j\delta)\}}{\partial \delta} = j*\vec{Y}*\text{diag}\{\vec{V}\} \tag{3.13}$$

$$\frac{\partial \{\text{conj}(\vec{I})\}}{\partial \delta} = \text{conj}(j*\vec{Y})*\text{diag}\{\text{conj}(\vec{V})\} = \text{conj}(j*\vec{Y}*\text{diag}\{\vec{V}\}) \tag{3.14}$$

Thus,

$$\frac{\partial \vec{S}}{\partial \delta} = j*\text{diag}\{\text{conj}(\vec{I}).*\vec{V}\} - j*\text{diag}\{\vec{V}\}*\text{conj}(\vec{Y})*diag\{\text{conj}(\vec{V})\} \tag{3.15}$$

$$\frac{\partial \mathbf{P^N}}{\partial \boldsymbol{\delta}} = \left| real\left(\frac{\partial \vec{S}}{\partial \delta}\right)\right| \tag{3.16}$$

Once the eigenvalue for the Jacobian matrix is solved, the variation in every branch can be used to calculate the power variation matrix $\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}}$ and produce a nondirectional graph. No matter whether the power flow increases or decreases in each branch, it is the absolute of variation that is of interest. Therefore, the diagonal term of $\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}}$ is deleted and the sums of each column are inserted into the diagonal elements.

The block $\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}}$ obtained from this derivation, which still preserves the properties of the weighted Laplacian matrix [21], is then combined with the sensitivity area so spectral graph partitioning can be applied in the next stage.

## 3.4 The Sensitivity Area

Following the previous study to compute some prospective sensitivity areas near the fault bus, then further verify by newly-developed partition algorithm method. One will be able to identify a more precise area by using these two methods jointly.

Solving the sensitivity area requires injection of 1MW power into the fault bus, followed by calculation of the power flow difference in each branch to identify those branches with larger variations and find the buses in their two ends. For this method, the power flow must first be solved; however, if the system is too large, an alternative method can be used to find the nearest n layers of buses from the fault bus, which then become the sensitivity area. If a system is small, for example Thailand system, the first method can be used. If systems are too large, for example the power systems in US, the second method need to be applied.

Once the sensitivity area is identified, the original system can be combined with the sensitivity area to produce the following equation:

$$J_{final} = J + w\mathcal{L} \tag{3.17}$$

Where,

$J_{final}$ : Final Jocobian matrix

$J = \frac{\partial P^N}{\partial \delta}$ : Part of Jocobian matrix of the original system

$\mathcal{L}$ : Laplacian matrix of the sensitivity area

$w$ : The weighted parameter

If $w$ is higher, the branch in the sensitivity area is stronger and therefore not easily cut off in the program.

## 3.5 Eigenvector Based partitioning

The next step is the spectral graph partition of the $J_{final}$ matrix. In this Laplacian matrix, the second smallest eigenvalue $\lambda_2(J_{final})$, referred to as the algebraic connectivity of graph [22], and its corresponding eigenvector are calculated. The Fiedler value indicates a relative coupling between the resulting 2 subareas – i.e., a higher value represents higher coupling. The eigenvector is then sorted to separate the system. The following flow chart describes this partition process.

Compute the eigenvector $v_2$

corresponding to $\lambda_2$ of $J_{final}$

for each node n of $J_{final}$

if $v_2(n) < 0$

put node n in partition N-

else

put node n in partition N+

endif

endfor [23]

In accordance with theorem [24], G is assumed to be connected. If N- and N+ are defined by the above algorithm, N- is connected and if no $v_2(n) = 0$ , N+ is also connected. Because the two subareas after cutting are still connected, further recursive partition is possible.

The recursive spectral graph partitioning method terminates recursion given the following two conditions:

$$N_{bus} < N \text{ or } \lambda < C \tag{3.18}$$

Where,

$N_{bus}$ : The bus number of an oncoming re-cutting area

$N$ : A reasonable bus number

λ: The second smallest eigenvalue of an oncoming recutting area. The second smallest eigenvalue means that the two subnetworks display a higher or lower degree of connectivity.

C: A reasonable connective parameter. If the value of λ is high, the connection inside this area is so strong that continuous separation of the area is impossible.

The connected property of this partition method guarantees no island in the subarea after partitioning.

## 3.6 A Small Example

The above theory is deduced from formulae. A simple example of the IEEE 9-bus power system is used to explain the entire process. This small system includes three generators injecting power into the system and feeding three loads, as shown in figure 3.1.

First, calculate the Y matrix of the system and then consider the initial condition as the bus current and voltage, I and V, into equation (3.15).

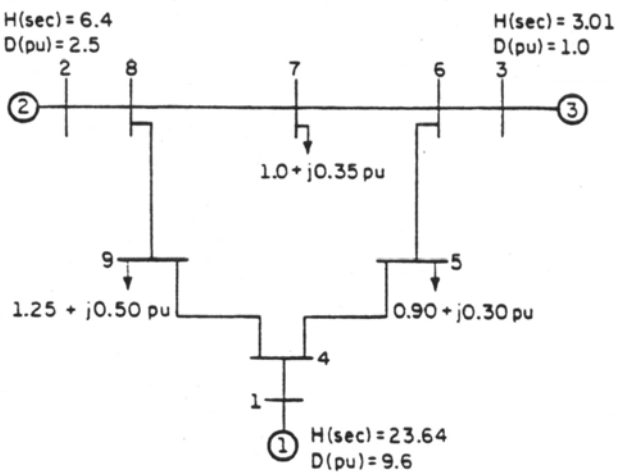



Figure 3.1 The IEEE 9-bus sample system

In order to obtain the power variation matrix $\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}}$, the absolute variation of each branch is considered in equation 3.14. For example, if the equivalent circuit in a branch is inductive, the corresponding term in $\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}}$ will be negative. On the other hand,

if the equivalent circuit in a branch is capacitive, the corresponding term in $\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}}$ will be positive. Because the graph is non-directional for the proposed algorithm, the absolute value of the variation will accord with the applicable purpose. Irrespective of whether the power flow increases or decreases in each branch, its variation will influence the power system stability.

Calculate the block Jacobian $\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}}$ by using the median cut spectral bisection graph partitioning method; this matrix is also a weighted Laplacian matrix as shown in the following equation:

$$\frac{\partial \mathbf{P}^{\mathbf{N}}}{\partial \boldsymbol{\delta}} = \begin{bmatrix} 20.9 & 0 & 0 & -20.9 & 0 & 0 & 0 & 0 & 0 \\ 0 & 19.2 & 0 & 0 & 0 & 0 & 0 & -19.2 & 0 \\ 0 & 0 & 20.4 & 0 & 0 & -20.4 & 0 & 0 & 0 \\ -20.9 & 0 & 0 & 47.1 & -12.5 & 0 & 0 & 0 & -13.7 \\ 0 & 0 & 0 & -12.4 & 18.9 & -6.5 & 0 & 0 & 0 \\ 0 & 0 & -20.4 & 0 & -6.8 & 38.9 & -11.8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -11.7 & 28.0 & -16.3 & 0 \\ 0 & -19.3 & 0 & 0 & 0 & 0 & -16.5 & 42.9 & -7.1 \\ 0 & 0 & 0 & -13.5 & 0 & 0 & 0 & -6.9 & 20.4 \end{bmatrix} \quad (3.19)$$

If only the median cut spectral bisection is considered, simply calculate the second smallest eigenvalue and its eigenvector and follow the flow chart process provided in section 3.4. The bisection result can be obtained as shown in Figure 3.2.

However, in order to simulate the transient stability, it is necessary to introduce fault in the system; for example, if the fault occurs on bus eight, the nearby bus nine should be affected by the fault and both buses should be positioned in the same area.

Therefore, it is necessary to define the sensitivity area near a fault and then readjust the sub-areas.

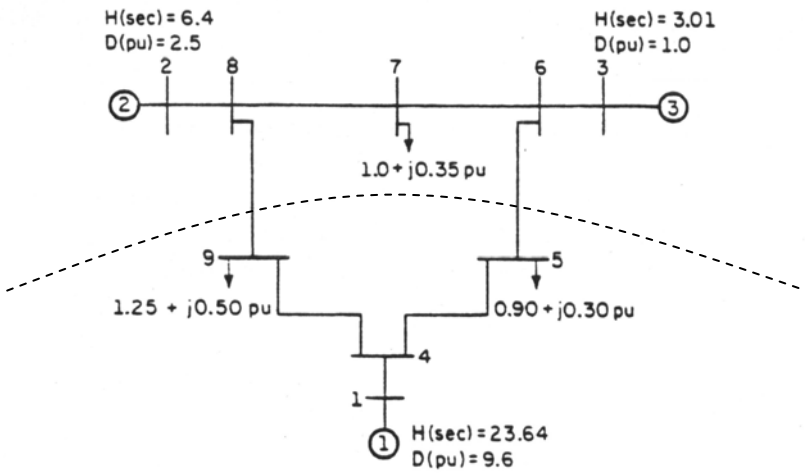



Figure 3.2 The median cut spectral bisection result for the IEEE 9-bus sample system

There are two methods to define the sensitivity area. First, a power of 1 MW can be injected into the fault bus to identify more buses that sensitive. This method requires the calculation of the power flow of the system. If the system is too large to calculate the power flow, one can identify the nearest n layers of buses from the fault bus.

In this example, the objective is to combine the sensitive areas and force buses 4, 5, and 6 into the same sub-area.

The adjacency matrix of bus 4, 5, and 6:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.20}$$

Calculate Laplacian matrix, $\mathcal{L}$, from A.

$$\mathcal{L} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.21}$$

Here, a new parameter, w, will be defined to fit the application. If w=20, use the Laplacian matrix, $\mathcal{L}$, of sensitivity area to increase the weight of bus 4, 5 and 6 and then the final Jacobian matrix is shown in following equation.

$$J_{final} = \begin{bmatrix} 20.9 & 0 & 0 & -20.9 & 0 & 0 & 0 & 0 & 0 \\ 0 & 19.2 & 0 & 0 & 0 & 0 & 0 & -19.2 & 0 \\ 0 & 0 & 20.4 & 0 & 0 & -20.4 & 0 & 0 & 0 \\ -20.9 & 0 & 0 & 67.1 & -32.5 & 0 & 0 & 0 & -13.7 \\ 0 & 0 & 0 & -32.4 & 58.9 & -26.5 & 0 & 0 & 0 \\ 0 & 0 & -20.4 & 0 & -26.8 & 58.9 & -11.8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -11.7 & 28.0 & -16.3 & 0 \\ 0 & -19.3 & 0 & 0 & 0 & 0 & -16.5 & 42.9 & -7.1 \\ 0 & 0 & 0 & -13.5 & 0 & 0 & 0 & -6.9 & 20.4 \end{bmatrix} \quad (3.22)$$

Use this new parameter to bisect the system and two algorithms can be successfully combined. The partitioning result is shown in figure 3.3.

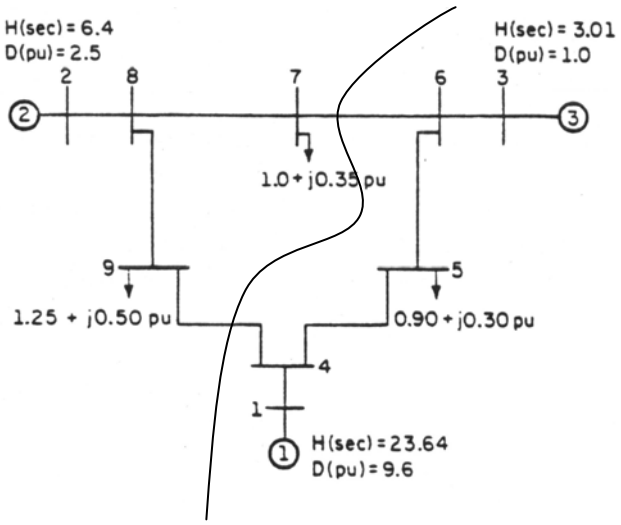



Figure 3.3 The spectral graph partitioning result for the IEEE 9-bus sample system

In order to improve the executive efficiency when performing the iterations, a new system matrix can be extracted from the Jacobian matrix. The parameters of the subsystem need not be calculated again. This method is explained with the following simple example. The branch from bus 1 to bus 4 is to be extracted. Only the elements (1,4) and (4,1) in the final Jacobian matrix need to be extracted since it is a symmetric matrix. Therefore, the subsystem can easily be extracted as below.

$$J_{subsystem} = \begin{bmatrix} 0 & -20.9 \\ -20.9 & 0 \end{bmatrix} \qquad (3.23)$$

Transform this subsystem matrix to a Laplacian form. This matrix then represents the variation of the entire subsystem. We need not recalculate it from raw data.

# CHAPTER 4

# THE TEST RESULTS

## 4.1 Introduction

The stability problem has existed for a long time since the development of power industry. Traditionally, stability study focuses on the ability of synchronous generator to maintain synchronous operation during the first swing after severe disturbances. In modern power system, due to many fast response devices are installed for better control, it has been shifted to deal with those situations that may create a low frequency oscillation due to insufficient damping torque. The evaluation of stability problems then requires the study of power system's response during the transient period right after fault, the post fault condition, and during the steady state operation conditions. In addition to the information that is required for power flow study, information concerning the dynamic response of generators and their controller devices, including generator inertia, transient and sub-transient impedance, governor-control (GOV) characteristic, automatic voltage regulator (AVR) characteristics, and protective device settings and response time, are necessary for stability analysis. It should be noted that the accuracy of the input parameters affected the simulation results significantly. [28]

Therefore, analyzing power system's dynamic stability will be a complicated and time consuming task. Fortunately, most of the dynamic stability problems only affect local areas. This dissertation offers an approach for fast investigation on the dynamic response of the power system after the disturbances with reasonable accuracy.

The proposed algorithm for power flow and transient stability described in chapter 3 is tested in two different sizes of systems. The first system, Thailand, is a smaller system that consists of 1204 active buses and one DC interconnection with Malaysia in the southern part of the system. The basic Thailand system information lists in table 4.1.

The other one is the complete 2005 summer peak Southwest Power Pool (SPP) system with 29,657 buses, 40,080 branches, 5,208 generators, and 19,450 loads. It should be pointed out, however, only 29,528 buses are active and the system contains only one swing bus. These tests were performed to compare the execution time and accuracy of results for the proposed approach against those for the full size system simulations [29]. All simulations presented in the dissertation are performed on 2.4 GHz Pentium 4 processor and 512 MB of RAM.

Table 4.1 The basic system information of Thailand and SPP system

| | Thailand | SPP |
|---|---|---|
| Total load | 18710 MW | 522282 MW |
| Total generation | 19214 MW | 536188 MW |

## 4.2 The Simulation Process

The first step is to generate reasonable power flow base case. The power flow study is the most commonly used tool in power system calculation. It calculates steady state flow through lines and transformers, and bus voltages throughout power system under normal or contingency conditions.

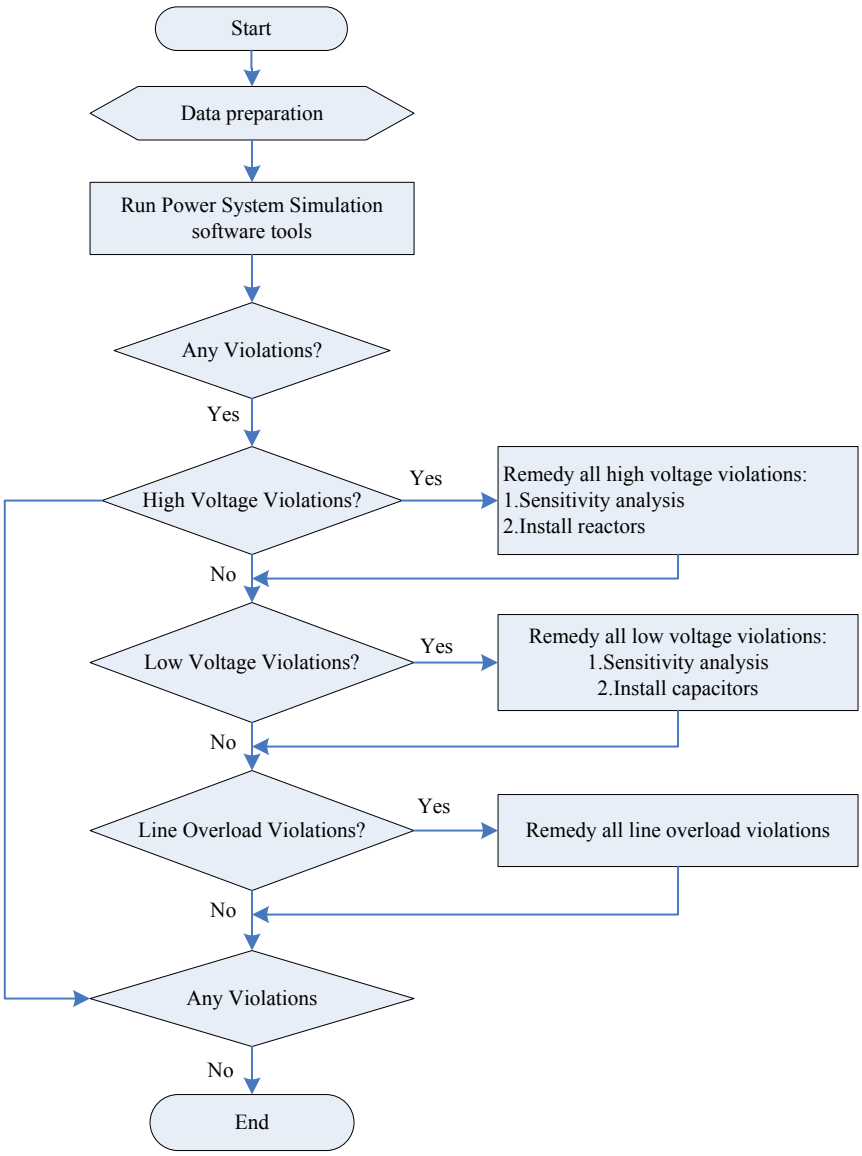



Figure 4.1 Systematic approaches to create reasonable power flow base case.

During the process of generating reasonable power flow base case, several repetitive steps of switching on/off of capacitors and reactors need to be executed in order to bring the bus voltage within the range between 0.95 and 1.05 per unit and eliminate any transmission thermal rating violation.

As shown in the second block of figure 4.1, all system data including network topology and peak load are first checked and verified. Consistencies of system base MVA, base kV, and base frequency are the primary concerns. The procedure described in figure 4.1 is used to eliminate high voltage, low voltage, and overloading situations through capacitor/reactor switching, re-dispatching, load curtailment and/or generation rejection.

After obtaining the power flow solution, the proposed partitioning method in chapter 3 can continue to divide the large system into the smaller sub-sets in order to further analyze the system performance. Before creating a fault, a no-fault test run was performed using the PSS/E dynamic simulation tool to ensure the accuracy of the data. The dynamic study model was initialized using the steady state operating point from the load flow case as the initial condition. When the size of the system exceeds the capability of the simulation program, basic load flow data can also be obtained from CADA/EMS system. If a simulation performed with no disturbance for a 5-second period, this result shows that the dynamic model is working as desired. [30] Otherwise, the dynamic models need to be modified again. If the data is clear, PSS/E can properly run stability problem in sub-systems in the tests.

## 4.3 Simulation Examples

The main objective of this section is to perform a full stability test for Thailand power system in various system conditions ranging from normal operation, and during and after fault conditions. While the power flow study result examines the power system under steady-state conditions, dynamic study inspects the system responses to various types of disturbance. These dynamic studies analyze system behavior over time intervals ranging from cycles to seconds.

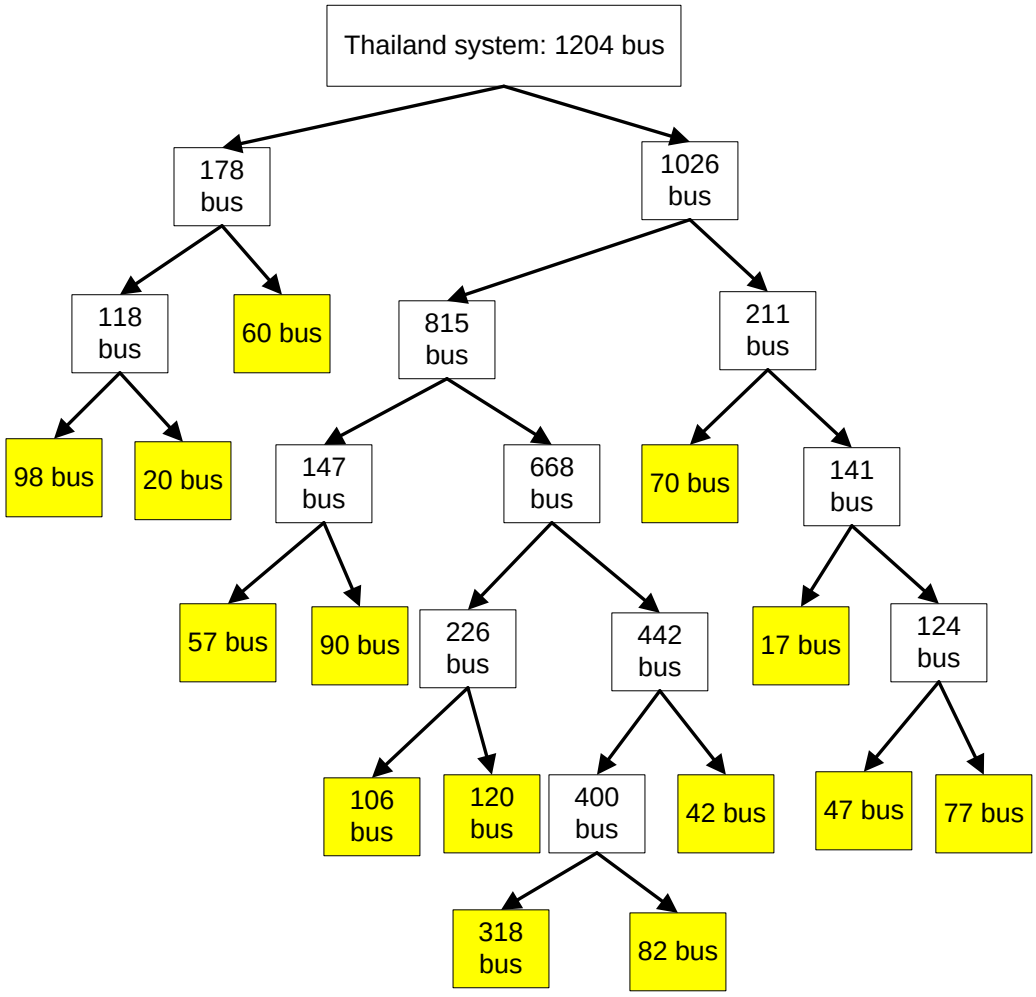



Figure 4.2 The partition of the original Thailand system.

The system models include load flow data and dynamic data. When transient stability is simulated, the Dynamics Raw Data File, which are also ASCII text files need to be added. The steady state and stability simulations in this chapter were performed using the PSS/E v. 30.1.

Before testing the proposed algorithm, the Thailand system has to be partitioned first. The corresponding partitioning processes and results are shown in figure 4.2.

The yellow area of figure 4.2 shows the final sub-areas, the tip elements of the binary tree. The final result is shown in figure 4.3.

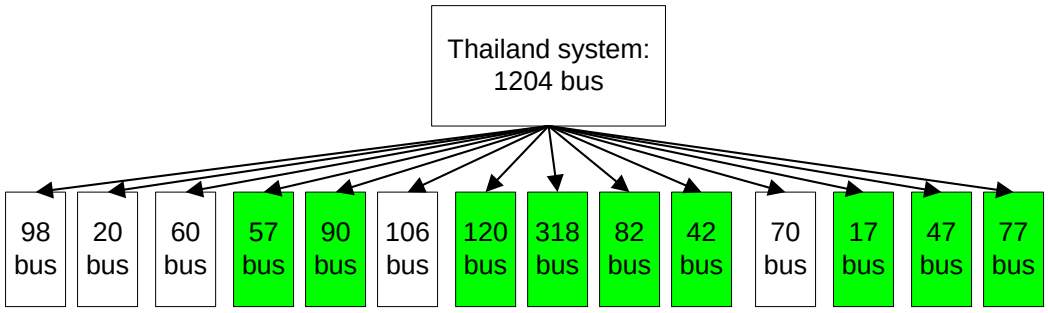



Figure 4.3 The final area chosen for the fault on bus 1013.

Once the sensitivity area is calculated, the final divided areas are the subareas that cover the sensitivity area. For example, if the fault bus is bus 1013, 1MW in the fault bus is injected and then chose 100 buses for the sensitivity area. After cutting the islands in the sensitivity area, the repeat areas showed in green blocks in figure 4.3 can be found. For injection of 1MW power method, the concept is illustrated in figure 4.4 and more details were discussed in section 3.4.

The pre-partitioning blocks in Figure 4.3 will remain the same if the system topology does not change. This process can be performed in advance to save the

computation time. When there are accidents in the systems, one can start the algorithm simply by calculating the sensitive areas. The information on the partition areas will be stored in the database. This will save computational time when several cases are studied. The computation time can further be reduced if computer network is used to solve multiple cases simultaneously.

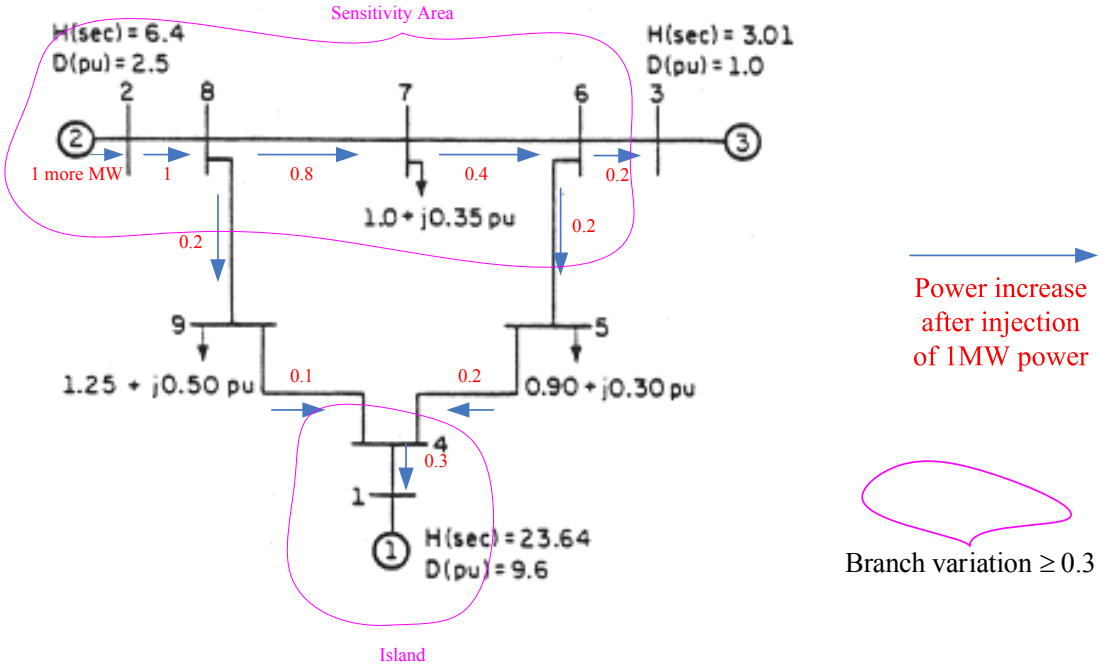



Figure 4.4 An example of an island in the sensitivity area.

The stability simulation files and test cases for Thailand system are discussed below.

1_case10_Oct6_V30.raw: The system power flow raw data.

2_Machine_DataY04.dyr: The program will automatically add some new generator models to compensate the boundary flow in separated system.

3_PowerFlow_CONLG.py : Use python program to divide the system and run power flow.

4_Dynamic_Initial.idv: No fault test. Just test whether the initial condition is stable.

5_Transient_gen1013_fault.idv: The three-phase short circuit fault in generation bus 1013.

6_Transient_gen1013_fault&out.idv: After the fault, trip the generator in bus 1013.

7_Transient_gen7035_fault.idv: The three-phase short circuit fault in generation bus 7035.

8_Transient_gen7035_fault&out.idv: After the fault, trip the generator in bus 7035.

9_Transient_Line_fault&trip.idv: The three-phase short circuit fault in the transmission line between bus 1804 and 6801. After the fault, trip the line.

After simulated all stability process, the result listed in appendix I. Table 4.2 shows the comparison of the simulation results.

Table 4.2 The comparison of the simulation results for Thailand system

| How many layer of buses near the fault? | 10 | 8 | 7 | 6 | 5 |
|---|---|---|---|---|---|
| Step 5 | O | O | V | V | X |
| Step 6 | O | O | V | V | X |
| Step 7 | O | O | V | O | V |
| Step 8 | O | O | V | O | V |
| Step 9 | O | O | V | V | X |

O: The simulation results for the partitioning system are almost the same as those for the whole system.

V: The simulation results for the partitioning system are similar with those for the whole system.

X: The simulation results for the partitioning system are different from those for the whole system.

## 4.4 The Comparison of the Simulation Results

In order to have true response in Southwest Power Pool (SPP) system, complete 2005 summer peak SPP data has been used in the simulation. Similar partitioning process is applied. The corresponding partitioning processes and results are shown in figure 4.5.

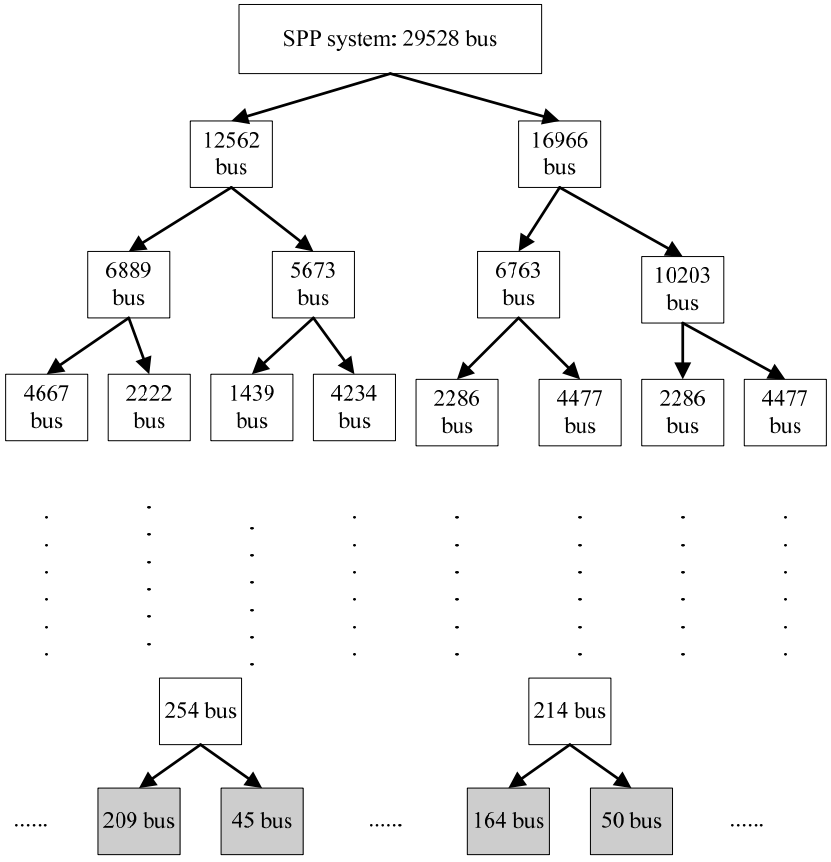



Figure 4.5 The partition of the original SPP system.

The gray area of figure 4.5, the tip elements of the binary tree, shows the final subareas. Once the sensitivity area is calculated, the final cutting areas are the subareas that cover the sensitivity area as shown in the black blocks in figure 4.6.

Another partition process is also deduced by first calculating the Laplacian matrix of the sensitivity area and then directly weighting the Laplacian matrix to the original system. The final Jacobian matrix can now apply to any partitioning method. The more detail calculation method is listed in formula 3.17. As long as the weighting is not large, both methods are feasible and their results are almost identical. After partitioning, users must first run the power flow, after which the program will assign a swing bus if the separated area does not include the swing bus from the original system.

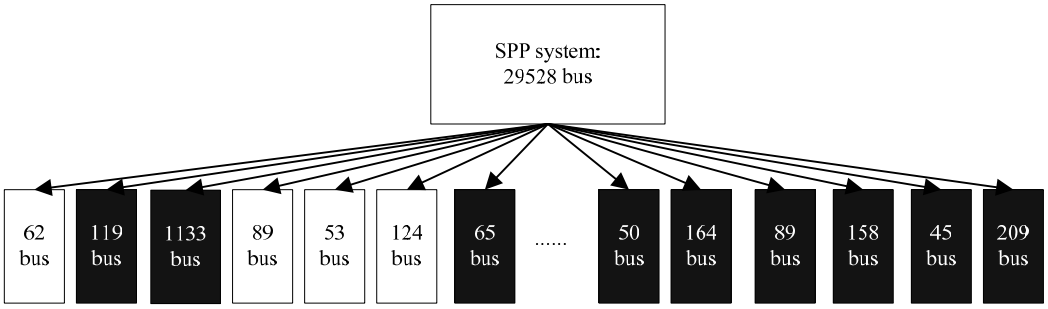



Figure 4.6 The final area chosen for the fault on bus 25044.

*4.4.1. Single fault scenarios*

To verify the proposed algorithm, six independent faults at different locations were studied. Due to the confidentiality of the data, only the relative locations of the faults are shown in Fig. 4.7.

The adjustable parameters chosen for the simulation are explained below:

w = 1 (which means no weighting). The original Jacobian matrix is used to partition the power system.

N = 100, because the total number of buses in the SPP system is 29,528. Every subarea will then be around 100 buses.

C = 0.5. If the coupling of the next two subareas is tight, this parameter can avoid forward recursive partitioning.

During data validation, a two-second no-fault test was performed to check the data integrity. A 5-cycle fault is then created at two seconds and allows the simulation to run for 15 seconds.

All necessary files for simulation are listed below.

1_ts05s2x_ori_solve.sav: A reasonable system base case raw data.

2_nerc-all-NO-SPP.dyr, 2_Ts-05SP.dyr: The system dynamic model.

3_PowerFlow_CONLG.py , 3_CONVLOADS-GENS.idv: Use python program to divide the system. Then use idv file to run power flow in dynamic environment.

4_SPPsystem_initial.idv: No fault test to check the integrity of the data.

5_SPPsystem_F54121.idv: A three-phase short circuit fault on the load bus 54121.

6_SPPsystem_F87456.idv: A three-phase short circuit fault on the generator bus 87456.

7_SPPsystem_F64895.idv: A three-phase short circuit fault on the load bus 64895.

8_SPPsystem_F31426.idv: A three-phase short circuit fault on the load bus 31426.

9_SPPsystem_F25044.idv: A three-phase short circuit fault on the generator bus 25044.

10_SPPsystem_F83904.idv: A three-phase short circuit fault on the load bus 83904.

After all stability cases have been simulated, compare all the simulation graphs and the results are listed in Table 4.3.

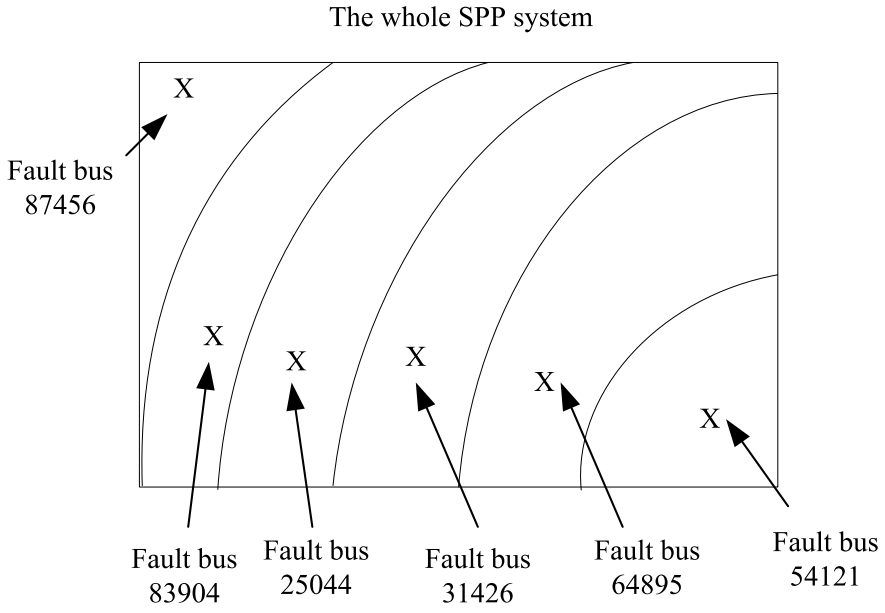



Figure 4.7 The six fault positions.

As shown in Figures 4.8 and 4.9, the simulation results for the partitioning system are almost the same as those for the whole system. In Figures 4.10 and 4.11, the simulation results for the partitioning system are similar with those for the whole system and in Figures 4.12 and 4.13, the simulation results for the partitioning system

are different from those for the whole system. Because the four simulation results on fault bus 25044 are the same as those for the original system and fault bus 87456 is near the boundary, both cases were chosen to compare simulation time. The simulation environment is a laptop with 2.4G Hz CPU, 512MB RAM and Microsoft Windows XP Home Edition. The comparative results for the simulation time of dynamic stability are listed in Table 4.4.

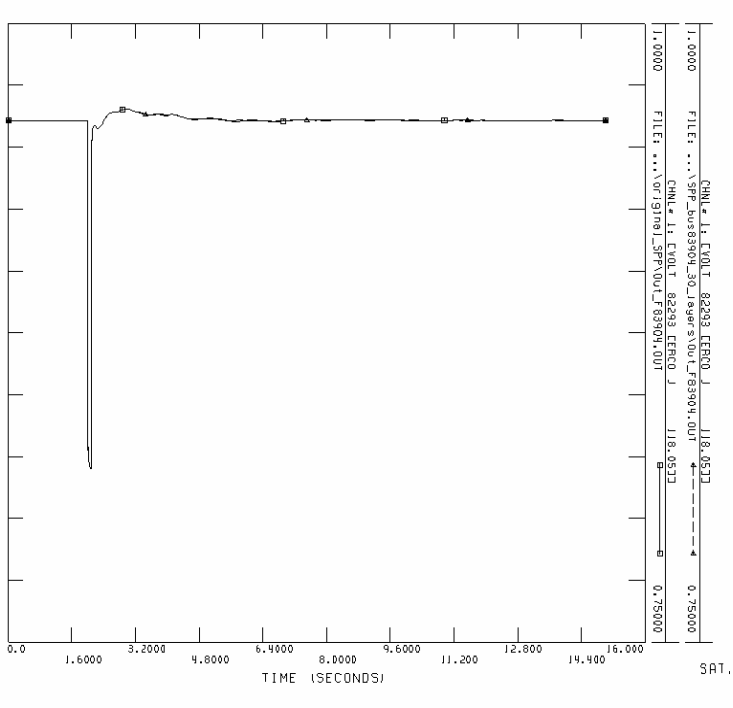



Figure 4.8 The voltage profile comparison on the Bus 82293

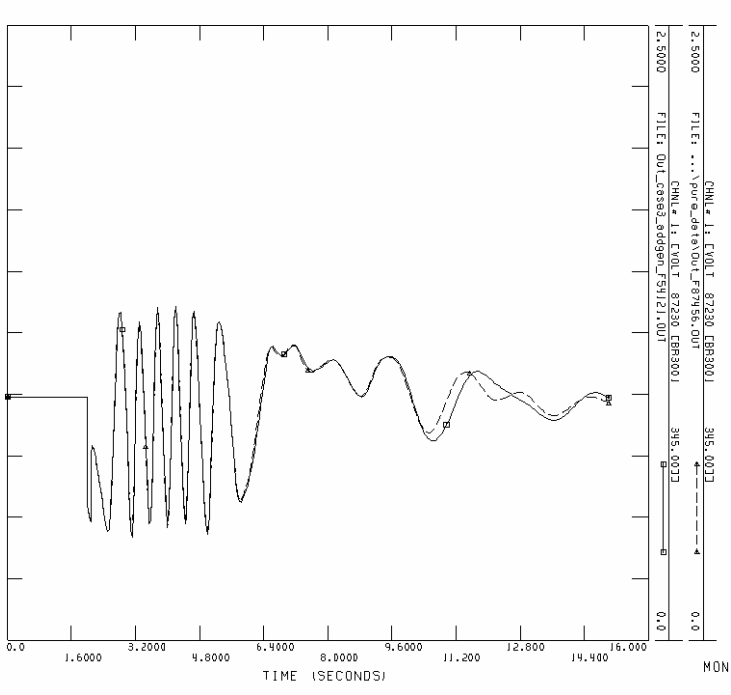



Figure 4.9 The voltage profile comparison on the Bus 87230

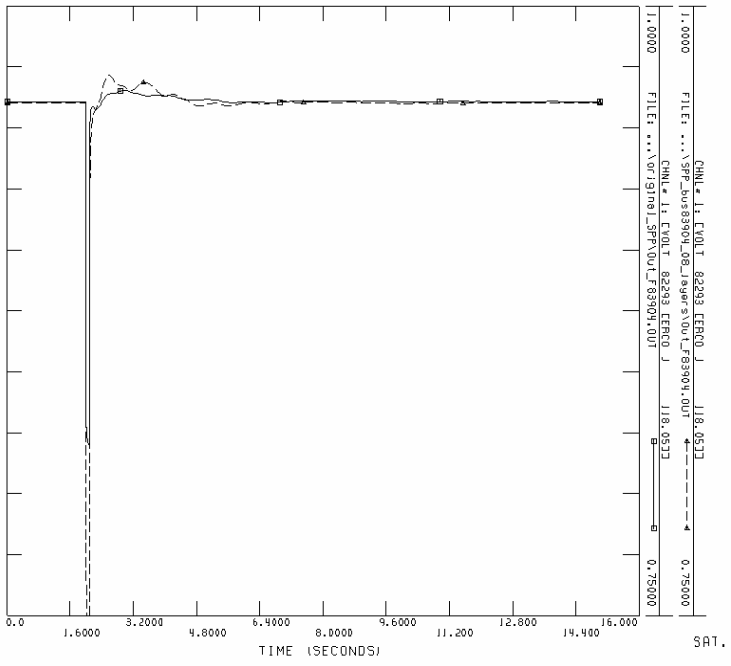



Figure 4.10 The voltage profile comparison on the Bus 82293

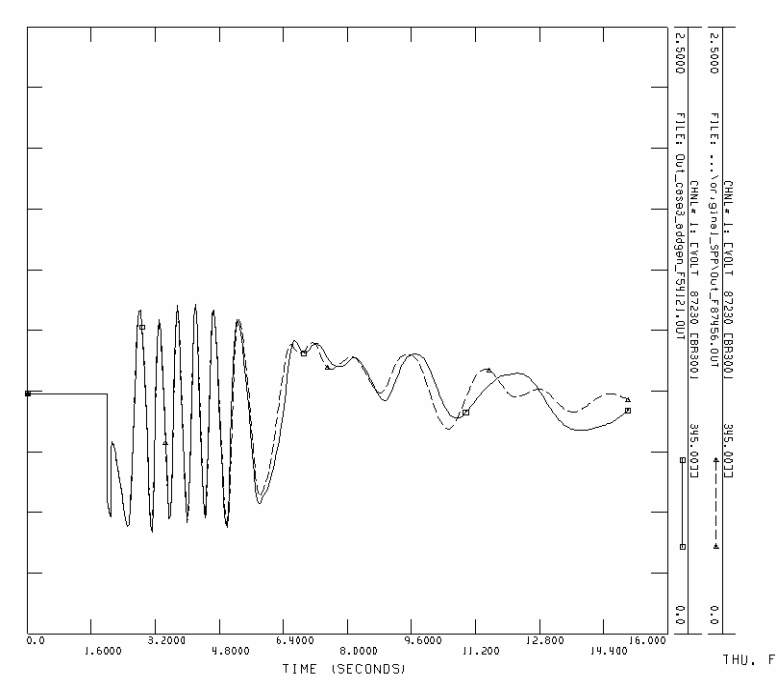



Figure 4.11 The voltage profile comparison on the Bus 87230

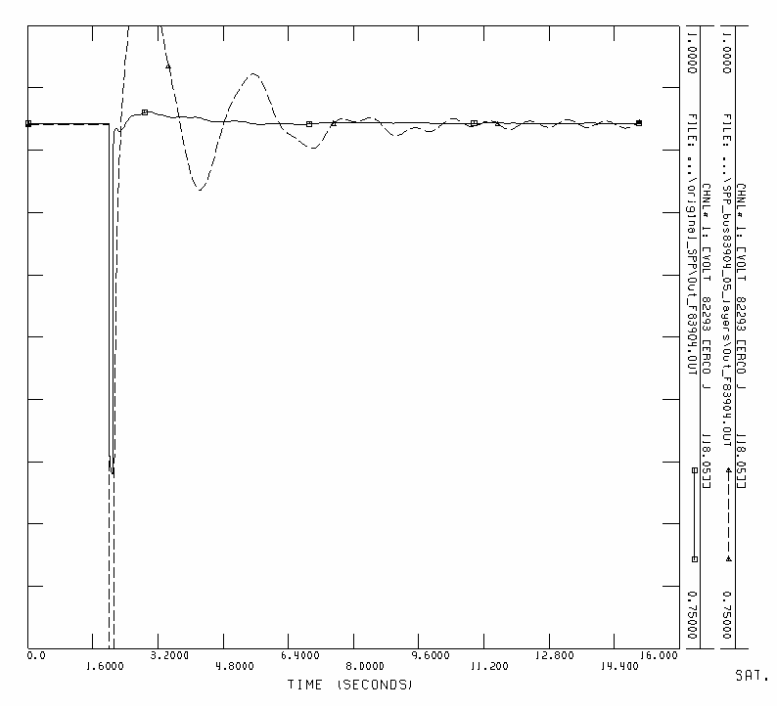



Figure 4.12 The voltage profile comparison on the Bus 82293

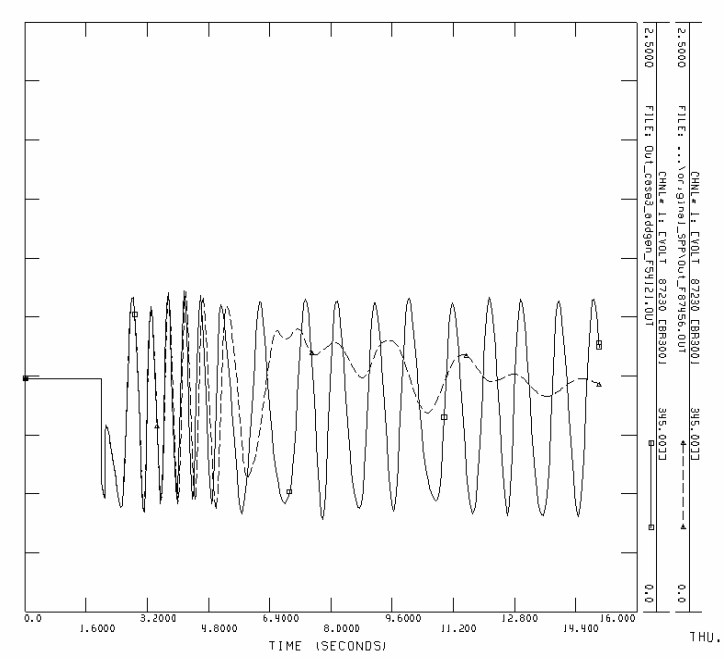



Figure 4.13 The voltage profile comparison on the Bus 87230

Table 4.3 Comparing the stability simulation results between the partitioning system and the whole system for single fault

| How many layer of buses near the fault? | 30 | 15 | 8 | 5 |
|---|---|---|---|---|
| The fault in load bus 54121 | O | O | V | V |
| The fault in generator bus 87456 | O | V | X | X |
| The fault in load bus 64895 | O | O | V | V |
| The fault in load bus 31426 | O | O | O | O |
| The fault in generator bus 25044 | O | O | O | O |
| The fault in load bus 83904 | O | V | V | X |

O: The simulation results for the partitioning system are almost the same as those for the whole system.

V: The simulation results for the partitioning system are similar with those for the whole system.

X: The simulation results for the partitioning system are different from those for the whole system.

The simulation results indicate that the safest method is to choose 30 layers of buses near the fault in the SPP system. The simulation results for the partitioning system are almost the same as those for the whole system in every case. However, only partitioning eight layers of buses would produce a similar simulation result as for the whole system if the fault is far away from boundary.

Table 4.4 Comparing the simulation time in the fault on bus 25044 and 87456

| | Simulation time (sec) | |
|---|---|---|
| The fault location | Bus 25044 | Bus 87456 |
| Whole System | 338 | 654 |
| 30 layers | 337 | 251 |
| 15 layers | 71 | 61 |
| 8 layers | 24 | 15 |
| 5 layers | 14 | 14 |

*4.4.2. Multiple faults scenarios*

As described below, three multiple faults cases are created to study two distant faults and two near-by faults scenarios.

Case 1: Two three-phase short circuit faults occur simultaneously in load bus 54121 and generator bus 87456 as explained in figure 4.14.

Case 2: Two three-phase short circuit faults occur simultaneously in load bus 64895 and load bus 83904 similar to figure 4.14.

Case 3: Two three-phase short circuit faults occur simultaneously in load bus 31426 and generator bus 25044 as explained in figure 4.15.

The whole files for simulation list below.

1_ts05s2x_ori_solve.sav: The system raw data was solved to an accepted level.

2_nerc-all-NO-SPP.dyr, 2_Ts-05SP.dyr: The system dynamic model.

3_PowerFlow_CONLG.py , 3_CONVLOADS-GENS.idv : Use python program to cut the system. Then use idv file to run power flow in dynamic environment.

4_SPPsystem_initial.idv: No fault test to check data integrity.

5_SPPsystem_F54121_87456.idv: Two faults simultaneously happen at load bus 54121 and generator bus 87456.

6_SPPsystem_F64895_83904.idv: Two faults simultaneously happen at load bus 64895 and load bus 83904.

7_SPPsystem_F31426_25044.idv: Two faults simultaneously happen at load bus 31426 and generator bus 25044.

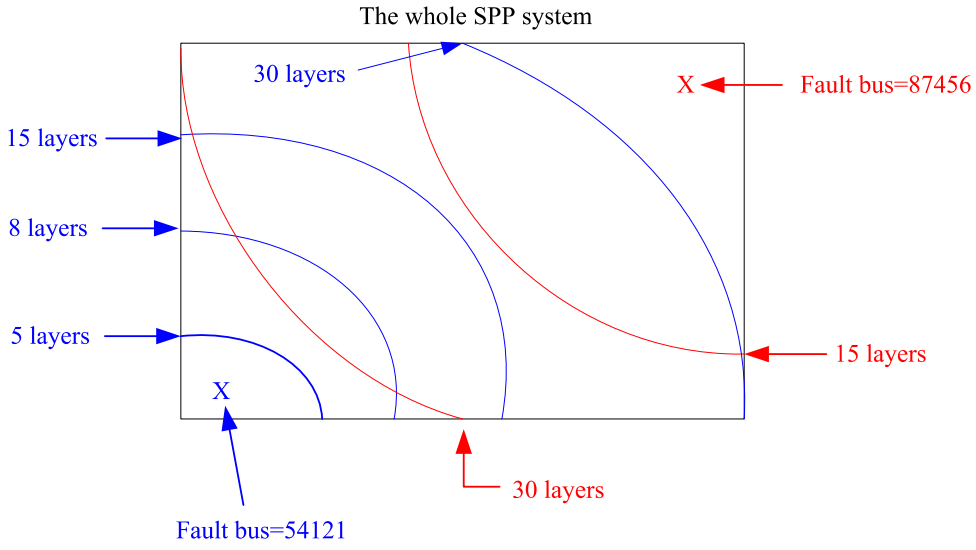



Figure 4.14 The sub-systems in case 1

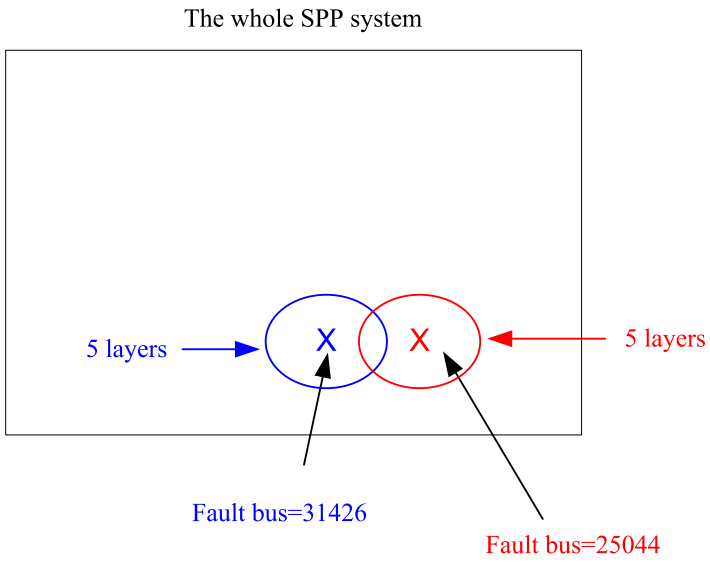



Figure 4.15 The sub-systems in case 3

Table 4.5 Comparing the stability simulation results between the partitioning system and the whole system for double faults

| How many layer of buses near the fault? | | Scenarios | Compare with the original Whole system |
|---|---|---|---|
| Bus 54121 | Bus 87456 | | |
| 5 | 15 | distant | V |
| 8 | 15 | distant | V |
| 15 | 15 | distant | V |
| 30 | 30 | near-by | O |
| Bus 64895 | Bus 83904 | | |
| 5 | 8 | distant | V |
| 8 | 8 | distant | V |
| 15 | 15 | distant | V |
| 30 | 30 | near-by | O |
| Bus 31426 | Bus 25044 | | |
| 5 | 5 | near-by | O |
| 8 | 8 | near-by | O |
| 15 | 15 | near-by | O |
| 30 | 30 | near-by | O |

The simulation results are shown in Table 4.5. For example, the simulation result for near-by scenario is almost the same as those for the whole system as shown in Figure 4.16. The examples of distant scenarios are the same as single fault scenarios. The comparison of the single and double fault test results shows that the double faults correspond to the single fault. Even though the fit is not 100%, the subsystem simulation is similar to the original whole system.

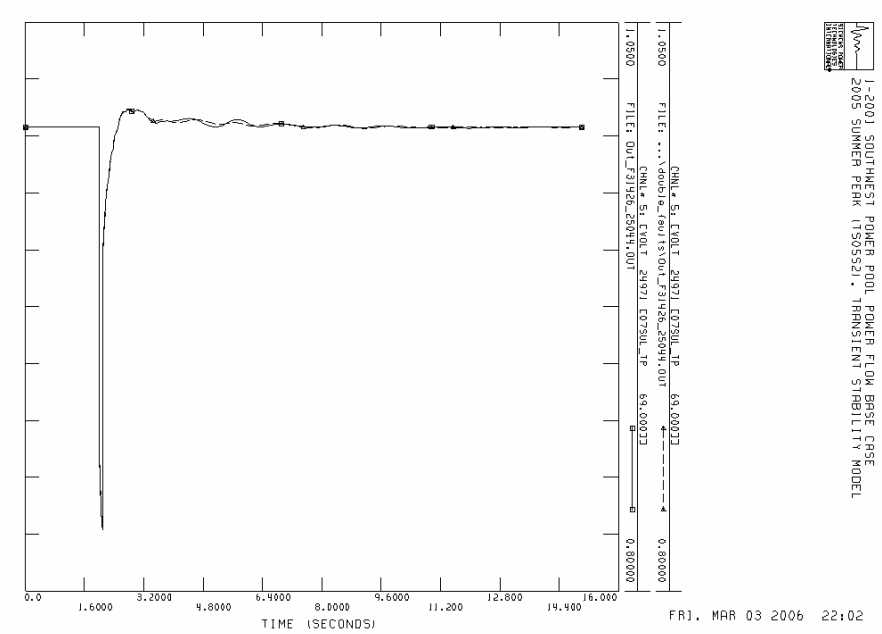



Figure 4.16 The voltage profile comparison on the Bus 24971

# CHAPTER 5

# INTEGRATED SOFTWARE ENVIRONMENT

## 5.1 Introduction

Despite the broad body of research on accelerating dynamic stability simulation, few applications exist for industrial systems because it is difficult to apply developed methods with commercially available software. Since the power system simulator for engineering (PSS/E), one of the most commonly used system analysis programs among industry, provides an embedded Python interpreter, this situation can be improved. The Python program not only executes the PSS/E functions but also can easily be extended using the code written in other languages, for example MATLAB syntax in our application. Therefore, it can reduce the time for research and development. However, the dynamic functions for Python syntax do not work in version 30 of PSS/E, the user has to control the process with batch commands. Figure 5.1 outlines the proposed approach in which the Python program is in control from the beginning until the choosing of the swing bus, and then, once the dynamic models are added, the batch commands control the remaining processes.

In order to allow the proposed algorithm be smoothly applied to the other power system by other users, all programming codes have been integrated with the graphical user interface in the Python program. The user only needs to choose his/her desired

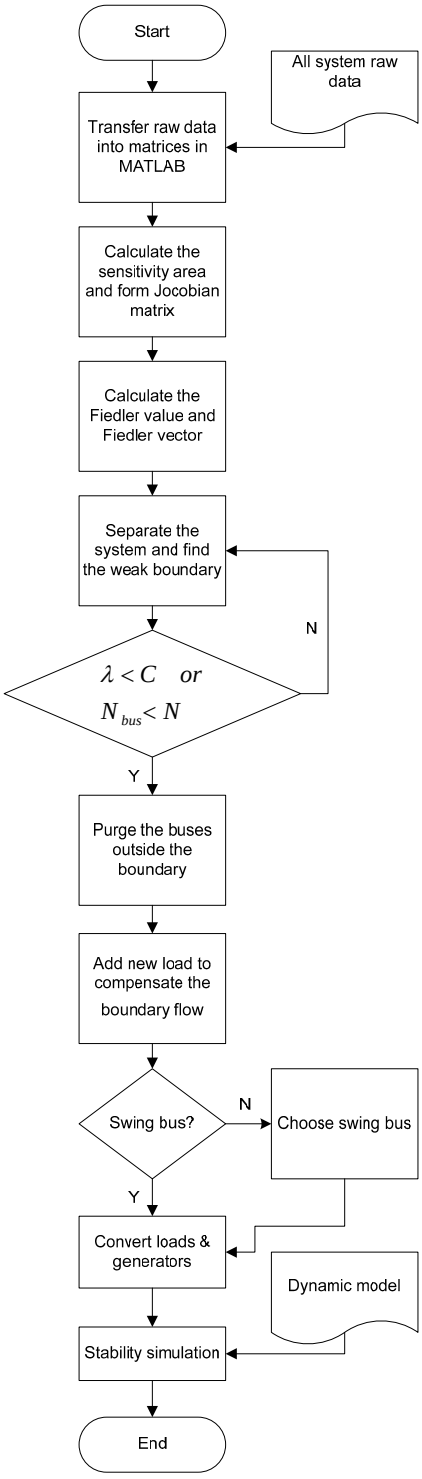



Figure 5.1 The flowchart of simulating a single fault transient stability in a large-scale power system

function. The procedure will automatically give the commands to PSS/E or link to Matlab for complex computation, and then produce the simulation results of the dynamic stability.

## 5.2 Graphical User Interface

An interface needs to be developed between Python and Matlab language because Python programs and statements can only be executed within the PSS/E user interface, while the proposed algorithm is developed in Matlab. Python language was initially developed in a UNIX-compatible environment; therefore, the entire Python source code can be downloaded free of charge. When users require certain special functions, similar functions that allow modifications for specific applications may be downloaded. Because the PSS/E version 30 uses the Python version 2.3 for its interface, an interface between Python 2.3 and Matlab is not available. The users can download a similar source code, pymat [31], which provides the interface source code between Python 2.0 and Matlab 5. In order to develop an interface between Python 2.3 and Matlab 6.5, the setup process in Pymat can be modified in Microsoft Visual Studio 6.0 and the source code should be compiled again. This is explained as follows.

Objective: Building PyMat.pyd, an executable interface file in Python, from PyMat.cpp, the PyMat source code, using Microsoft Visual Studio 6.0.

Situation: 1. At present, users have only PyMat.cpp and have to create the corresponding Visual Studio project from scratch.

2. PyMat.cpp is for Python 2.0 and Matlab 5 and need to modify it for Python 2.3 and Matlab 6.5.

The operating process in the Visual Studio:

1. Create a new MFC DLL project.
2. Delete all automatically created files
3. Add PyMat.cpp to the project
4. Disable the use of pre-compiled headers
5. Additional Include Directories: Python23\include & Matlab65\extern\include
6. Additional Library Directories: Python23\libs & Matlab65\extern\lib\win32\microsoft\msvc70
7. Additional Dependencies: libeng.lib & libmx.lib
8. Mark initpymat as DLL export function using __declspec( dllexport ).

Special code adjustments for Matlab 6.5:

Some functions for the communication with Matlab have changed from Matlab 5 to Matlab 6.5 and have to be changed (engGetArray → engGetVariable, mxSetName & engPutArray → engPutVariable).

Finally the resulting PyMat23.dll has to be renamed to pymat.pyd and placed into Python23\Lib\site-packages. [33] The purpose of this procedure is to create an executable interface file for Python 2.3 and Matlab 6.5 and put this file into the automatically imported directory in Python.

On the other hand, users should also install Numeric-23.8.win32-py2.3.exe [32] to transfer arrays between the Python language and Matlab because Numerical Python will add a high-speed and sophisticated array facility to the Python language. The

purpose of this installation is to expedite the data transfer because one-dimensional arrays can be communicated directly between Python and Matlab. After setting up the simulation environment, test-pymat.py [34] can be used to test whether the Python to Matlab interface works well.

Because of the graphical user interface, users are required to run Python IDLE directly from within the PSS/E user interface using the I/O Control>Run Program Automation File option. [35] In the Python IDLE environment, which provides an interactive window, the interface for the proposed algorithm can be executed as shown in Figure 5.2.

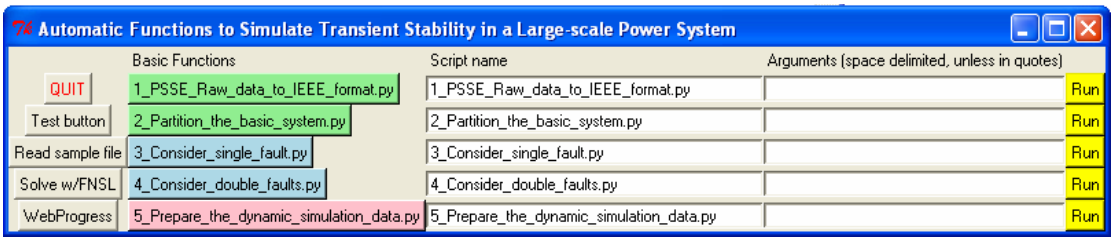



Figure 5.2 The graphical user interface for the proposed algorithm

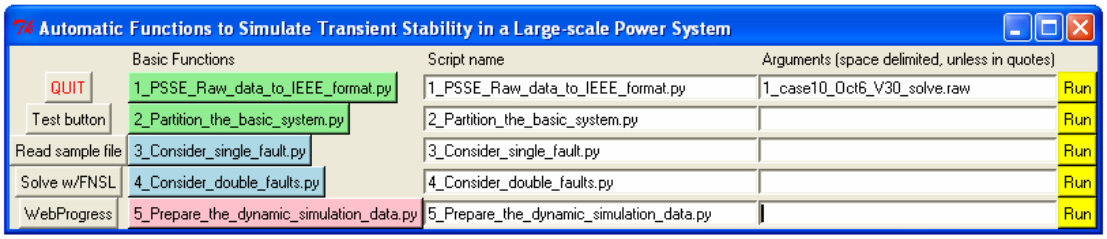



Figure 5.3 Transfer PSS/E raw data to IEEE format

For a new stability case, users are required to sequentially run the basic functions 1–5. The control mode in Figure 5.2 is a simple batch mode. In this mode, the commands and data that would normally be provided by the user are typed into the argument list shown in the Figure 5.2. The following part will explain how to execute every function.

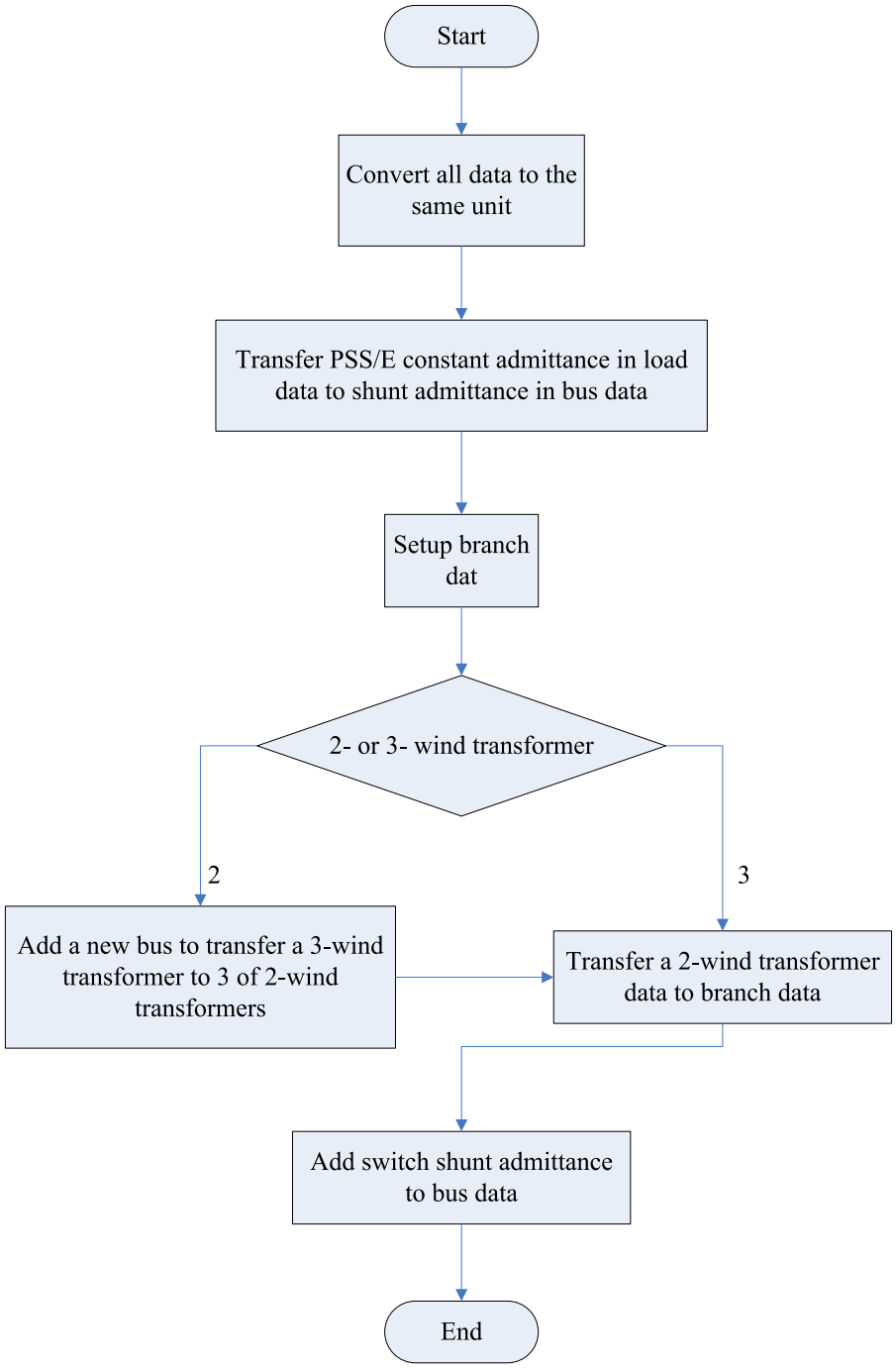



Figure 5.4 The flowchart of transformation from PSS/E raw data to IEEE format

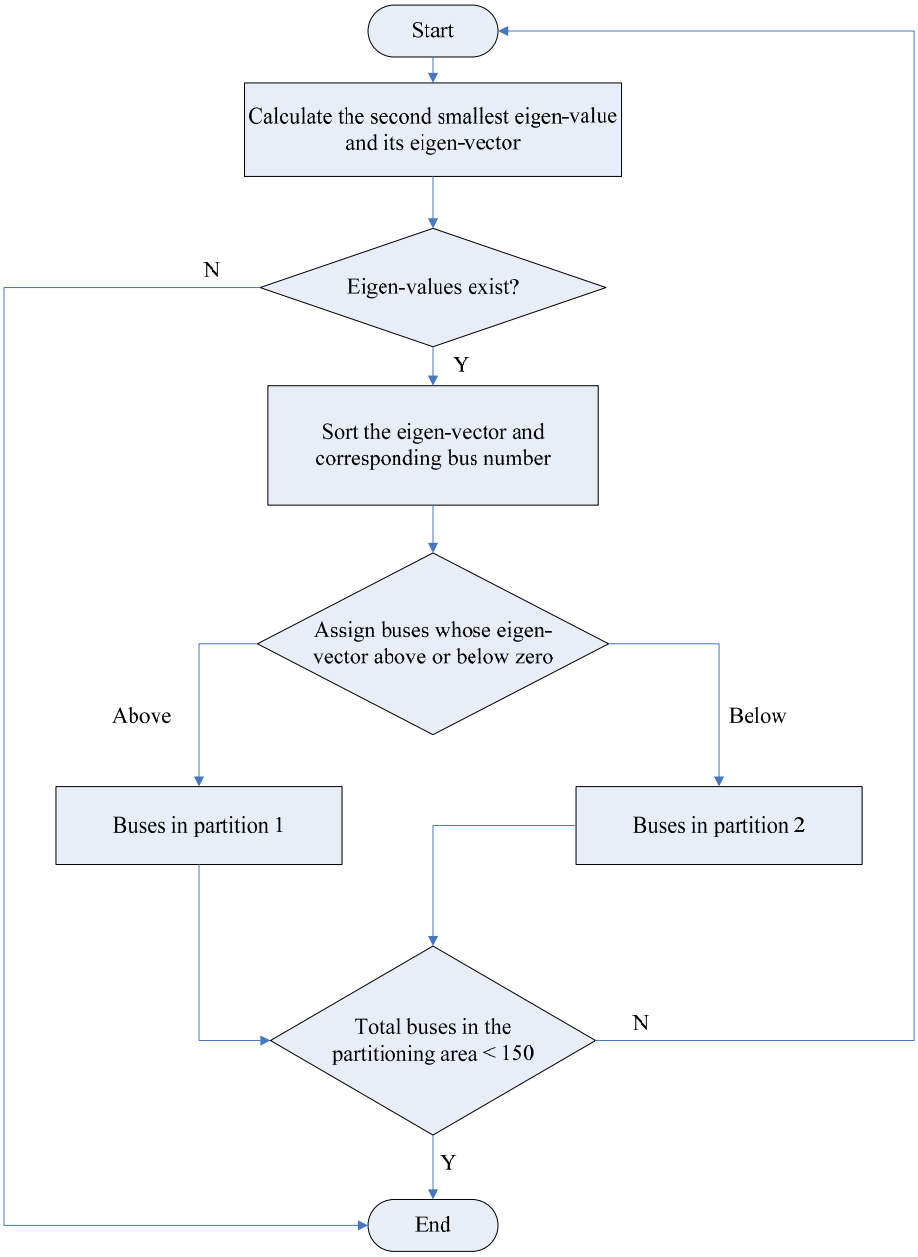



Figure 5.5 The recursive spectral graph partitioning algorithm

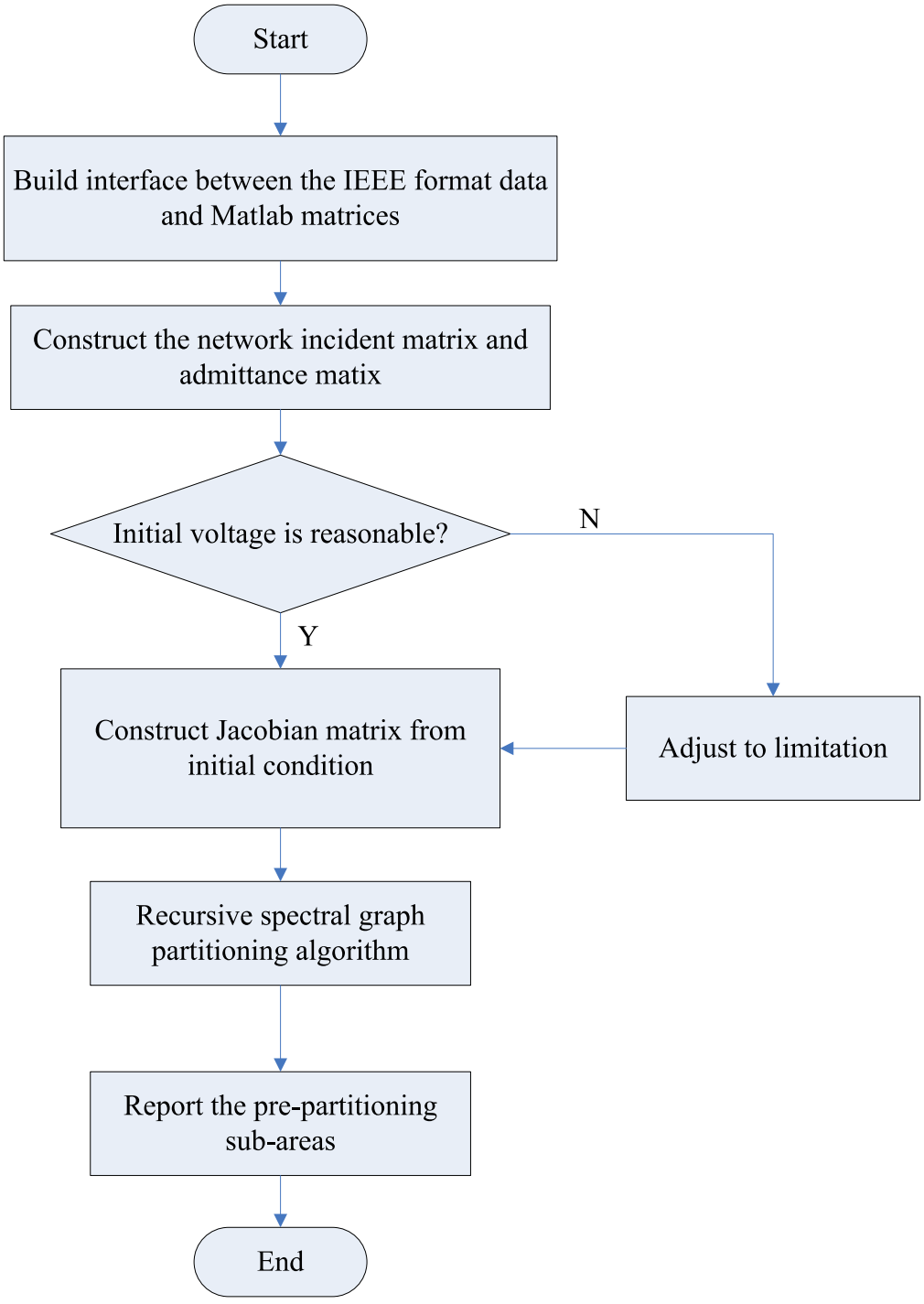



Figure 5.6 The whole process of the proposed partitioning algorithm

Basic function 1 transforms the PSS/E raw data to IEEE format. Although PSS/E has provided a function to transform the raw data to IEEE format, it is unable to run if the bus number exceeds 9999. Because the calculation in the proposed algorithm uses only some of the parameters in the raw data, the program extracts only the necessary parameters and transfers them to the IEEE format. The flowchart of function 1 is shown in Figure 5.4. The overall sequence of this step in transferring new power flow data is listed below:

1. Click basic function 1 directly to run the default case.
2. Assign the file name for the raw data and then click the yellow "run" button to run the assigned cases; for an example see Figure 5.3.

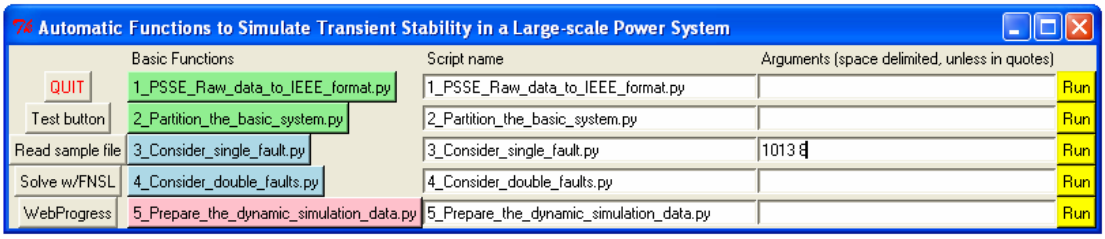


Figure 5.7 The fault occurs in bus 1013

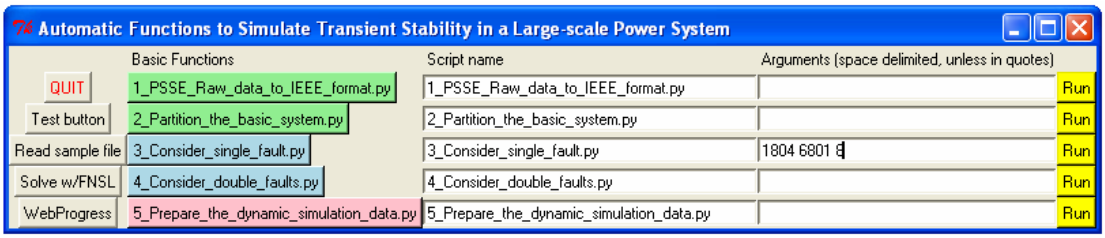


Figure 5.8 The fault occurs in the branch

When executing function 1, the program will automatically export the IEEE format file with the default file name case_sample; therefore, users simply click on basic function 2 to run the partitioning algorithm (flowcharts shown in Figures 5.5 and 5.6). It should be emphasized that function 1 and 2 can be executed in advance to save

the online computational time. For the same system, this calculation is required to be performed only once.

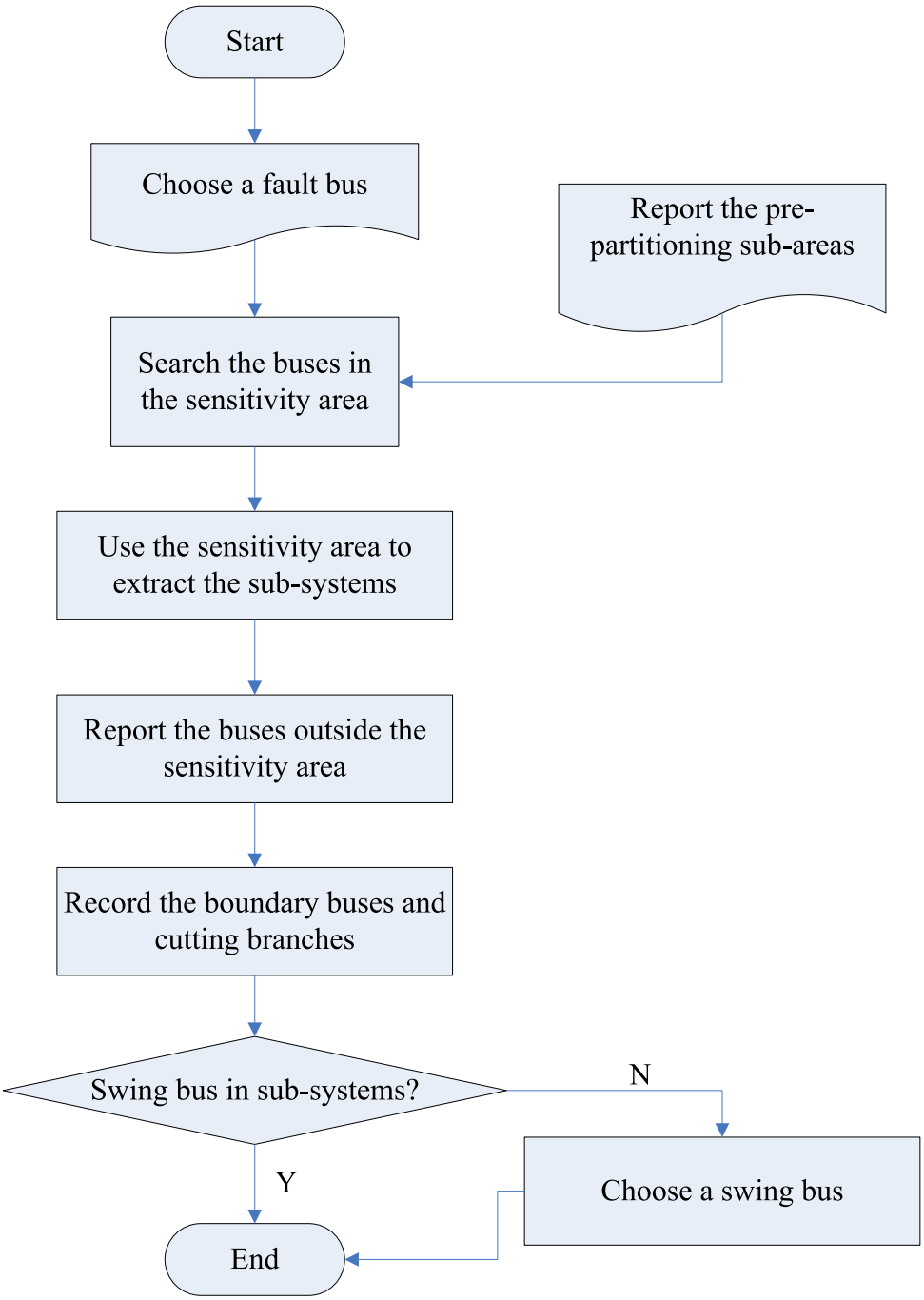



Figure 5.9 The flowchart of single fault process

When a fault occurs, identify the fault bus and layers to function 3. For example, in Figure 5.7, the fault bus is 1013 and 8 layers are chosen by the user, and in Figure 5.8, the fault occurs in the branch between bus 1804 and bus 6801 and 8 layers are chosen by the user. Subsequently, click on the yellow "run" button to execute the function. The flowchart of function 3 is shown in Figure 5.9.

When double faults occur, assign the buses for double faults and layers to function 4. For example, in Figure 5.10, the faults occur in bus 1013 and bus 8704 and 8 layers are chosen by the user. "-1" is to separate the two faults. In Figure 5.11, one fault occurs in the branch between bus 1804 and bus 6801 and the other fault occurs in bus 1013; 4 layers are chosen by the user. After assigning the faults, click the yellow "run" button to execute the function 4 (flowchart shown in Figure 5.12).

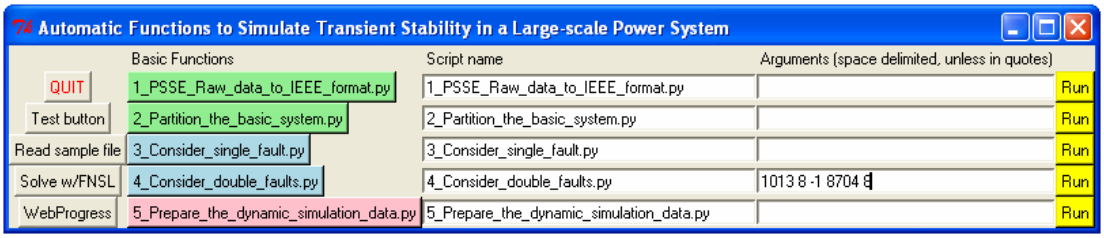


Figure 5.10 The double faults occur in two buses

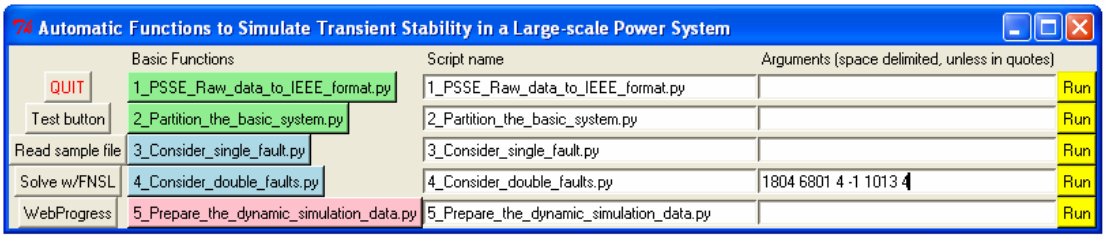


Figure 5.11 The double faults occur in a bus and a branch

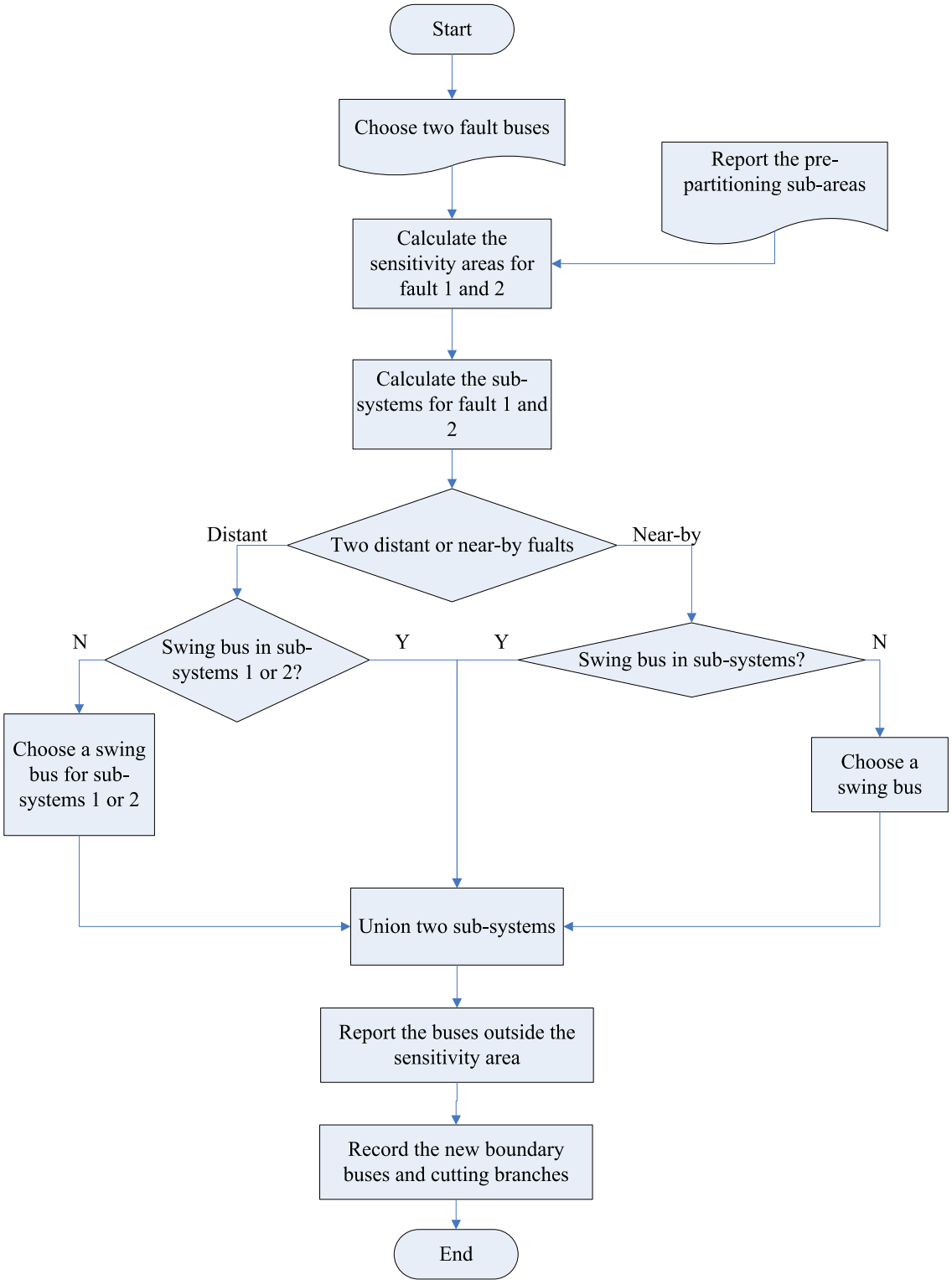



Figure 5.12 The flowchart of double fault process

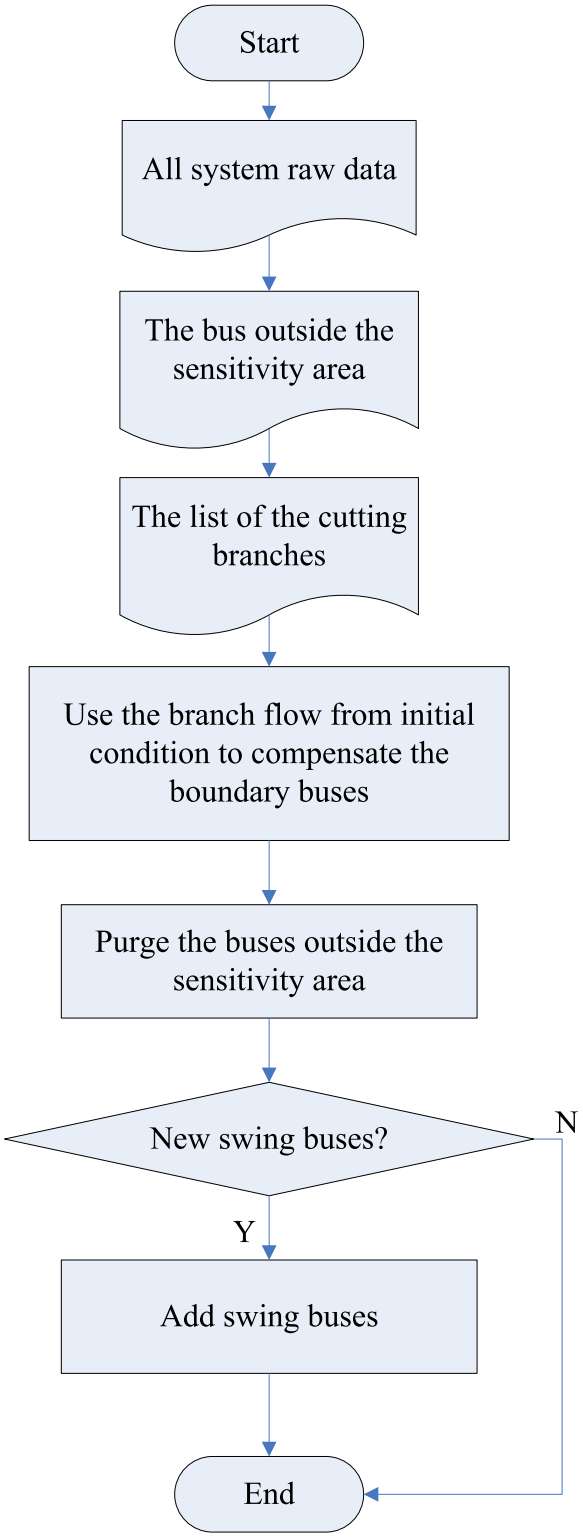



Figure 5.13 The flowchart of preparing the dynamic simulation data

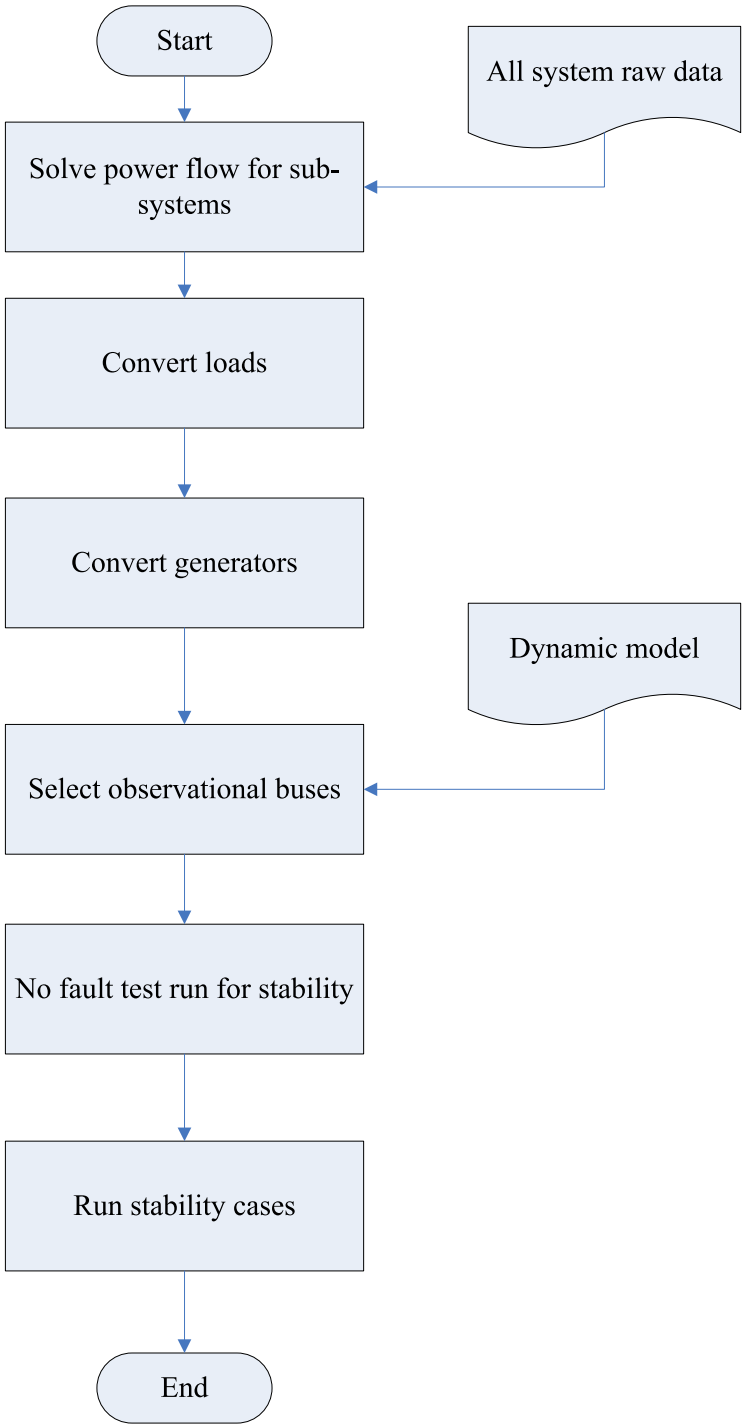



Figure 5.14 Simulation in PSS/E dynamic environment

Because the dynamic environment in PSS/E is not available for external interfacing, all the files for dynamic simulation will be prepared in a power flow environment to simplify the work. Users are only required to click basic function 5 and the program will automatically prepare all of the input files for the dynamic simulation. The flowchart of function 5 is shown in Figure 5.13.

The dynamic simulation interface is operated as a separate program, and independent from the new PSS/E interface so the Python program cannot control the dynamic environment. However, the users can use the response files (.IDV) instead. Before running the transient stability program, some preparation for network matrices and the conversion of loads and generators are required to be completed first. Users can use the internal functions to convert the generators and loads in the PSS/E dynamic environment. For example, run 6_CONVLOADS-GENS.IDV. When running the same system, this step will be performed only once. Finally, various dynamic simulations can be executed. For example, run 7_Transient_gen1013_fault_4channel.idv. The flowchart in the dynamic environment is shown in Figure 5.14.

## 5.3 The Double Check Boundary Method

After dynamic simulation, in order to guarantee the accuracy of simulation results, the user can use the verification procedure to double check the results. This test procedure will start from an area of eight layers of sensitivity, and then increase the layers to 15. After comparing the differences in terms of both simulation results, if the difference is huge, the program will increase the layers again. If the difference is smaller than a user setting, the program will decrease the layers as shown in figure 5.15.

The program will suggest whether the simulation result is accurate, and also draw the error between the suggestion layers and the compared nearby layers, as shown in figure 5.16 and 5.17. The Y-axis in both figures indicates the voltage in per unit. The average error in both figures is less than 0.05. Then users can decide whether this is accurate enough.

On the other hand, because there is no window interface in dynamic simulation and the output file is a binary file, users also need to install a new utility, PSSECHOP, to view Channel Output files. [36] This new utility can convert a binary output file to an Excel file which can be read by other programming languages.

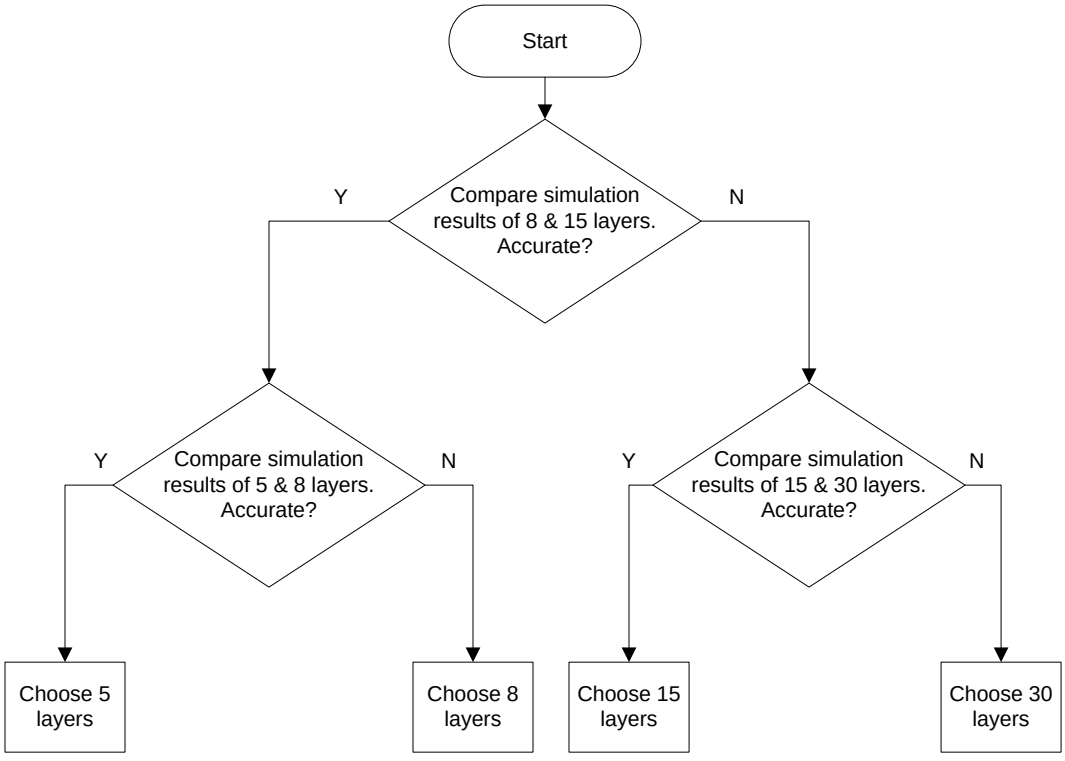



Figure 5.15 The verification procedure to double check the simulation results.

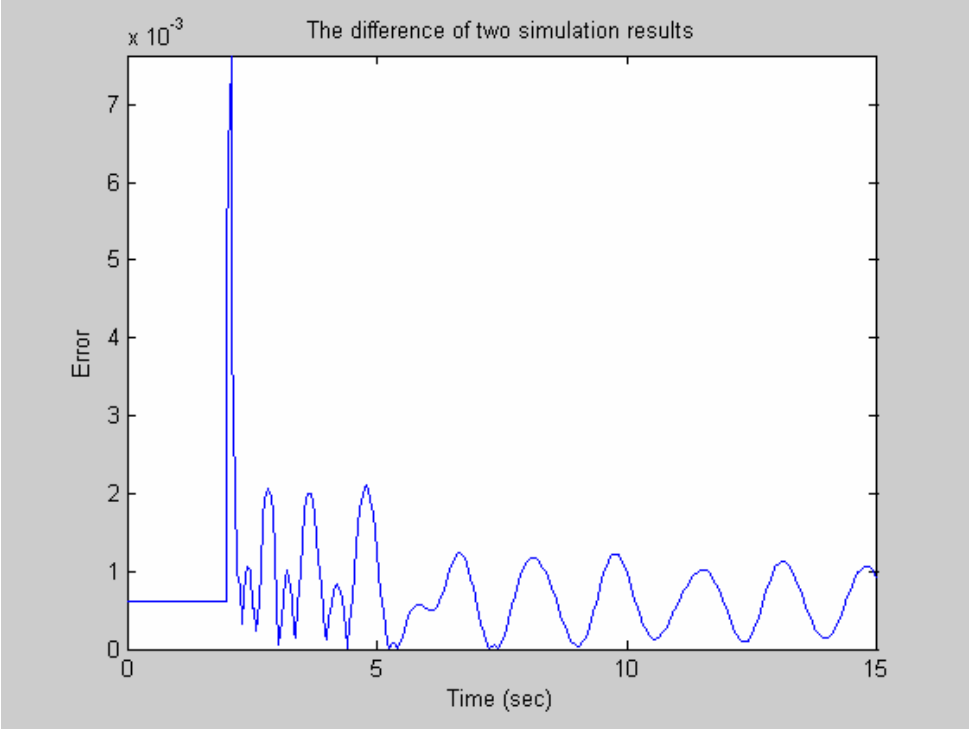



Figure 5.16 The example of error in one fault case.

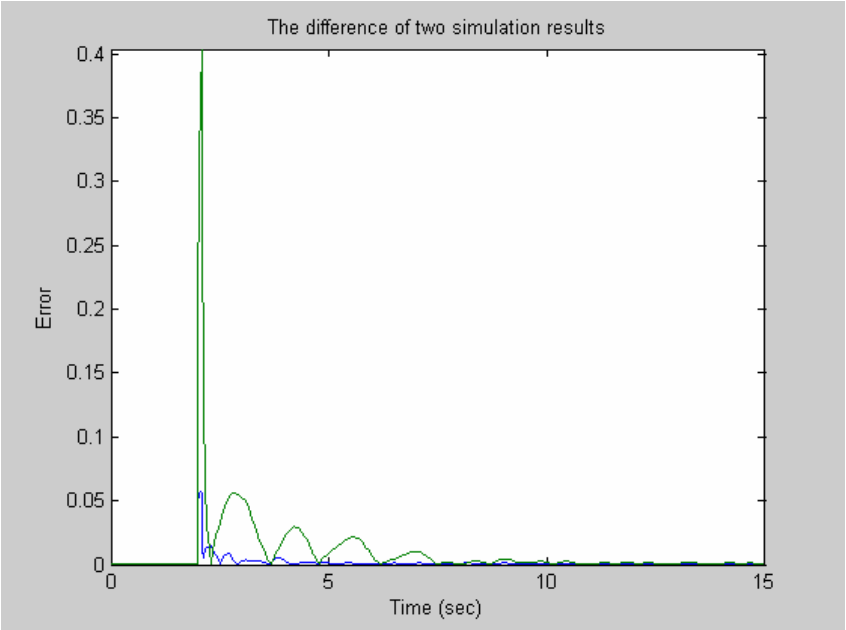



Figure 5.17 The example of error in double fault case.

# CHAPTER 6

# SUMMARY AND CONCLUSION

## 6.1 Summary and Conclusion

After the deregulation, in the face of free competition, many transmission lines will be fully loaded in order to maximize profits. On the other hand, in free trade, power flows are changed dynamically. All these factors threaten power system stability. In addition, the demarcation boundaries among areas have become increasingly fuzzy due to high-volume cross-boundary transactions. Therefore, traditional company-based dynamic studies are not suitable for this operation paradigm and a novel method is needed to investigate the stability of large scale interconnected systems. The Northeast blackout of August 14$^{th}$, 2003 emphasizes the importance of investigating power system stability. Because prevention is better than cure, carrying out rapid simulations can provide early warnings in the event of similar large blackouts. The proposed approach can rapidly simulate multifault stability problems. When the system is operated under local impact, the simulation results can evaluate system stability in order to prevent the similar major blackouts from recurring.

Because the usage of dynamic equivalency or parallel processing is also difficult due to complicated mathematical algorithms and software/hardware requirements, this dissertation proposes novel simulation approaches for partitioning

power systems and aggregating pre-divided sub-areas to perform a dynamic stability study of a large-scale interconnected power system, such as those in North America. The most significant feature of these approaches is that they can simulate large-scale power systems without recalculating equivalent circuit model. These approaches differ from the current simulation process in regard to each ISO as the boundary and place the equivalent circuit outside the boundary. The proposed technique focuses on faults as the center of the sub-systems and re-combines pre-divided sub-areas to form a new boundary. This technique can quickly form sub-systems and rapidly obtain simulation results. Because the proposed algorithm focuses on the faults as the center of whole systems, simulation results will theoretically have increased accuracy. In conclusion, the proposed scheme is simpler than other methods and generates rapid simulation for single or multiple faults. Test results for the SPP system indicate that this proposed partition method can reach the goal of rapidly investigating a large-scale power system.

According to the simulation results, the computational time of the proposed algorithm appears to be relatively low in comparison with the computational times simulated in the whole system especially when a lot of cases have been analyzed in the same systems. In addition, the computational time of the proposed algorithm will not increase with respect to the increase in the scale of the systems. Therefore, the proposed algorithm is suitable for large-scale systems.

The simulation process of power flow and transient stability is developed with an object-oriented programming (OOP) language that supports the user-friendly graphical user interface, complicated calculation and extended interface with other

software. These bring benefits for large-scale systems that require frequent alteration of the input data and integrating software together can reduce the time of software development.

## 6.2 Possible Future Researches

For graph theory application, some software executables, including CHACO [37], METIS [38], JOSTLE [39], PARTY [40] and SCOTCH [41], are freely available for academic and research purposes. While developing the software in this dissertation, METIS has been utilized to partition power systems; however, METIS is an independent executable software program that requires a fixed input and output format and was developed for parallel computing applications. The algorithm in METIS is not suited to the proposed method; however, METIS was useful for debugging raw data when raw data was transferred to the IEEE format. As MATLAB cannot accommodate infinite values in a matrix, to identify the factor causing infinity, the following process shows how METIS help debug the data.

Because METIS is a set of programs for partitioning graphs, a sparse matrix or graph can be separated into equal parts. The Jacobian matrix for an SPP system can be divided into 6 parts using METIS; each part contains roughly 5,000 buses. This study calculated the determinant of each small part and identifies terms related to graph points in a matrix are infinity. Debugging these points in raw data in a power system, the power flow calculation can be continued.

Although the dissertation did not use these software programs for source code, based on the development and debugging processes, the graph theory will be easily

applied in power systems if the interface between Python language and graph partitioning software can be established and a data format and database to standardize file input and output can be unified. Particularly, various partitioning methods such as the multilevel recursive-bisection, multilevel k-way, and multi-constraint partitioning schemes, can be applied to power systems and determine which part can accelerate partitioning of the power system.

As generation and load in power systems are always changing in real time, they are very similar to the situation in the previous minute, static partitioning will not need to be calculated every minute when dynamic load balance is applied. The dynamic load balance will finely tune the previous result and make on-line analysis fast. The optimal situation is to establish a well-designed interface program and libraries that can execute different programming codes, applications, and algorithms. The second significant point is that different software programs can use subroutine calls instead of file transfers to speed up simulation. When all of the above work can be done, graph theory and other algorithms can easily be tested and compared in terms of performance.

Finally, it should be noted that a database of the complete interconnected system should be developed in the future. The proposed approaches can then be merged with short-term load forecasts in order to partition power systems in advance. The simulation speed can be expected to approximate real time. Due to the improvement in response time, the prevention measures will improve. Graph partitioning can also be applied to parallel computing, and the pre-divided subareas in the proposed approach can be combined with parallel computing in order to rapidly solve transient stability problems

and obtain a complete solution using numerous computers. The approaches proposed in this dissertation can be expected to become important tools of real-time simulation in the future.

# APPENDIX A

# THE DETAIL SIMULATION RESULTS OF THAILAND SYSTEM

1. The stability simulation step for Thailand system:

1_case10_Oct6_V30.raw: The system power flow raw data.
2_Machine_DataY04.dyr: The program will automatically add some new generator models to compensate the boundary flow in separated system.
3_PowerFlow_CONLG.py : Use python program to divide the system and run power flow.
4_Dynamic_Initial.idv: No fault test. Just test whether the initial condition is stable.
5_Transient_gen1013_fault.idv: The three-phase short circuit fault in generation bus 1013.
6_Transient_gen1013_fault&out.idv: After the fault, trip the generator in bus 1013.
7_Transient_gen7035_fault.idv: The three-phase short circuit fault in generation bus 7035.
8_Transient_gen7035_fault&out.idv: After the fault, trip the generator in bus 7035.
9_Transient_Line_fault&trip.idv: The three-phase short circuit fault in the transmission line between bus 1804 and 6801. After the fault, trip the line.

2. Compare the simulation result of step 5-9.

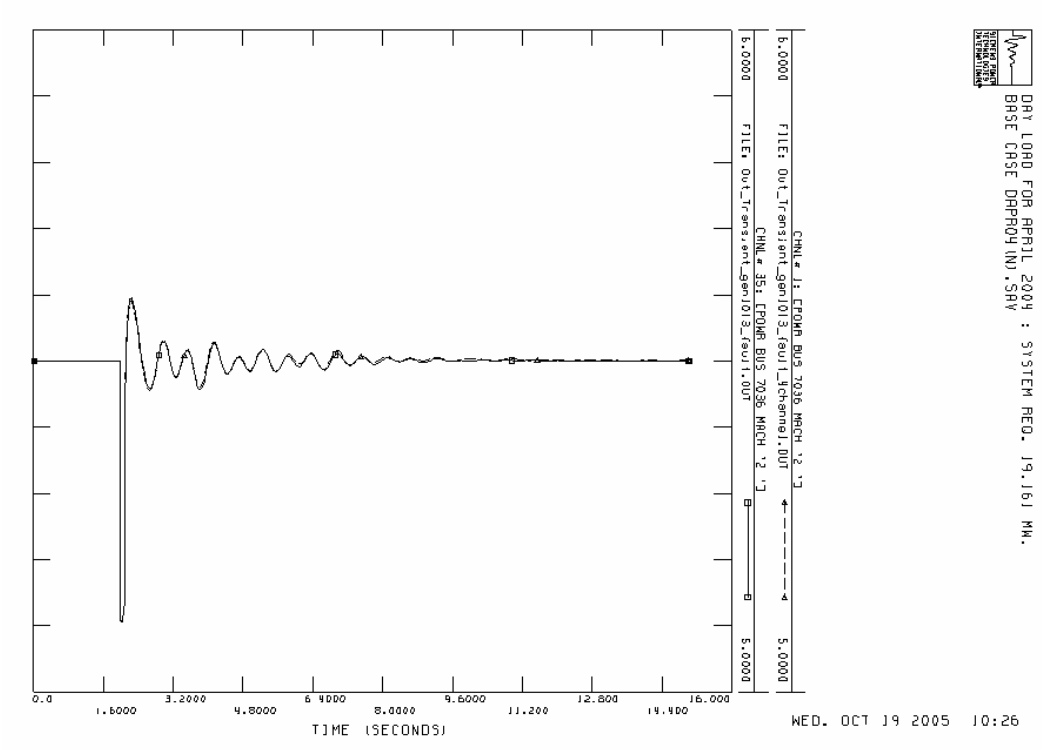
| How many layer of buses near the fault? | 10 | 8 | 7 | 6 | 5 |
|---|---|---|---|---|---|
| Step 5 | O | O | V | V | X |
| Step 6 | O | O | V | V | X |
| Step 7 | O | O | V | O | V |
| Step 8 | O | O | V | O | V |
| Step 9 | O | O | V | V | X |

O: The simulation results for the partitioning system are almost the same as those for the whole system.
V: The simulation results for the partitioning system are similar with those for the whole system.
X: The simulation results for the partitioning system are different from those for the whole system.
The detail simulation result lists from next page.

3. Step 5
Case 1: For 10 layers of buses near the fault:

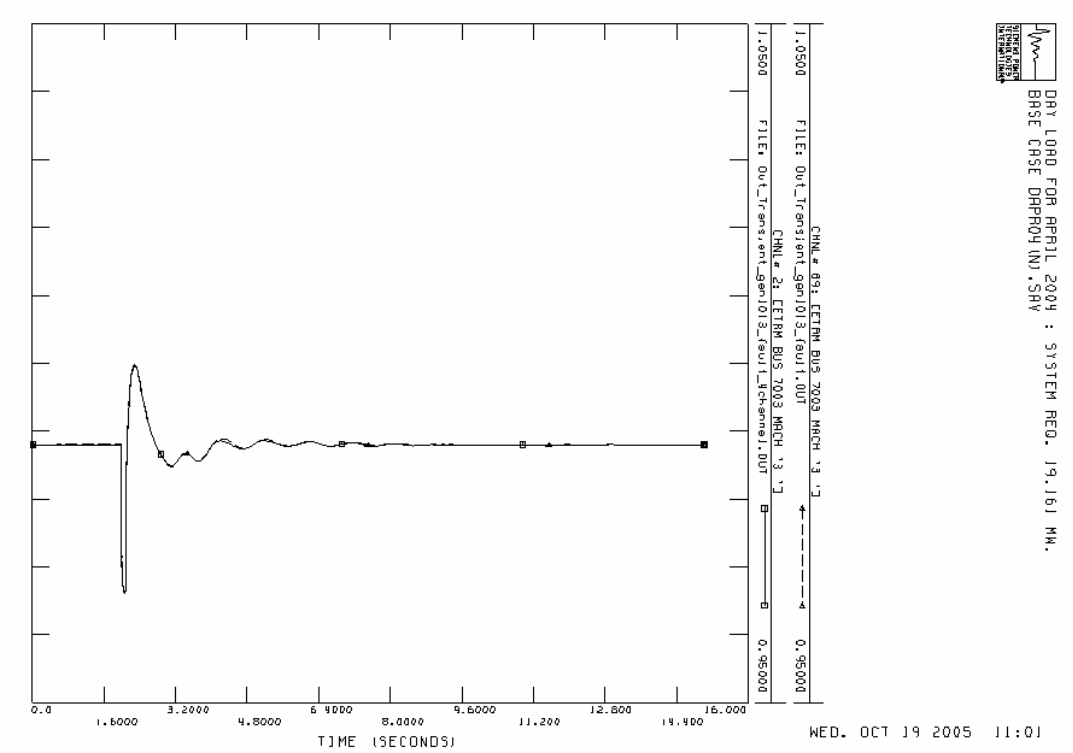
```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17568 MW   7602 MVA
Total generation of the separated system: 18008 MW   4680 MVA
```

The simulation result is almost the same as the original system.
The compared graphs list below.
Power: Bus 7036 “2”

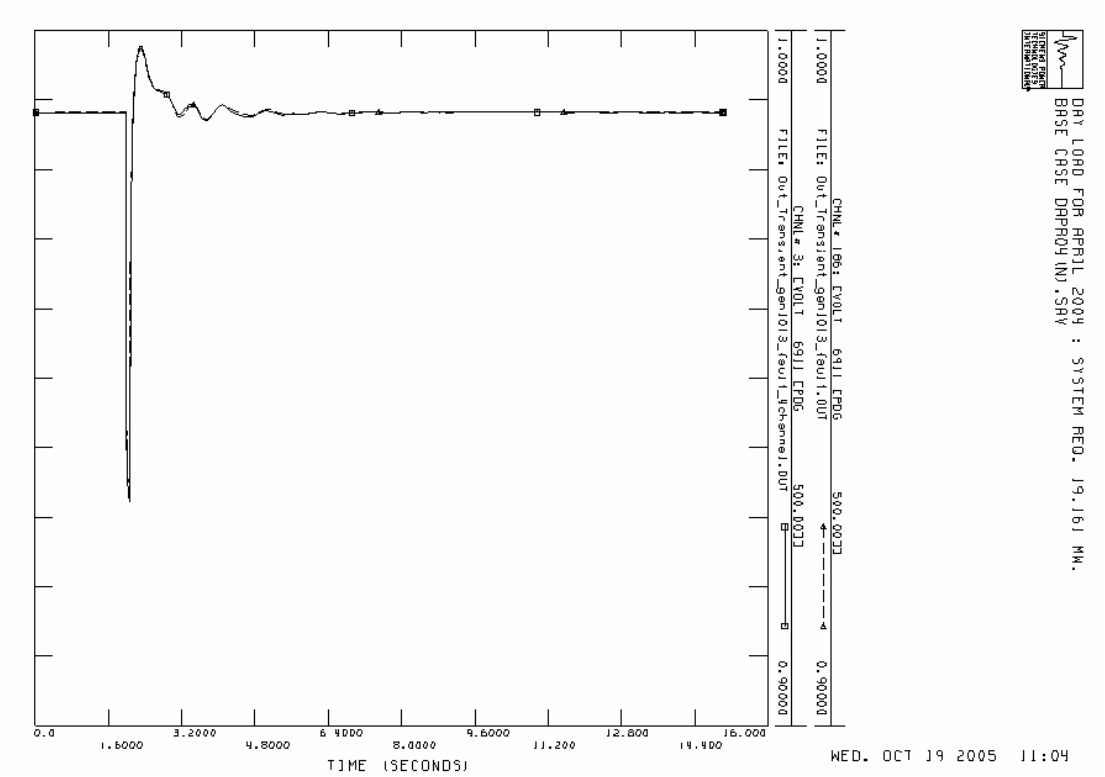



E

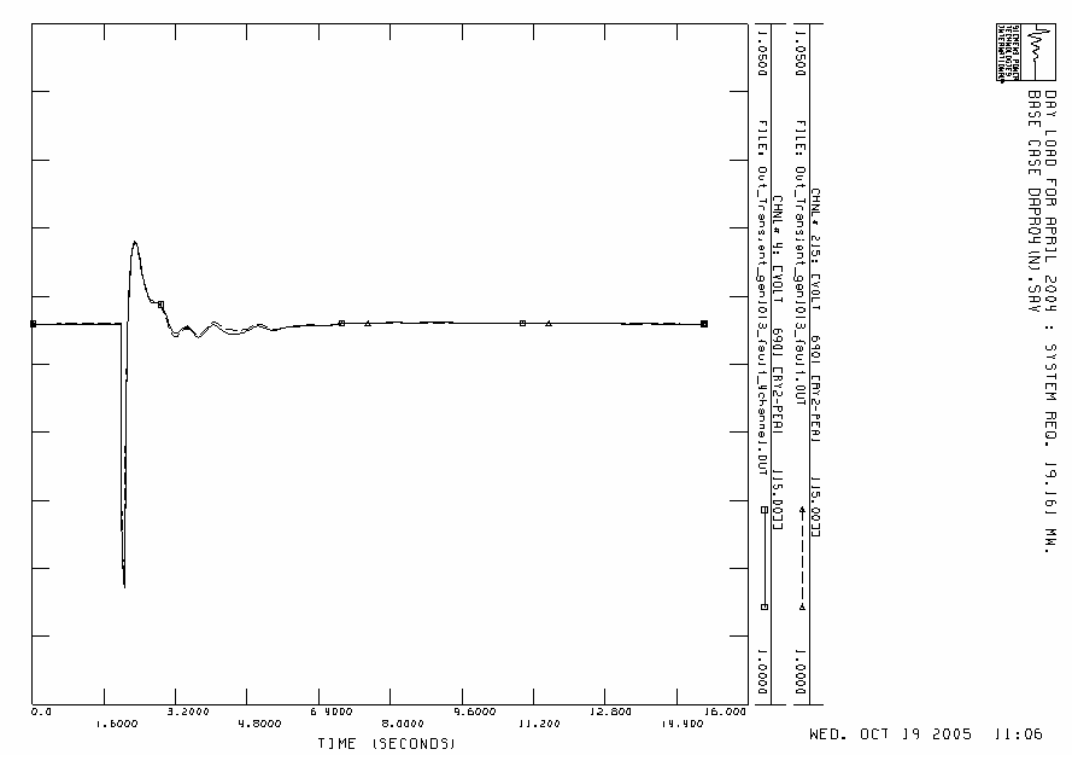
TRM: Bus 7003 “3”



Voltage: Bus 6911




Voltage: Bus 6901



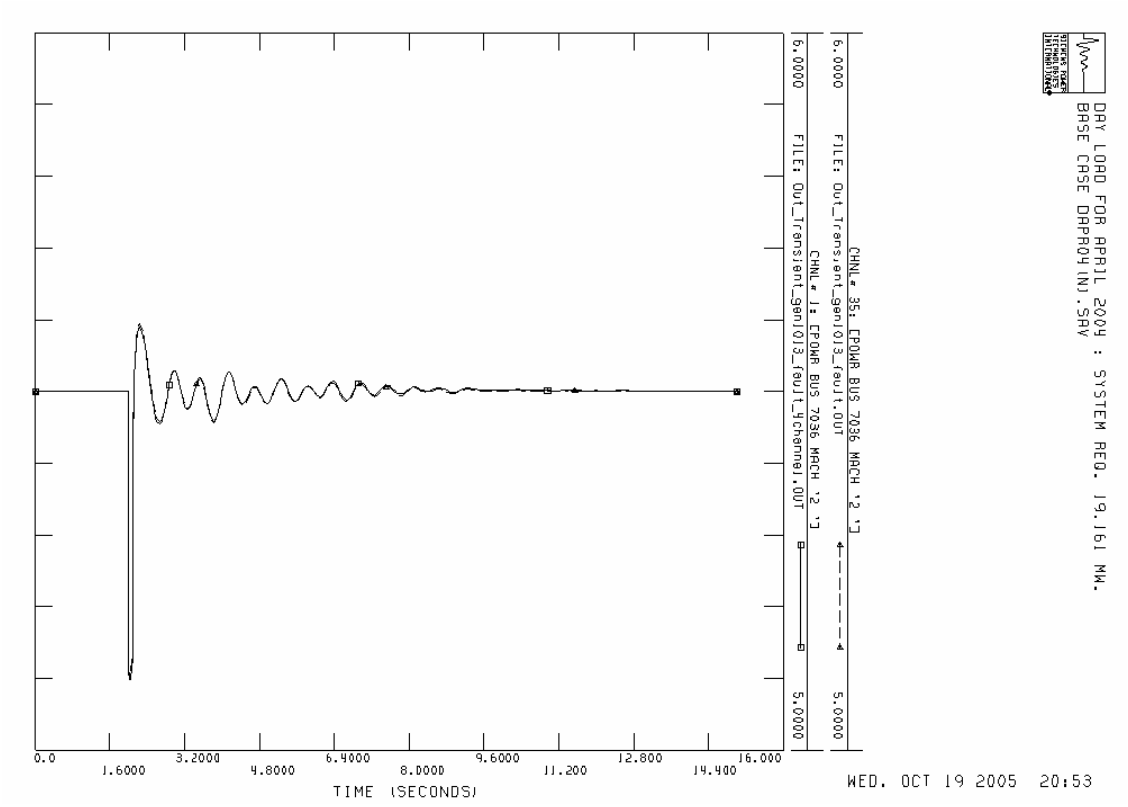
Case 2: For 8 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17432 MW   7471 MVA
Total generation of the separated system: 17845 MW   4697 MVA
```

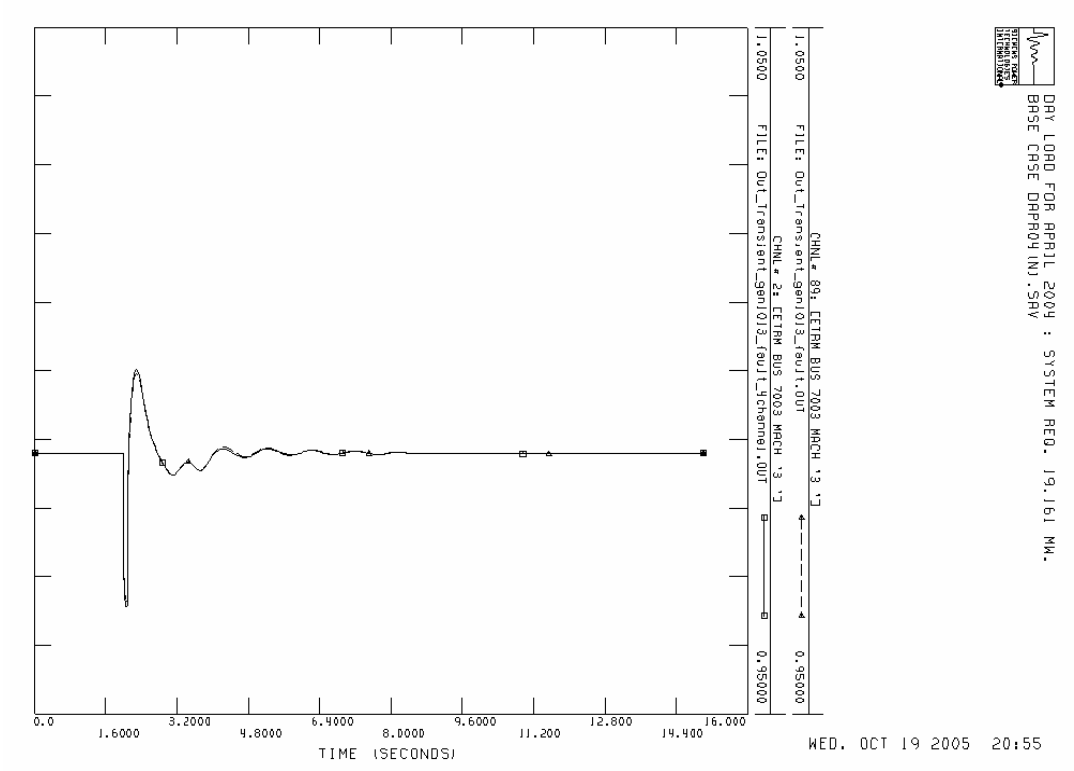
The simulation result is almost the same as the original system.

The compared graphs list below.

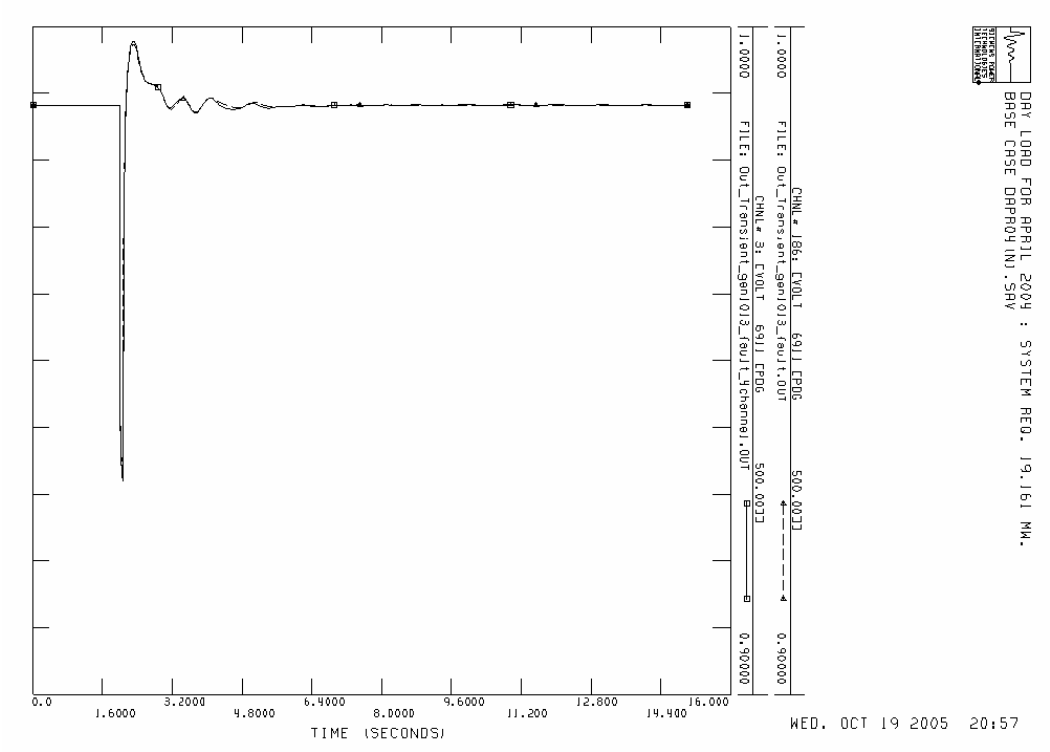
Power: Bus 7036 “2”



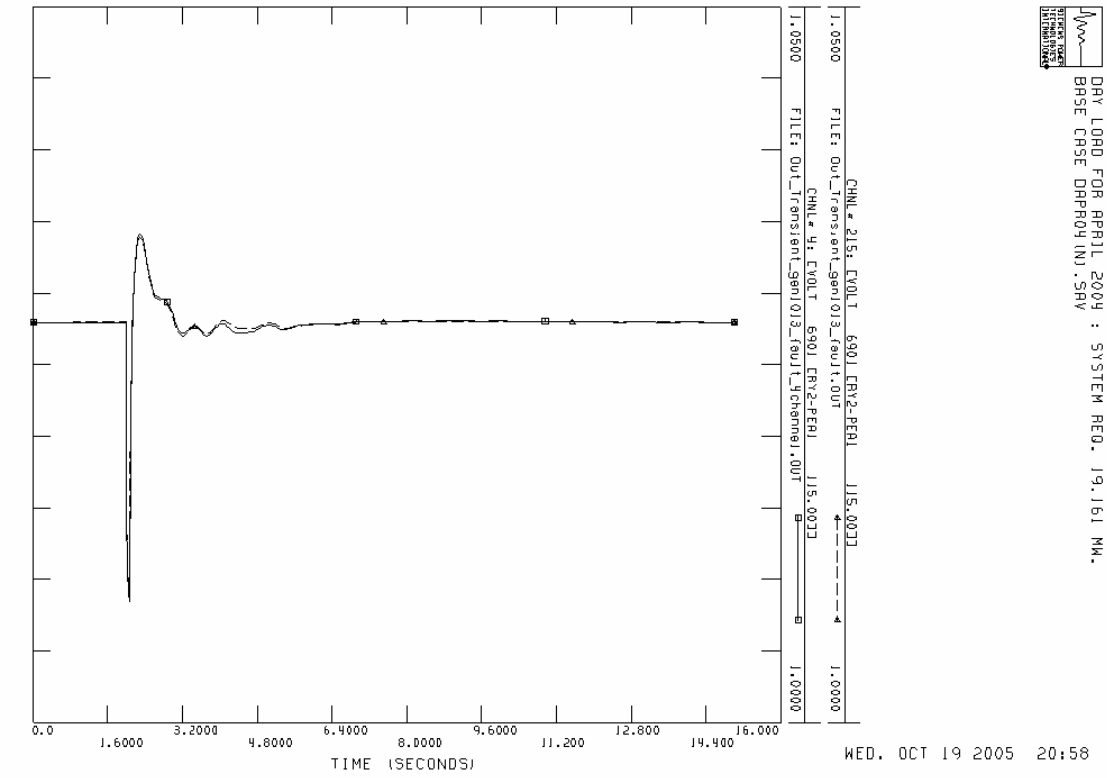
ETRM: Bus 7003 “3”




Voltage: Bus 6911



Voltage: Bus 6901



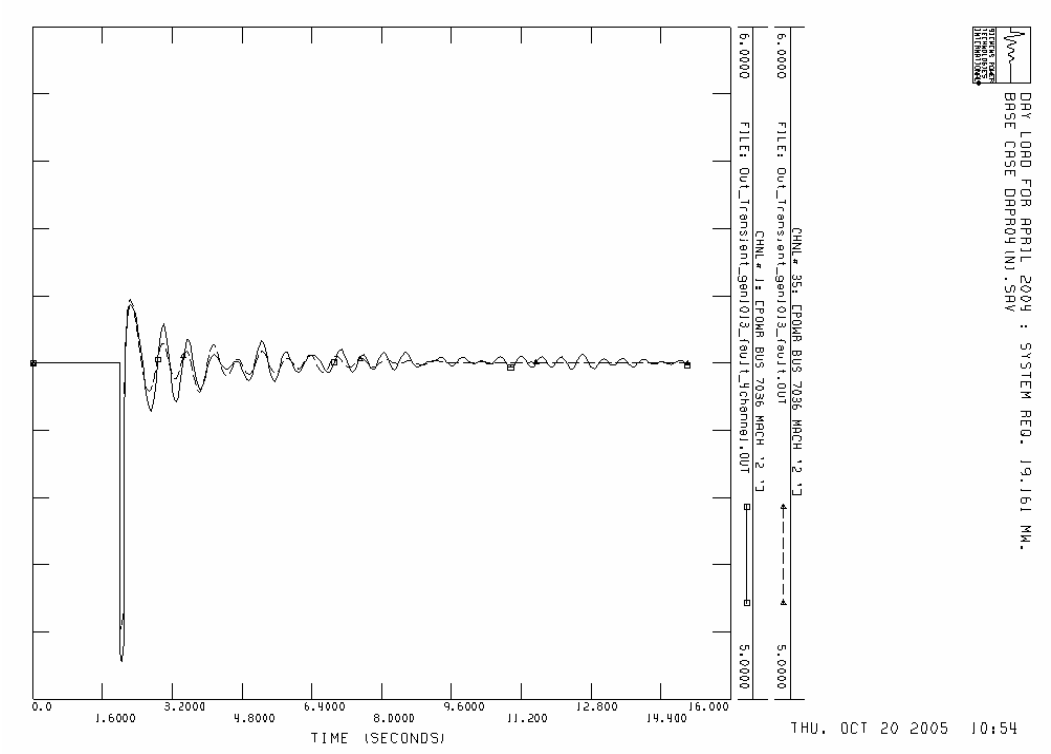

Case 3: For 7 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       15592 MW   6666 MVA
Total generation of the separated system: 15886 MW   4741 MVA
```

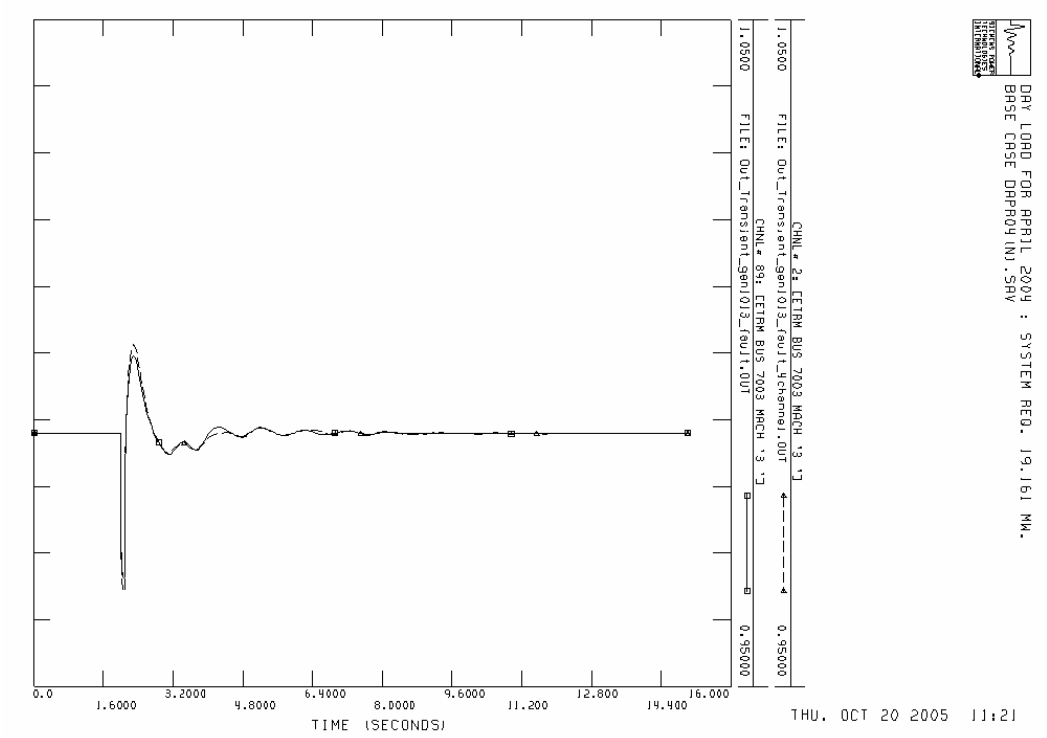
The simulation result is similar with the original system.

The compared graphs list below.

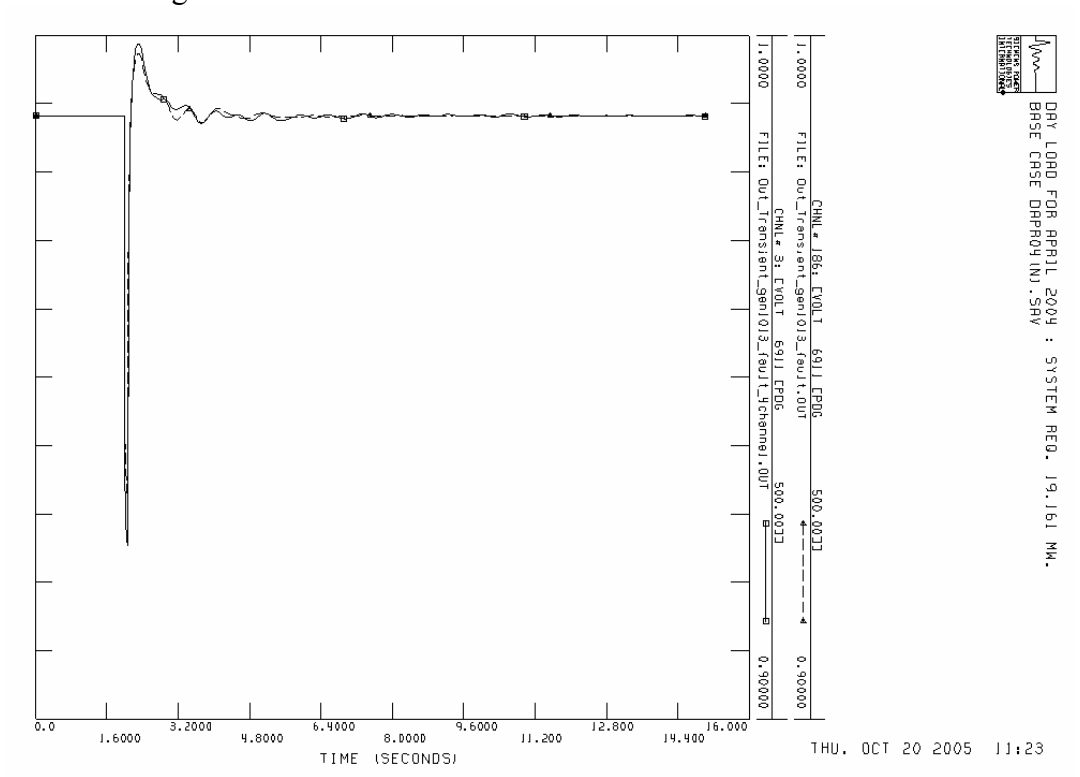
Power: Bus 7036 “2”



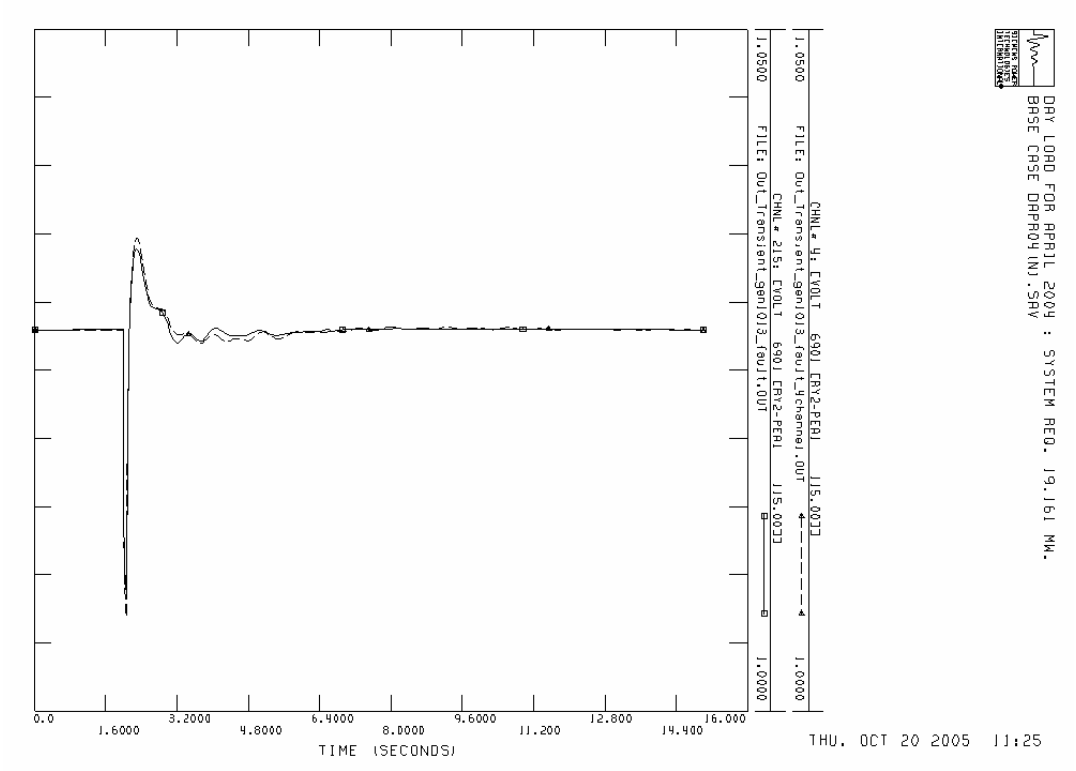

ETRM: Bus 7003 “3”



Voltage: Bus 6911




Voltage: Bus 6901



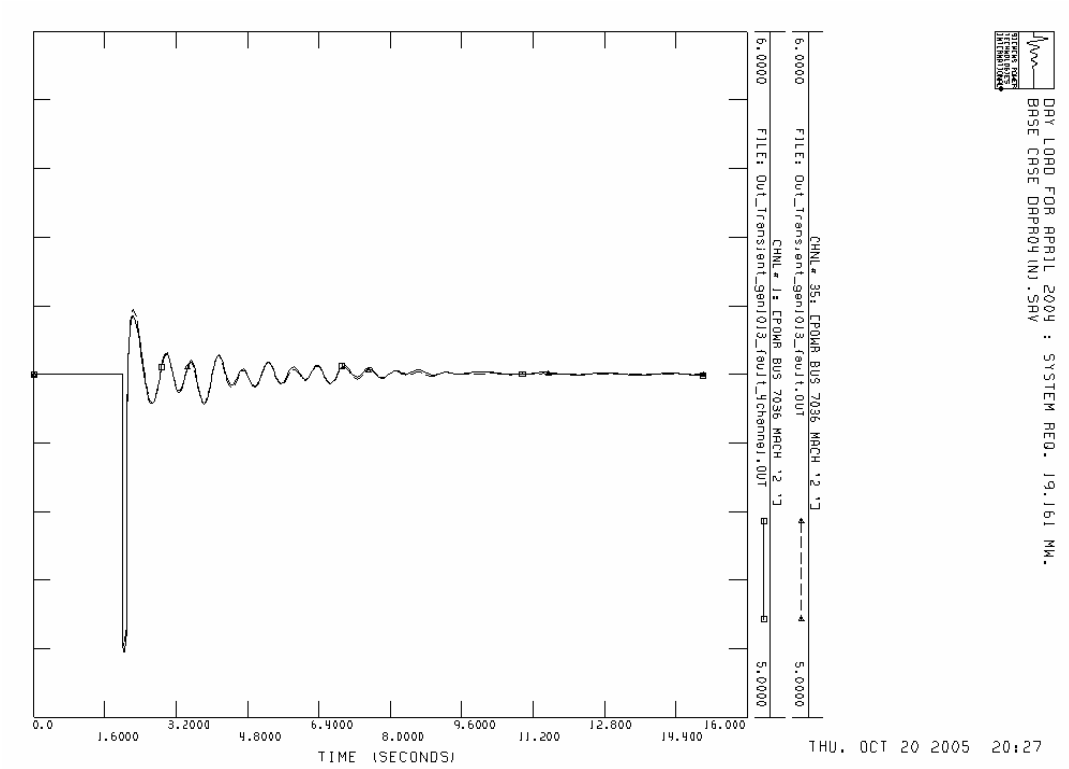
Case 4: For 6 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17288 MW   7217 MVA
Total generation of the separated system: 17636 MW   4631 MVA
```

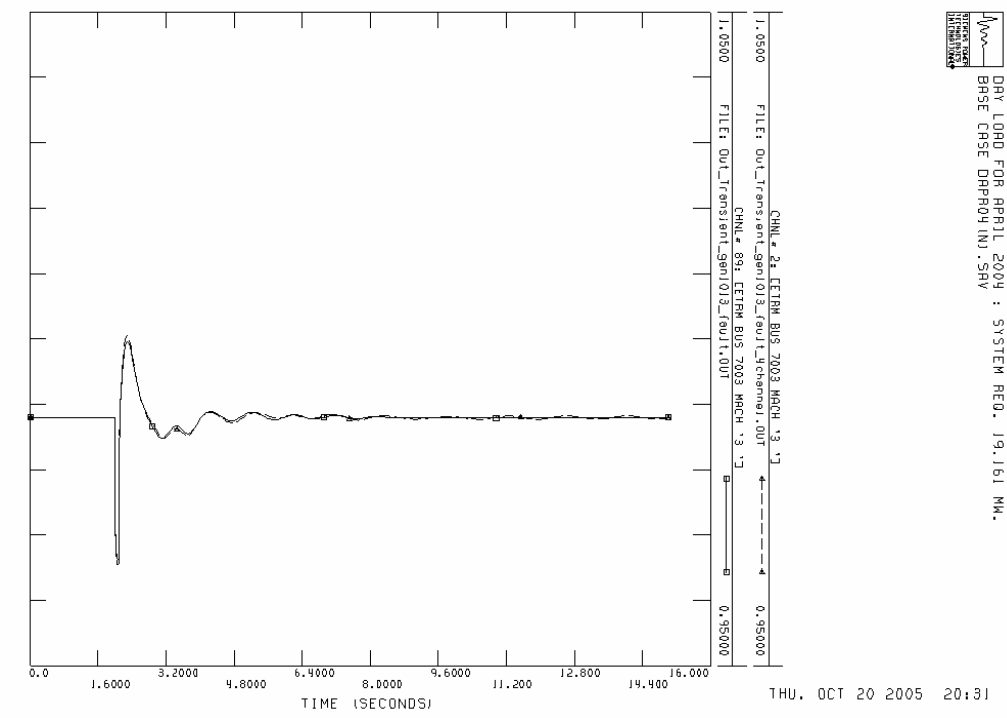
The simulation result is almost the same as the original system.

The compared graphs list below.

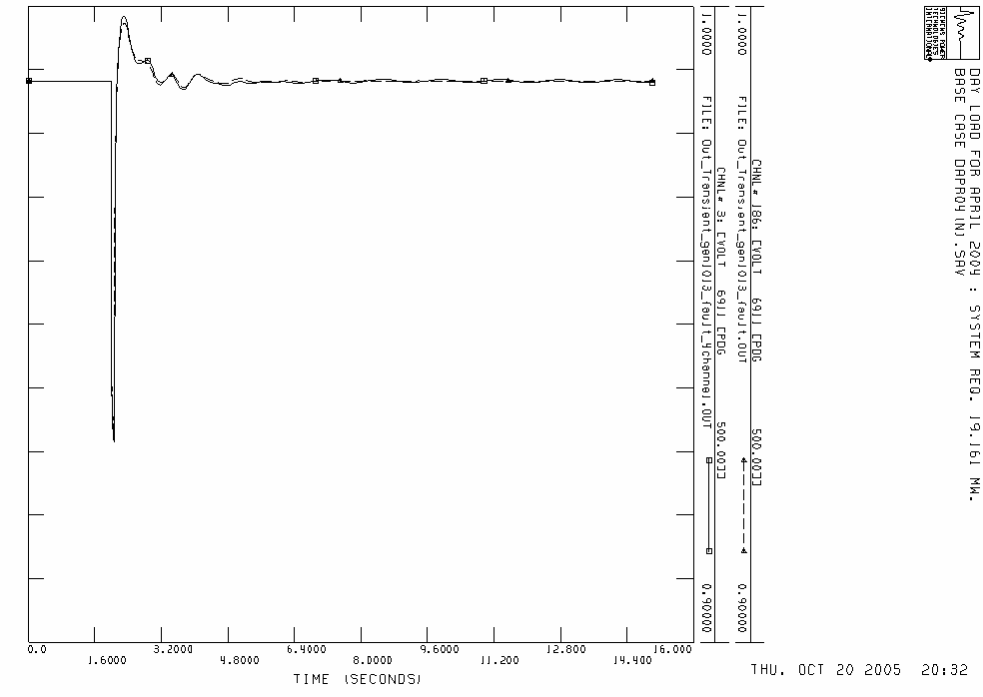
Power: Bus 7036 “2”



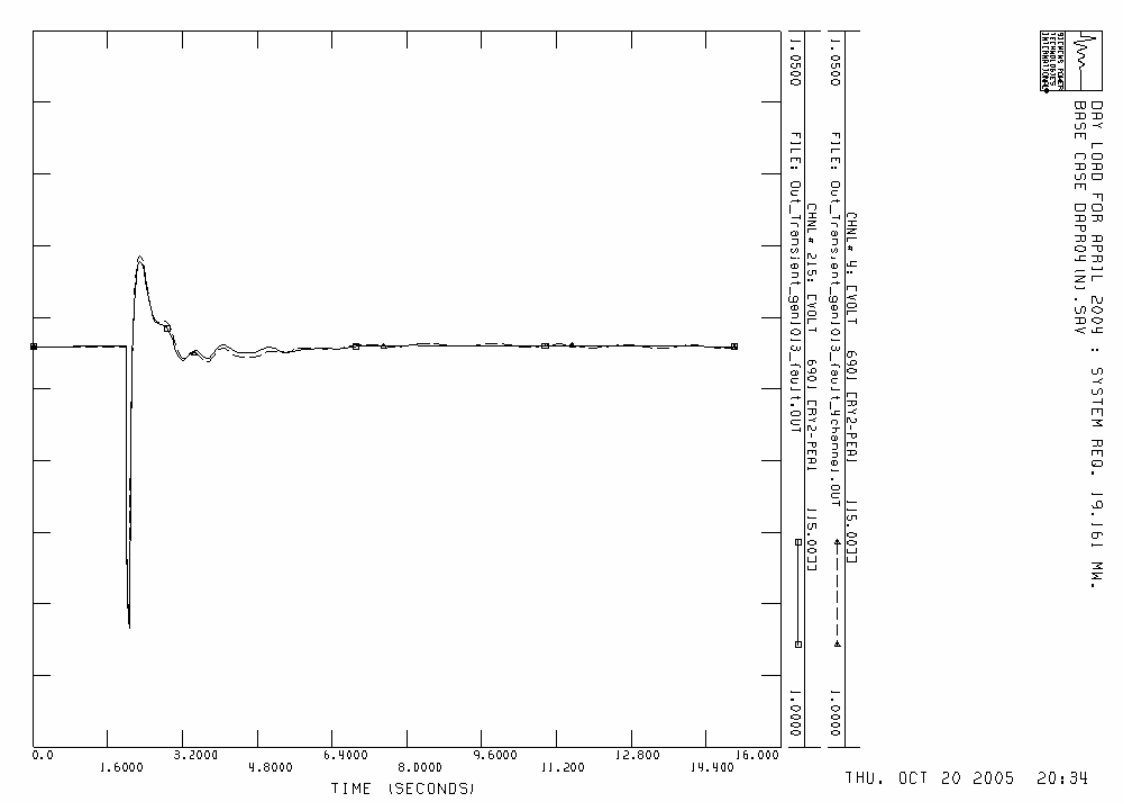
ETRM: Bus 7003 “3”




Voltage: Bus 6911



Voltage: Bus 6901




Case 5: For 5 layers of buses near the fault:

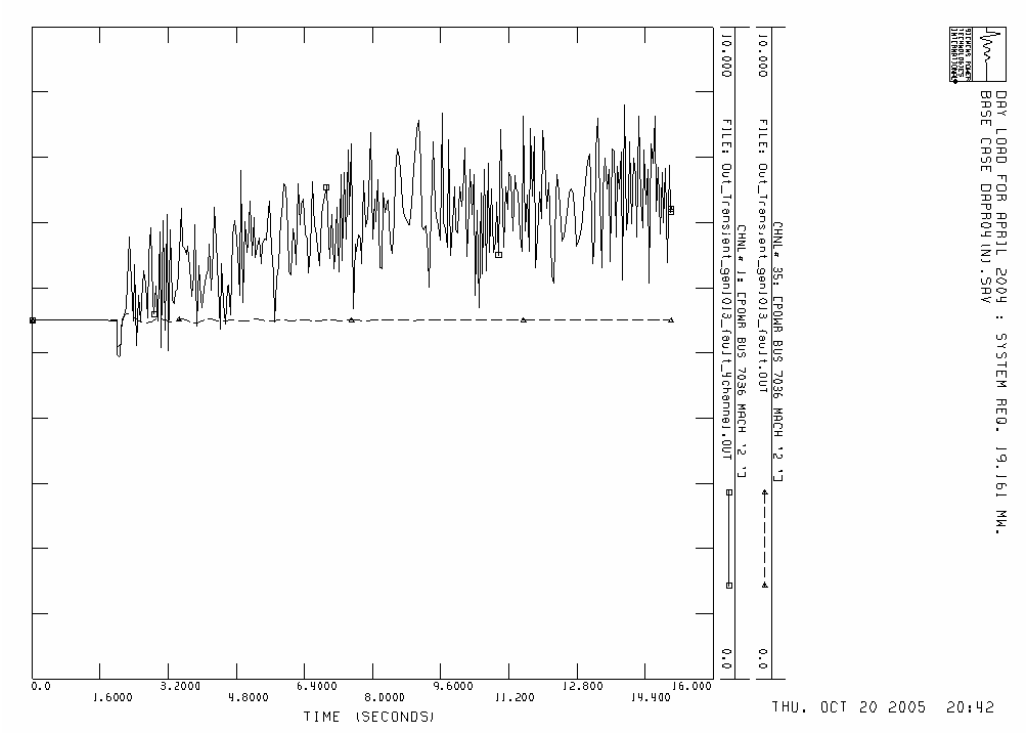
```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       13913 MW   5885 MVA
Total generation of the separated system: 14154 MW   3962 MVA
```

The simulation result is totally different from the original system.

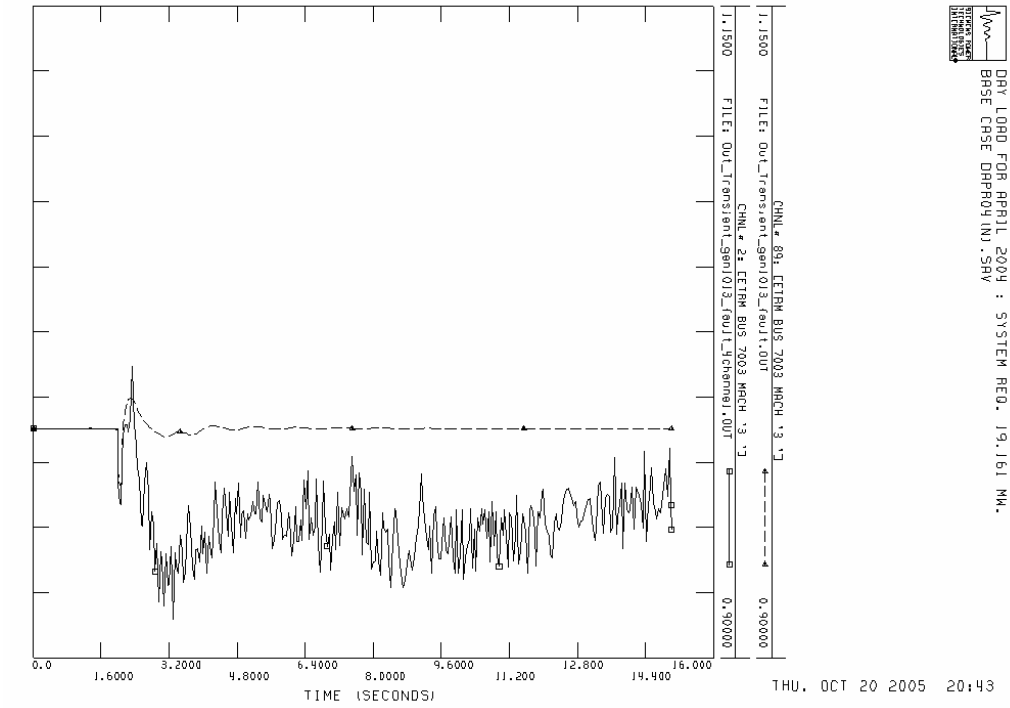
The compared graphs list below.

For 5 layers of buses near the fault:

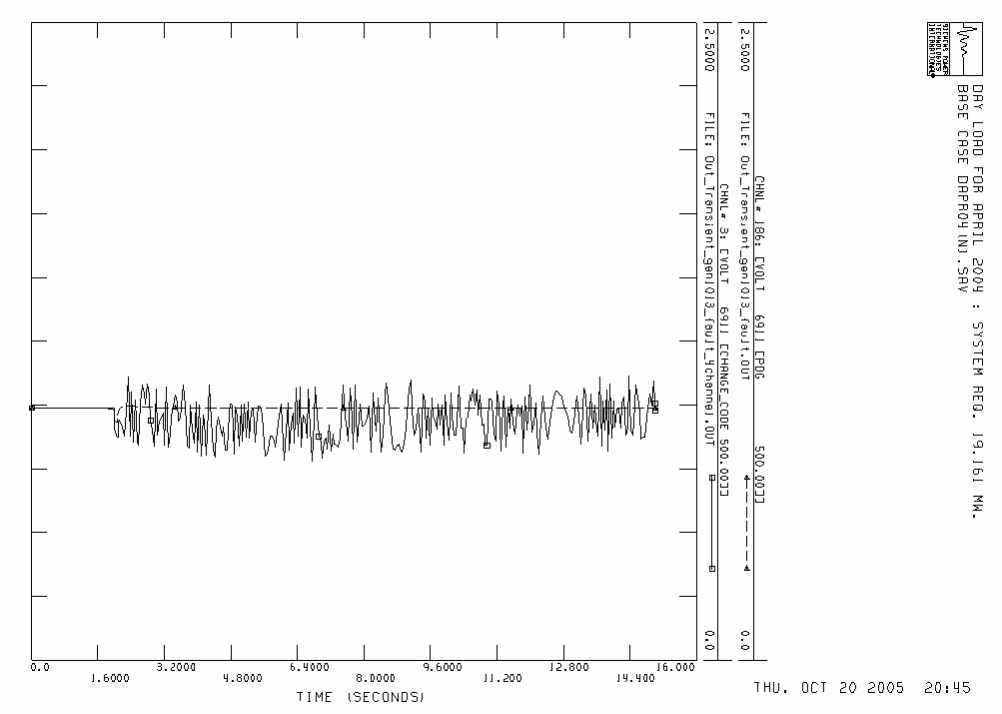
Power: Bus 7036 “2”



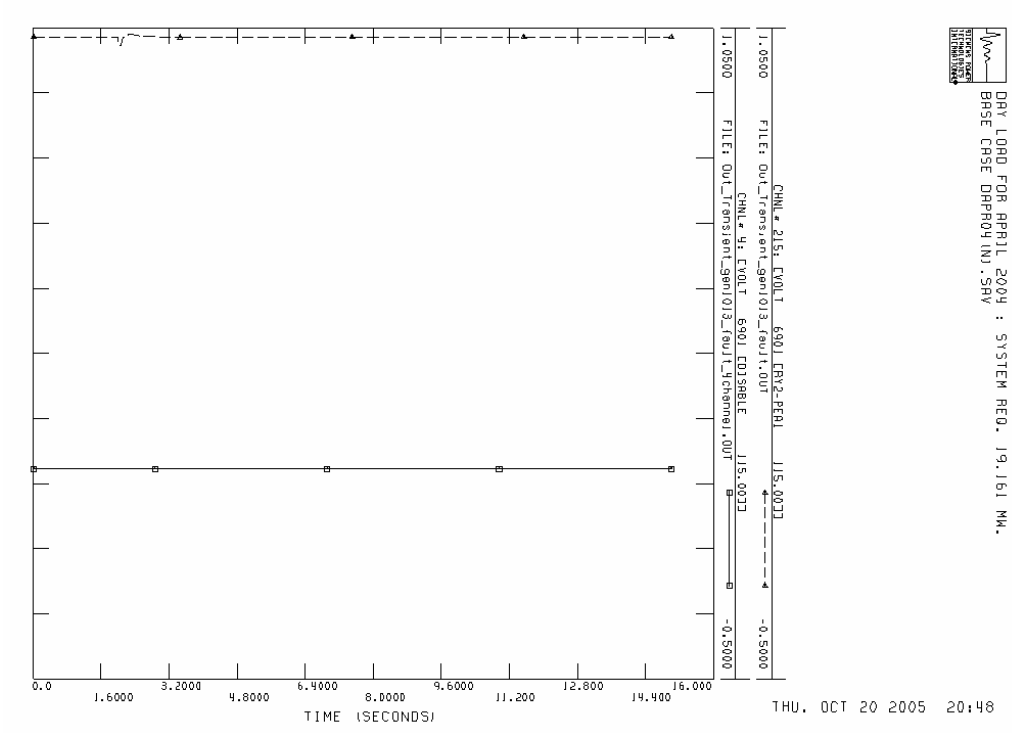

ETRM: Bus 7003 “3”



Voltage: Bus 6911




Voltage: Bus 6901



4. Step 6

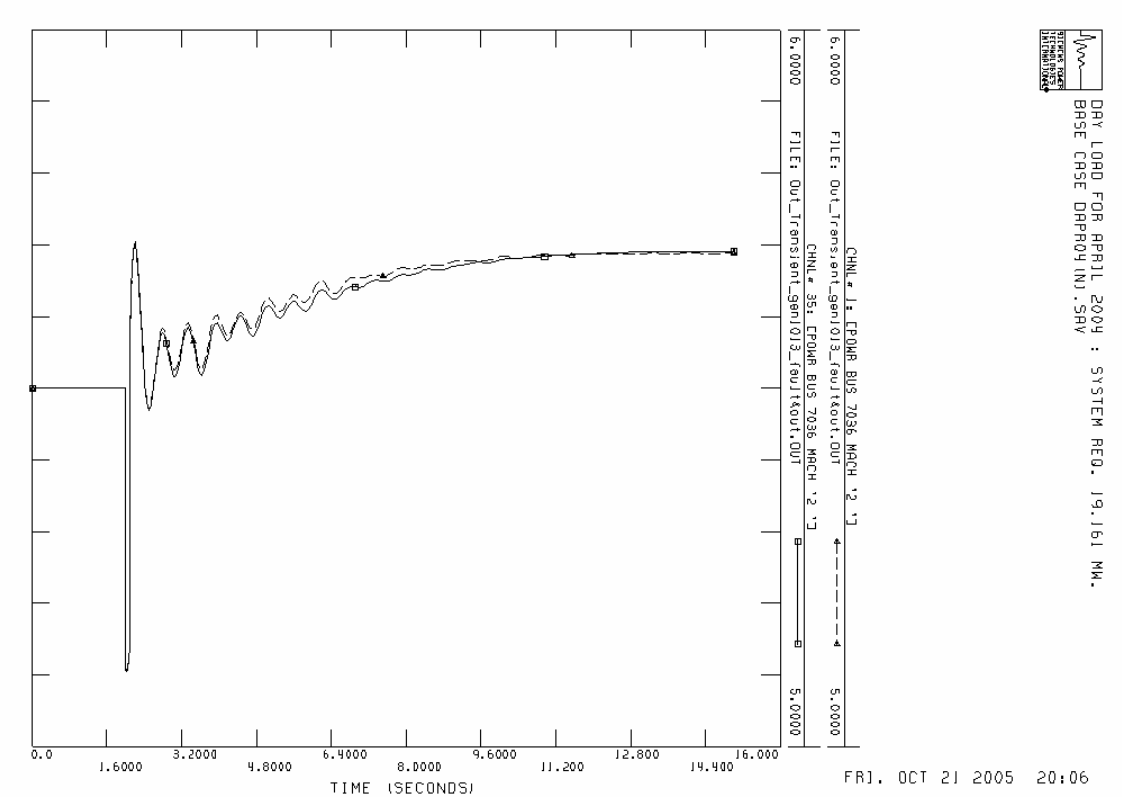
Case 1: For 10 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17568 MW   7602 MVA
Total generation of the separated system: 18008 MW   4680 MVA
```

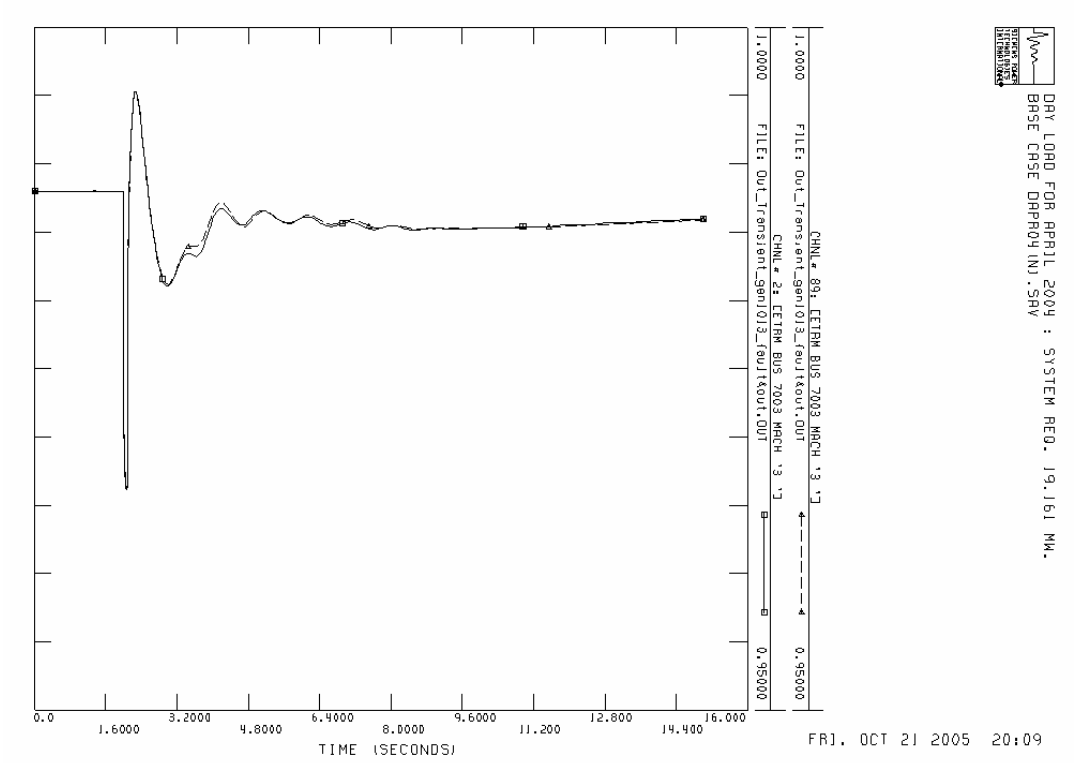
The simulation result is almost the same as the original system.

The compared graphs list below.

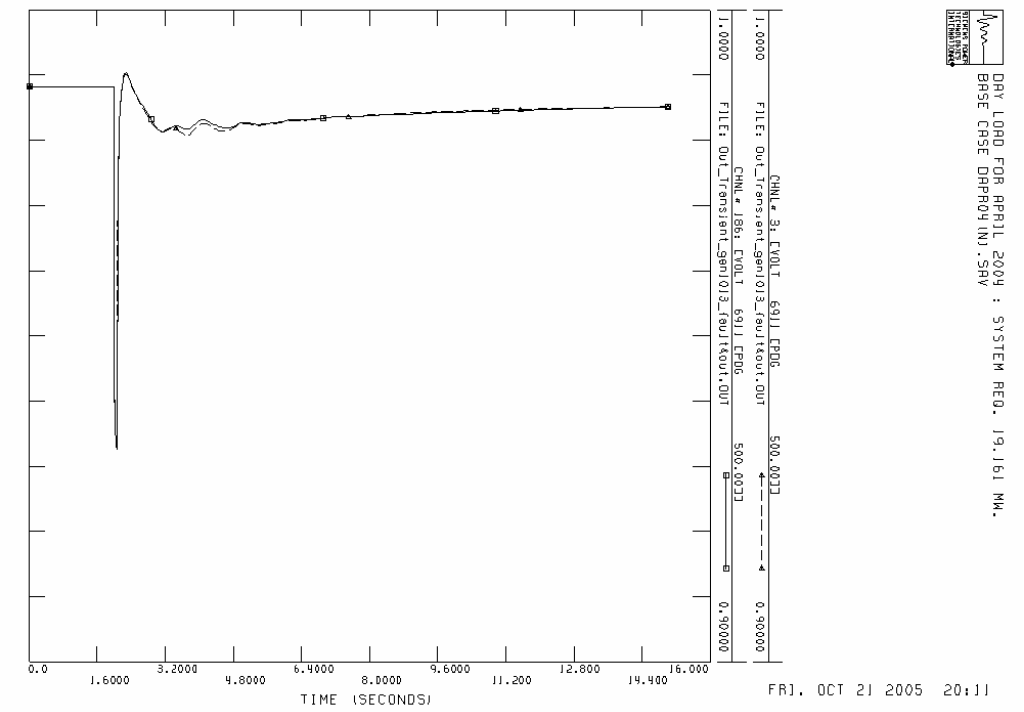
Power: Bus 7036 “2”



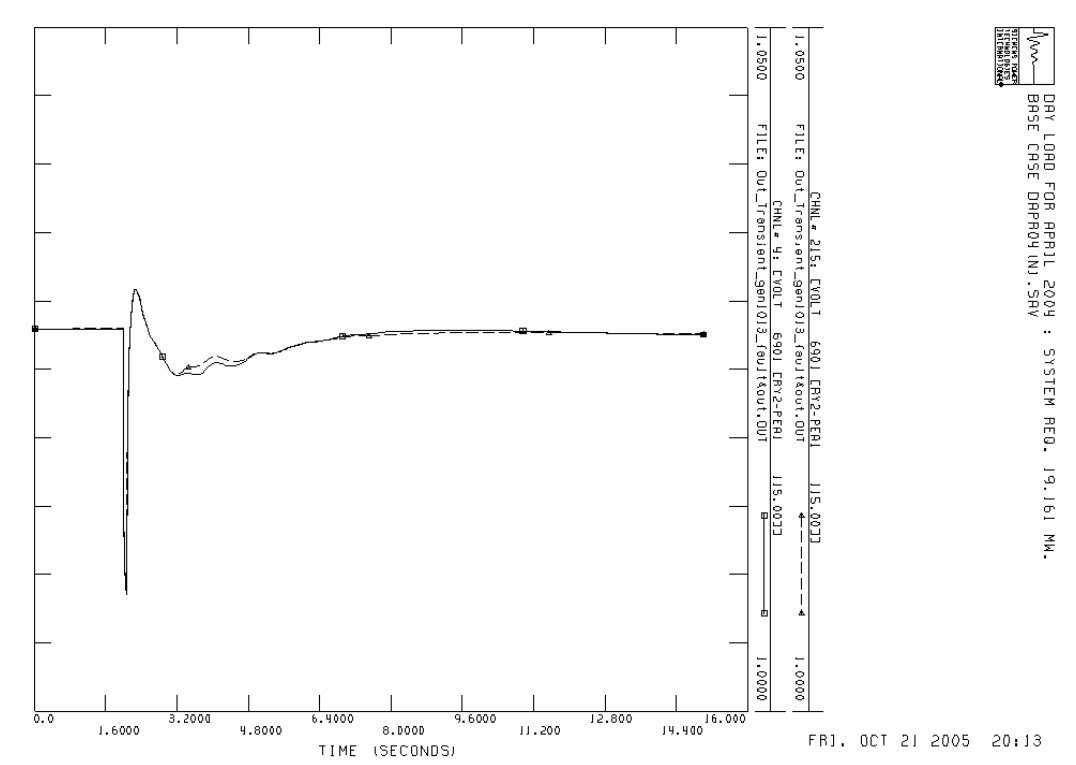
ETRM: Bus 7003 “3”




Voltage: Bus 6911



Voltage: Bus 6901



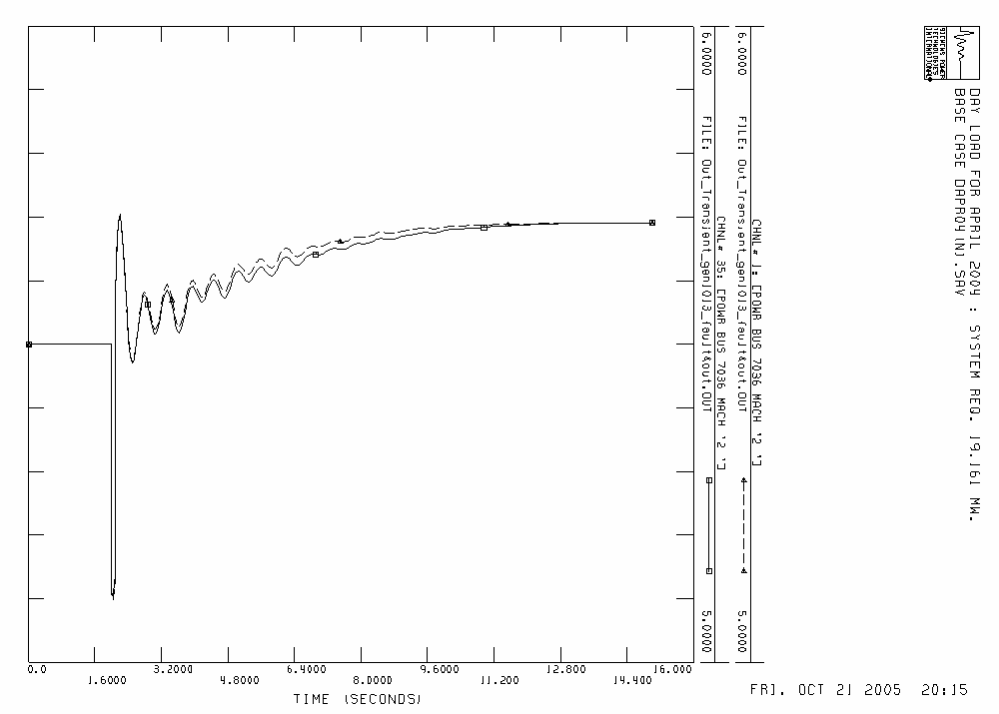

Case 2: For 8 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17432 MW   7471 MVA
Total generation of the separated system: 17845 MW   4697 MVA
```

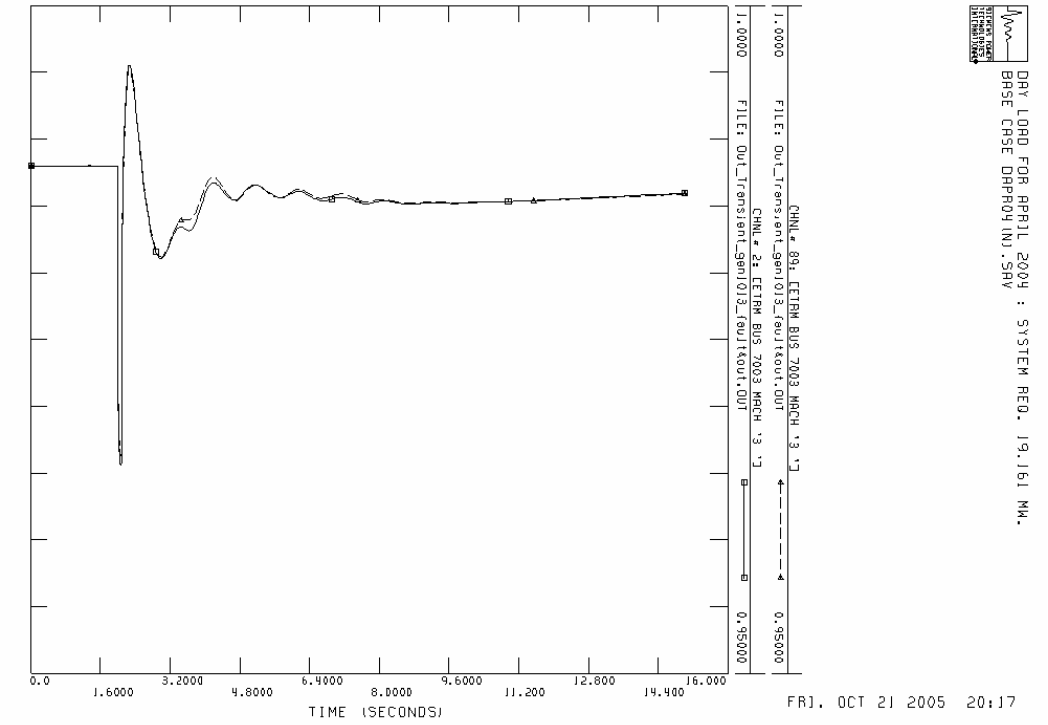
The simulation result is almost the same as the original system.

The compared graphs list below.

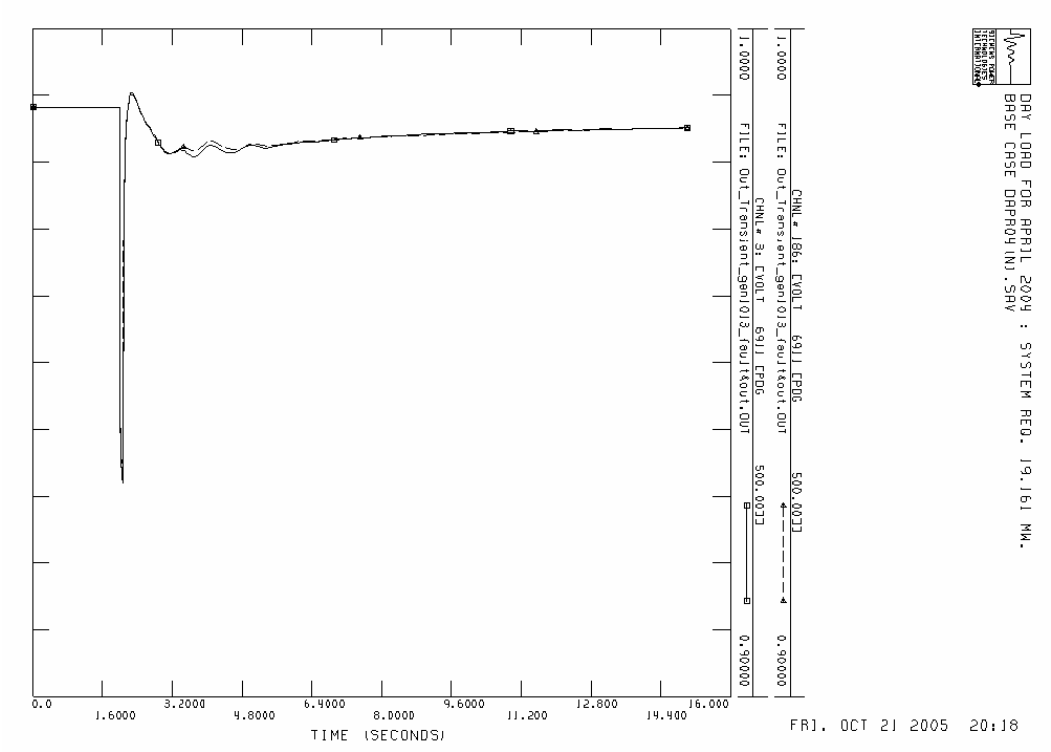
Power: Bus 7036 “2”



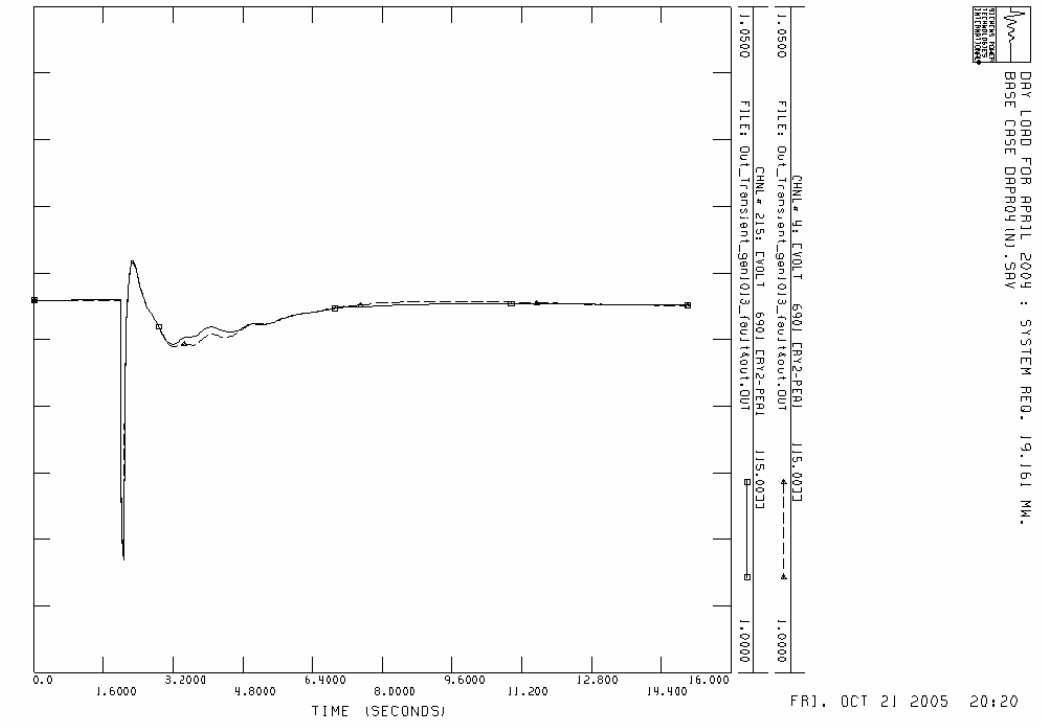

ETRM: Bus 7003 “3”



Voltage: Bus 6911




Voltage: Bus 6901



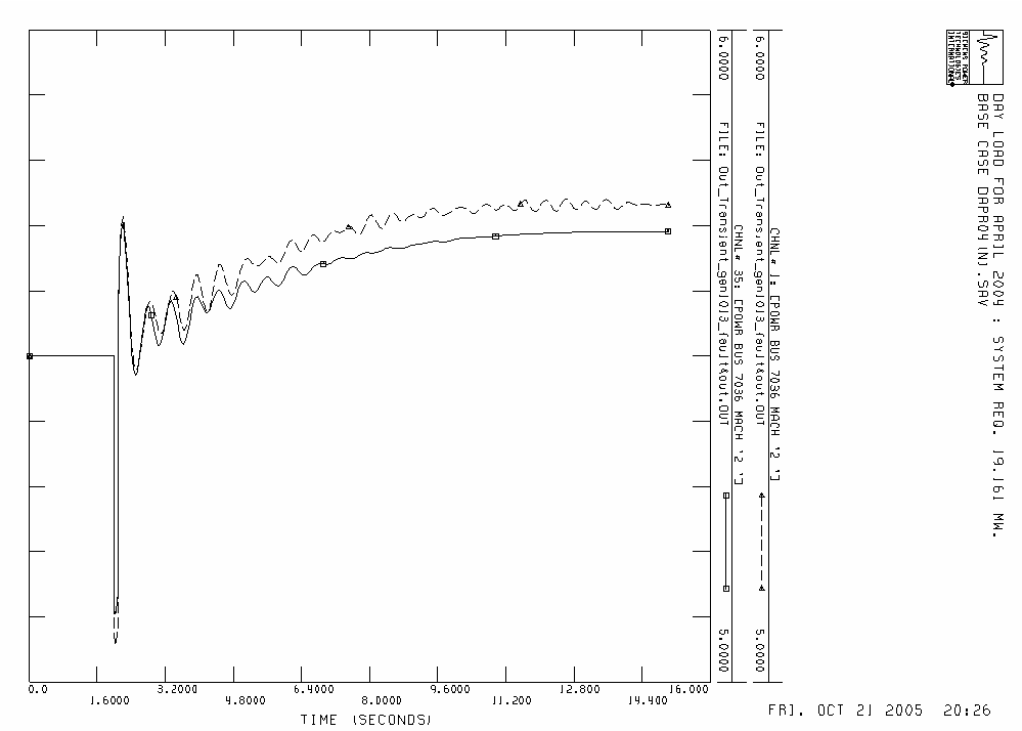
Case 3: For 7 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       15592 MW   6666 MVA
Total generation of the separated system: 15886 MW   4741 MVA
```

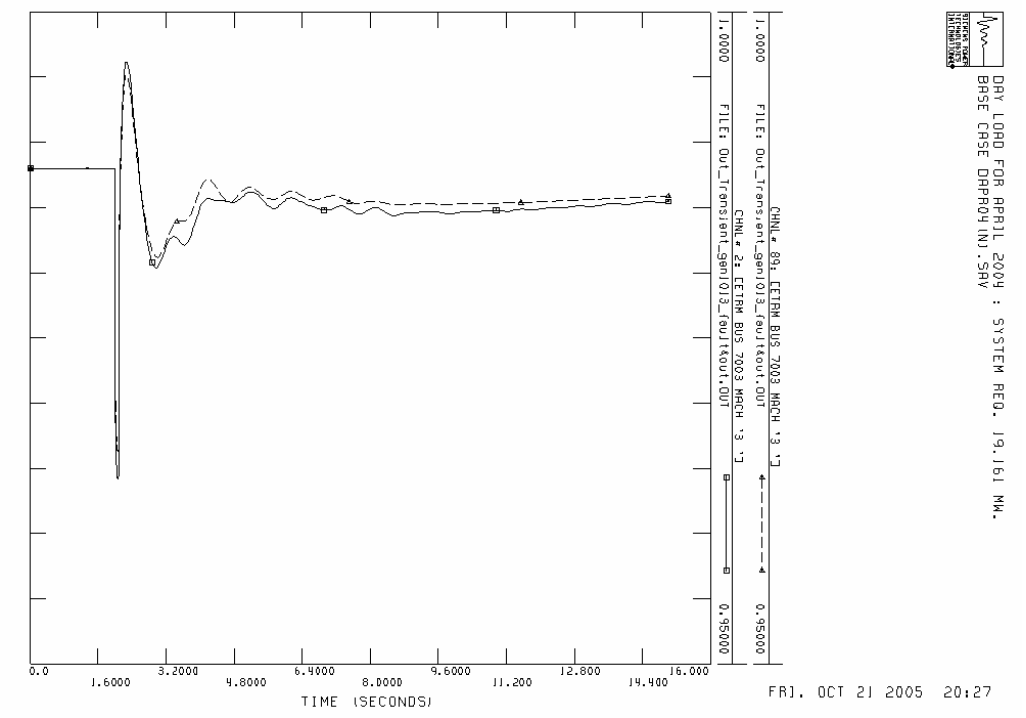
The simulation result is similar with the original system.

The compared graphs list below.

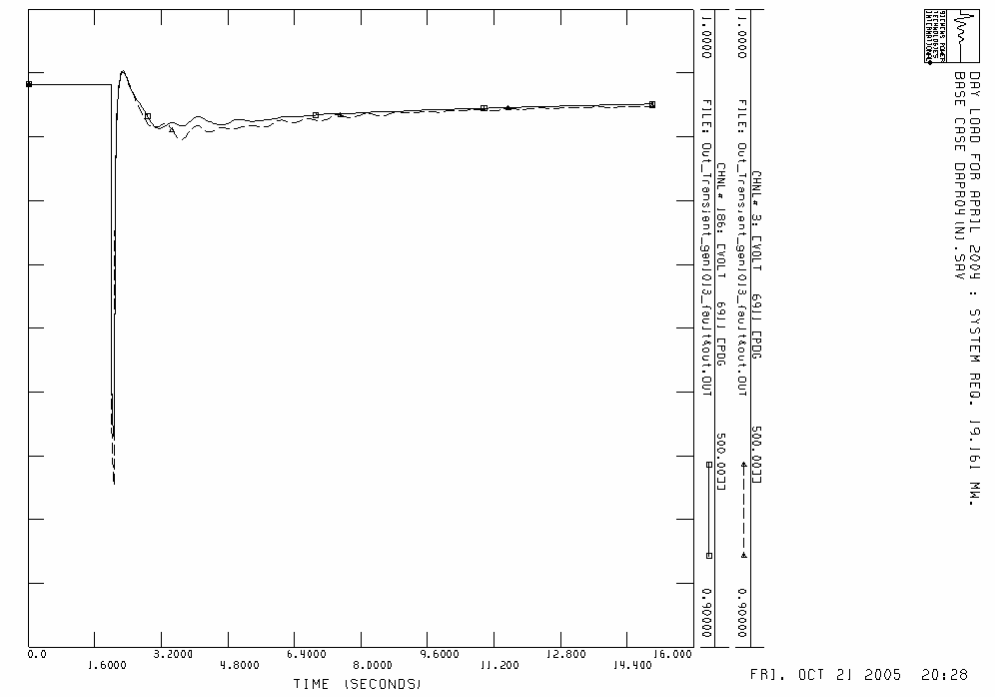
Power: Bus 7036 “2”



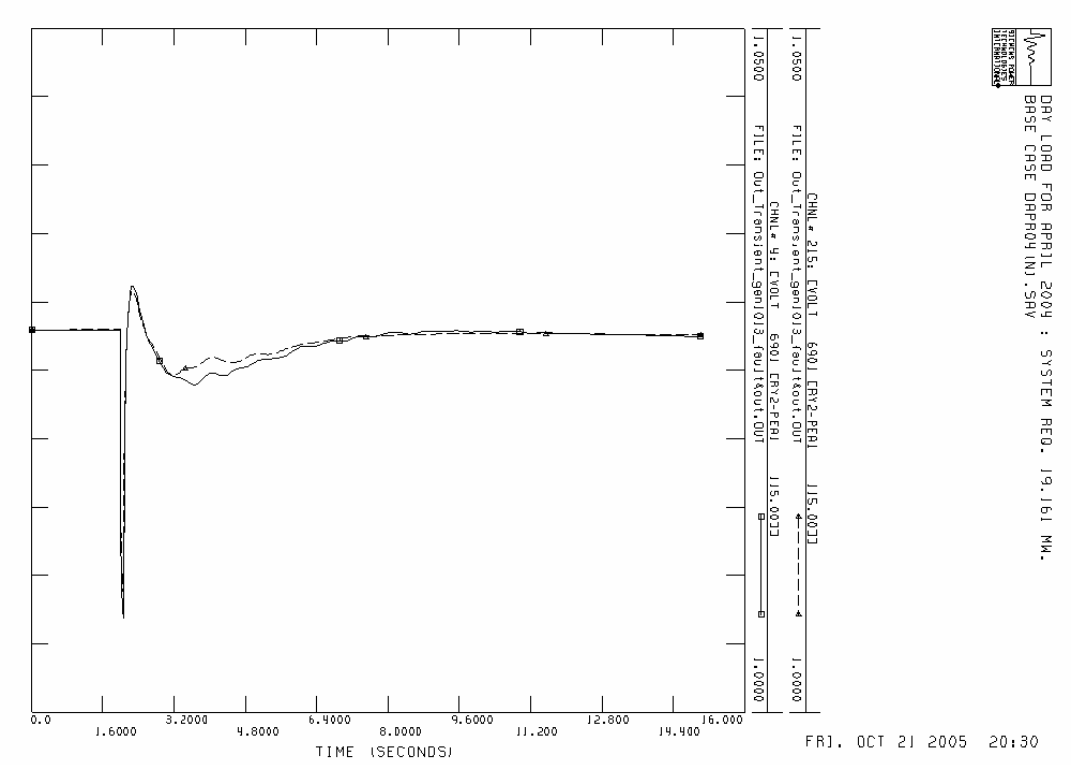
ETRM: Bus 7003 “3”




Voltage: Bus 6911



Voltage: Bus 6901



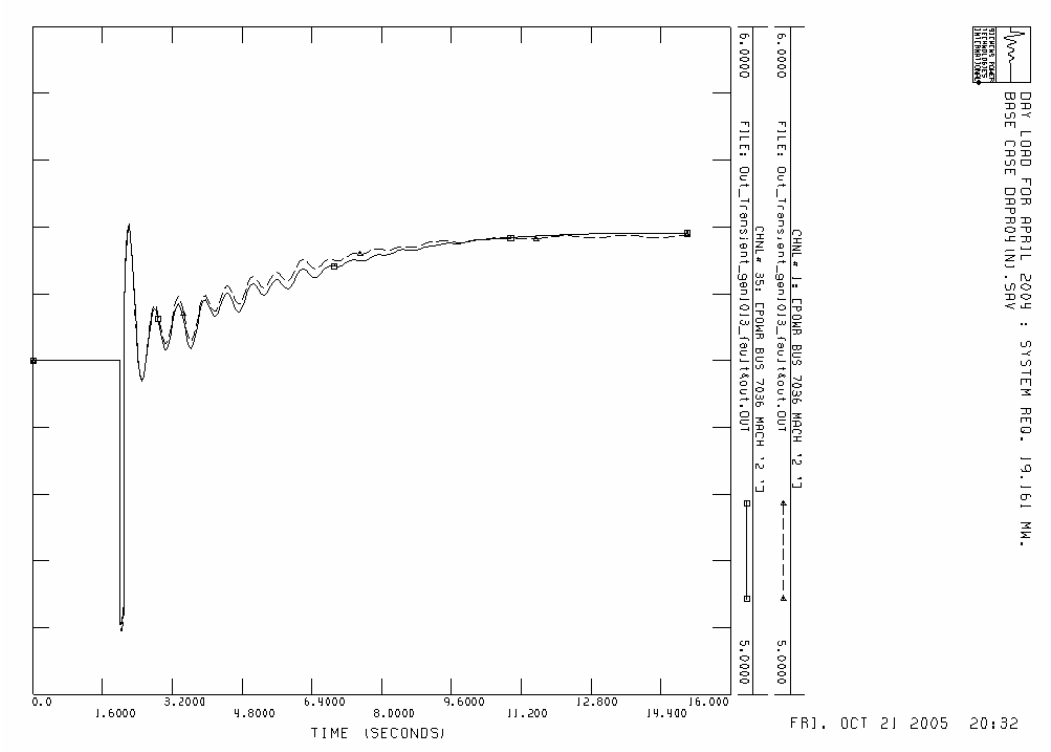

Case 4: For 6 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17288 MW   7217 MVA
Total generation of the separated system: 17636 MW   4631 MVA
```

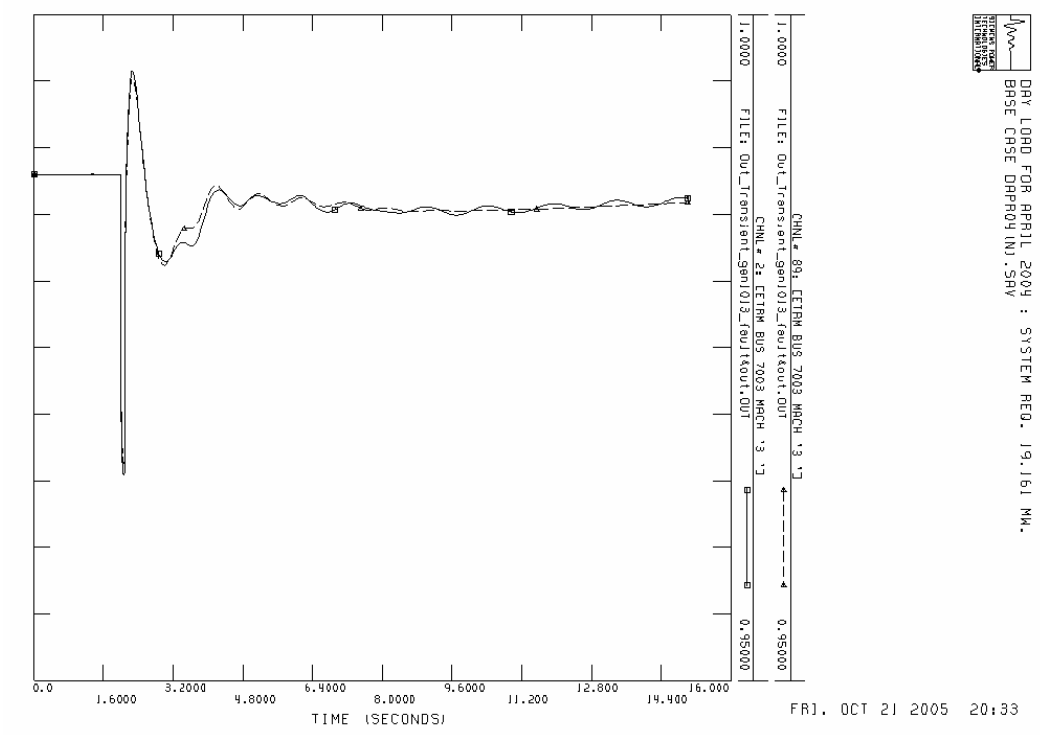
The simulation result is almost the same as the original system.

The compared graphs list below.

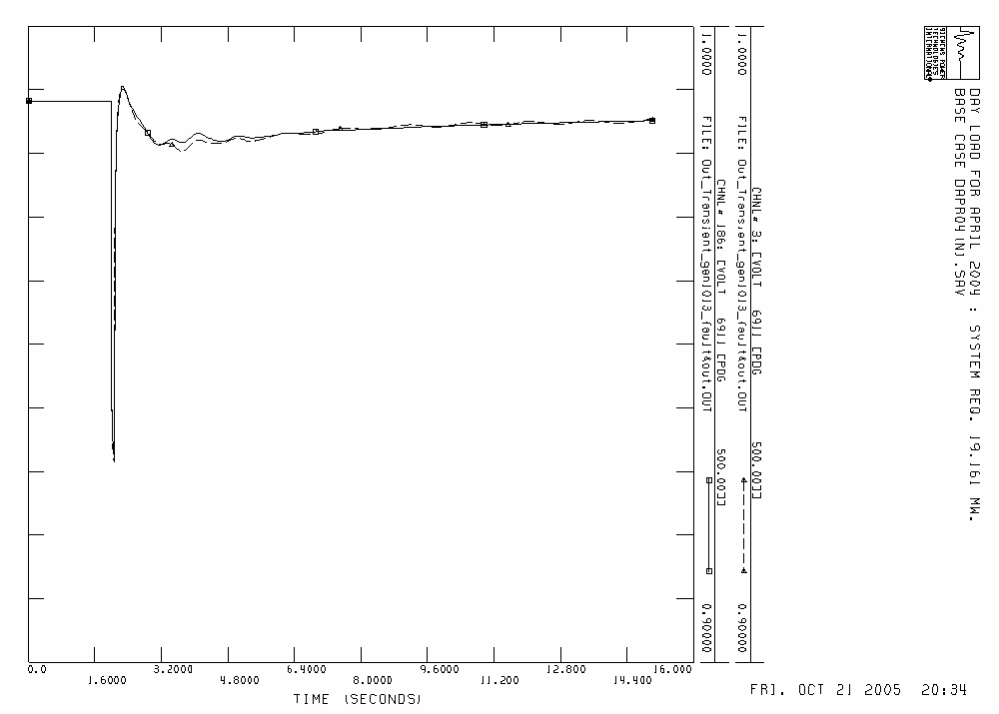
Power: Bus 7036 “2”



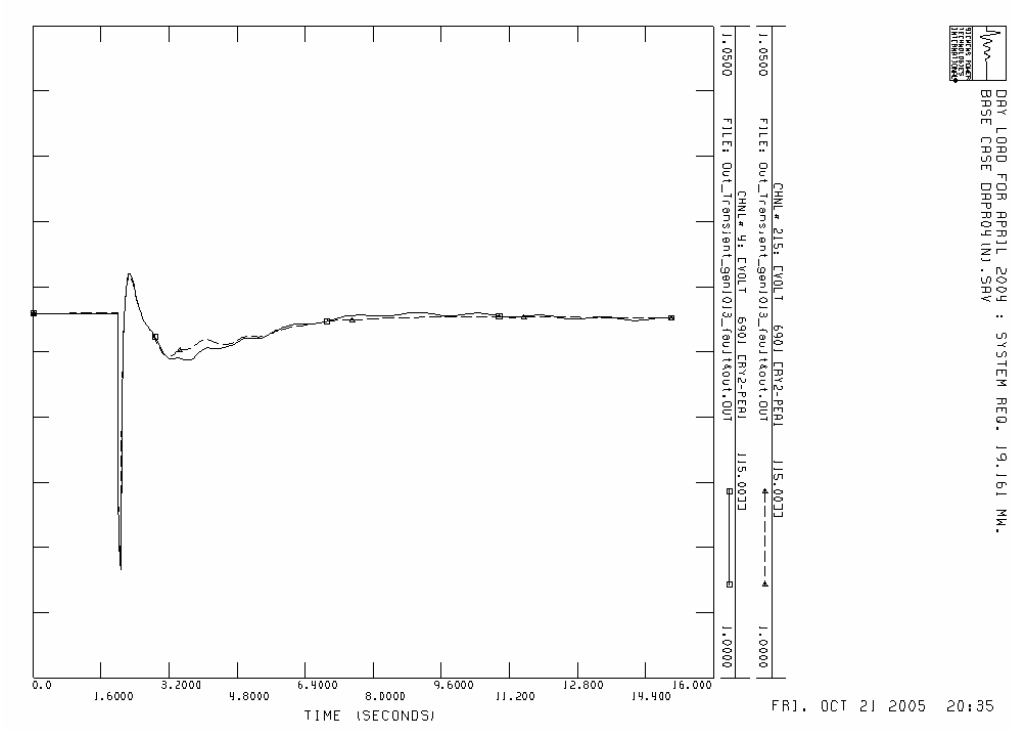

ETRM: Bus 7003 “3”



Voltage: Bus 6911




Voltage: Bus 6901



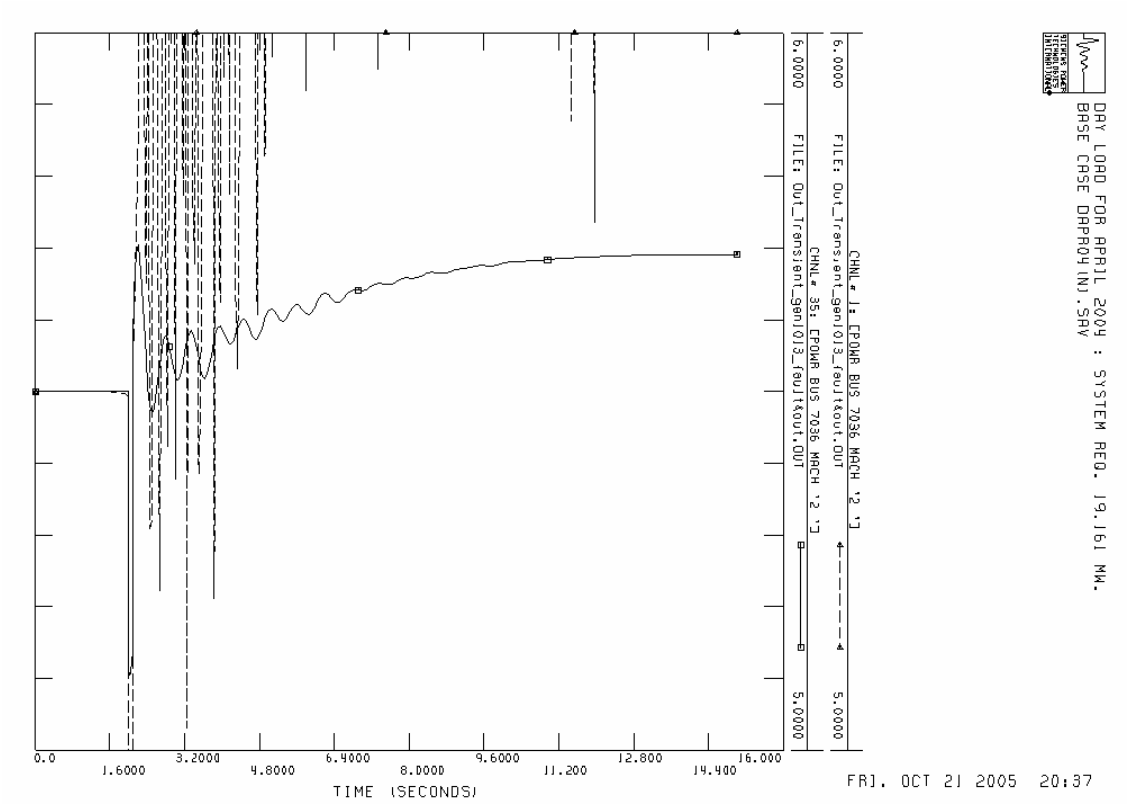
Case 5: For 5 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       13913 MW   5885 MVA
Total generation of the separated system: 14154 MW   3962 MVA
```

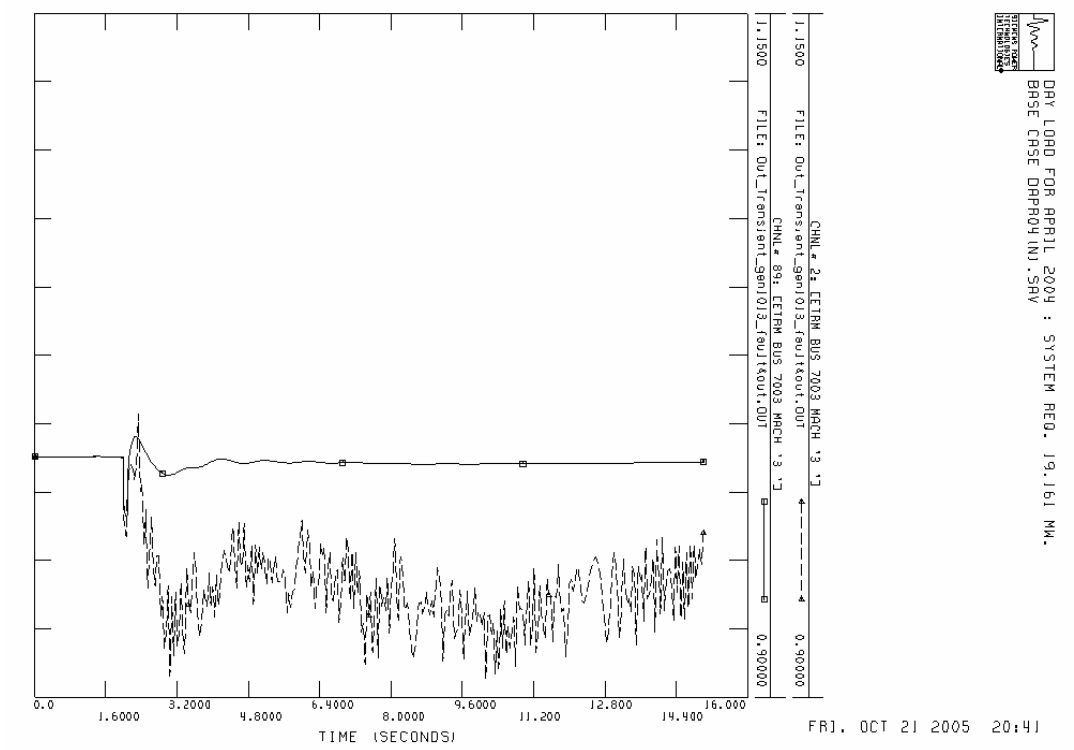
The simulation result is totally different from the original system.

The compared graphs list below.

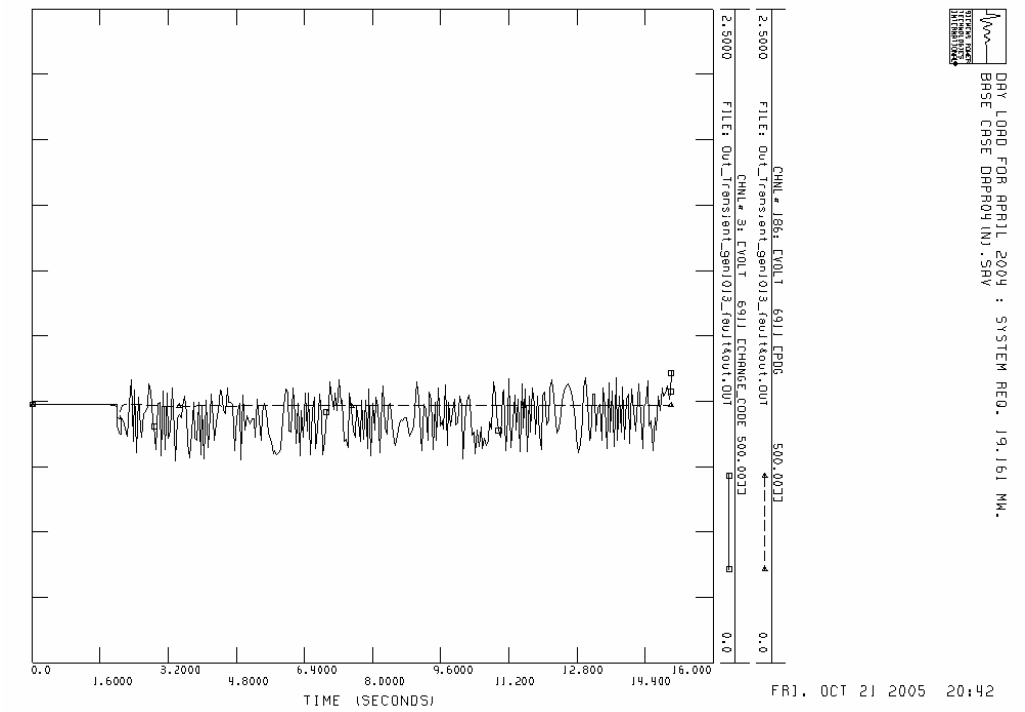
Power: Bus 7036 “2”



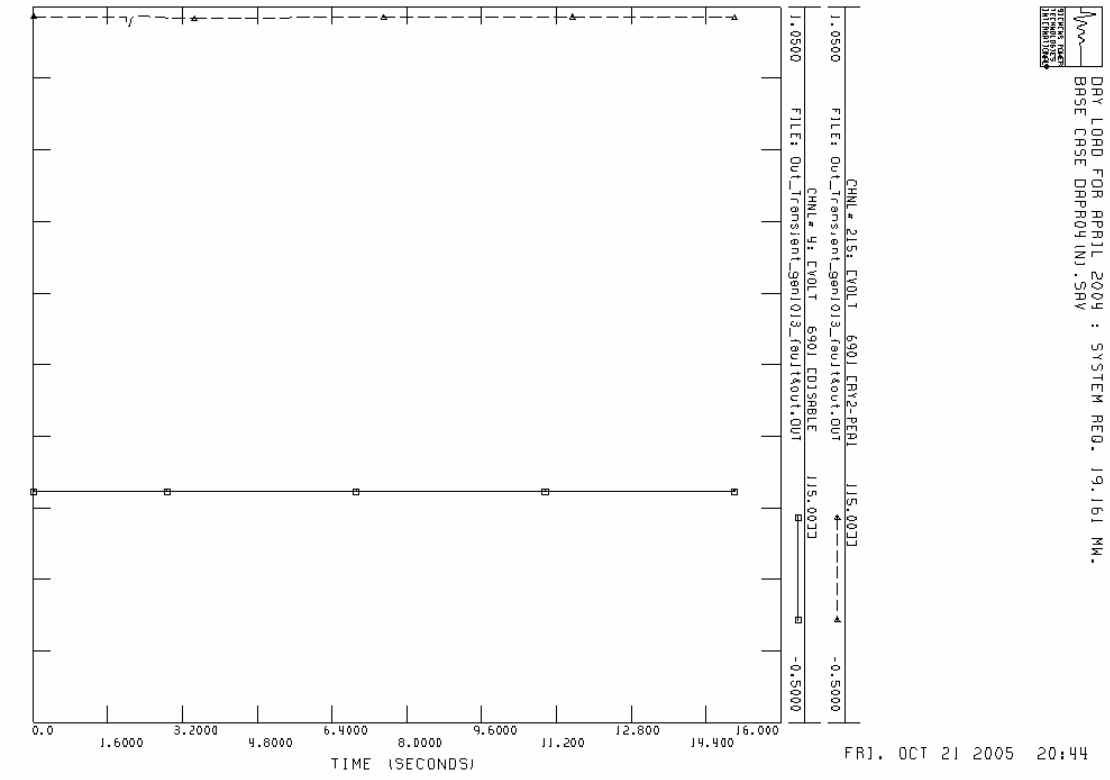
ETRM: Bus 7003 “3”




Voltage: Bus 6911



Voltage: Bus 6901




Step 7

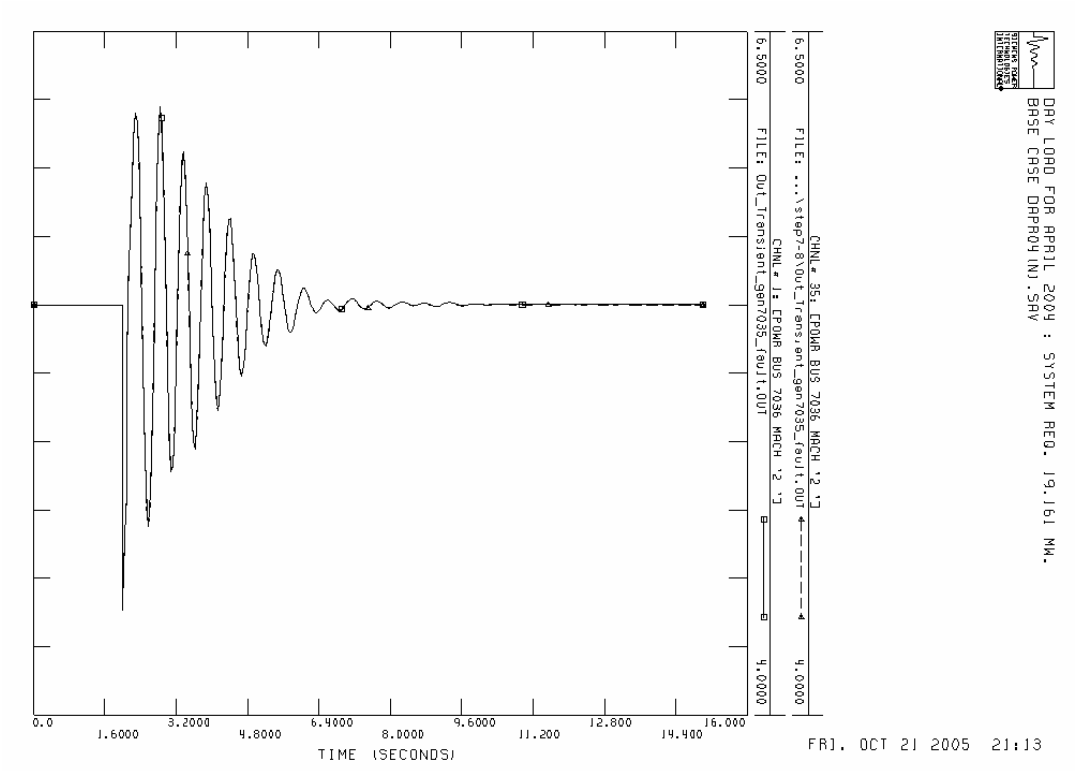
Case 1: For 10 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       18393 MW   7945 MVA
Total generation of the separated system: 18868 MW   4678 MVA
```

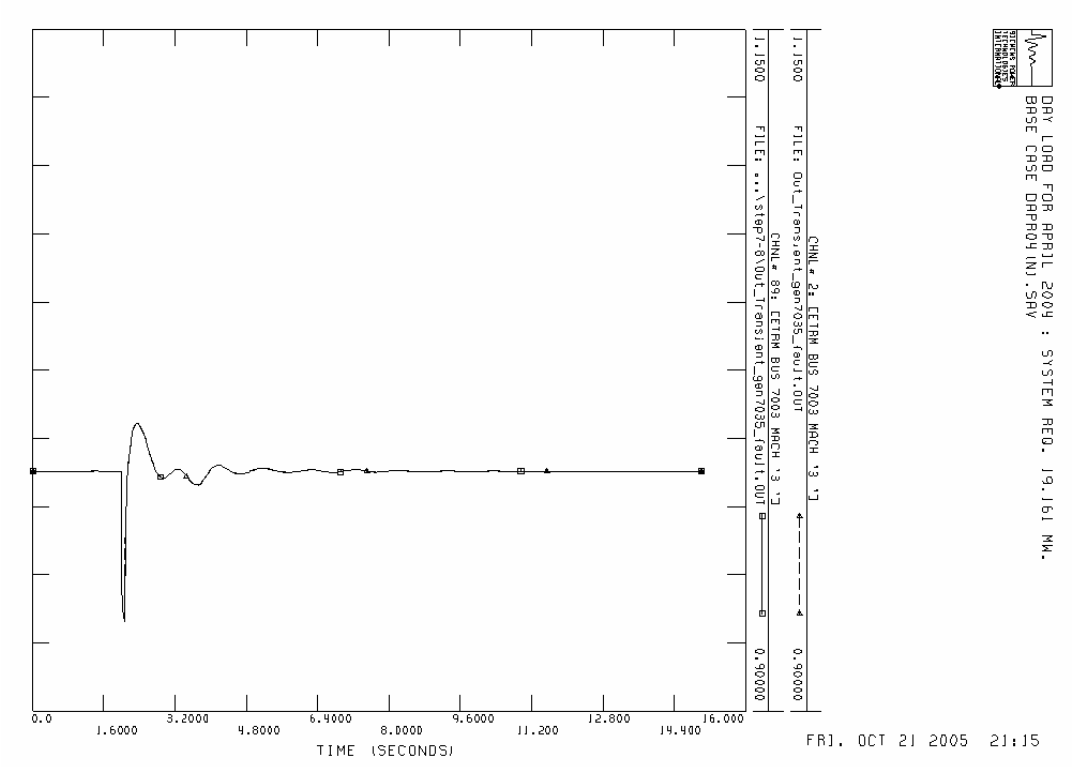
The simulation result is almost the same as the original system.

The compared graphs list below.

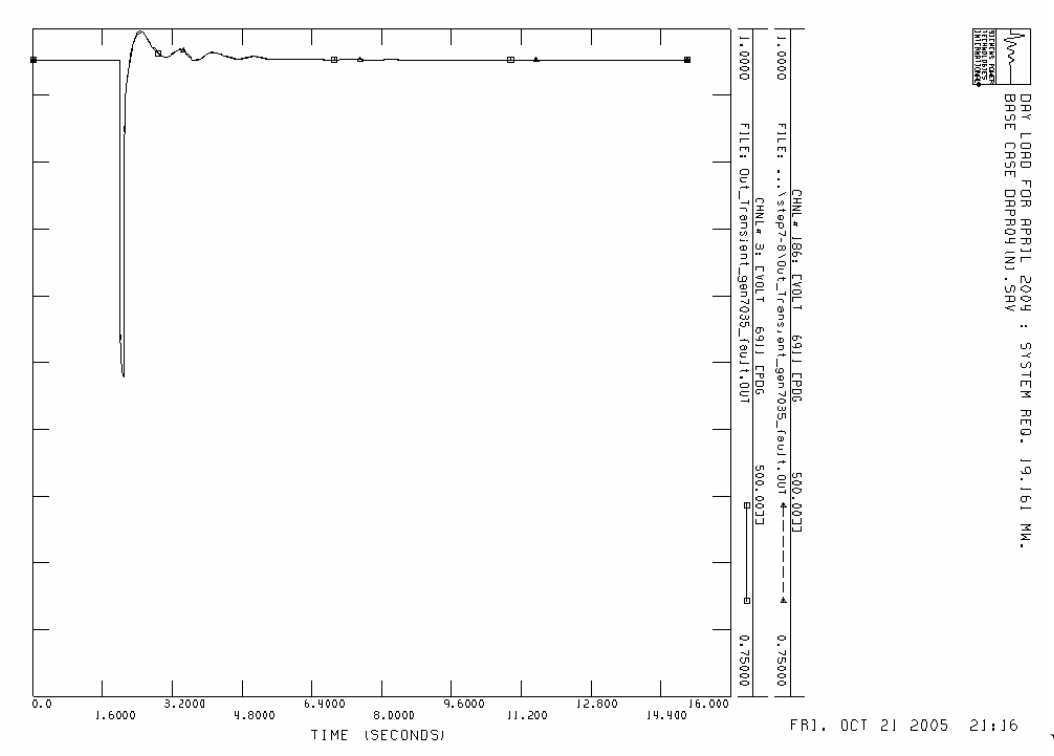
Power: Bus 7036 “2”




ETRM: Bus 7003 “3”

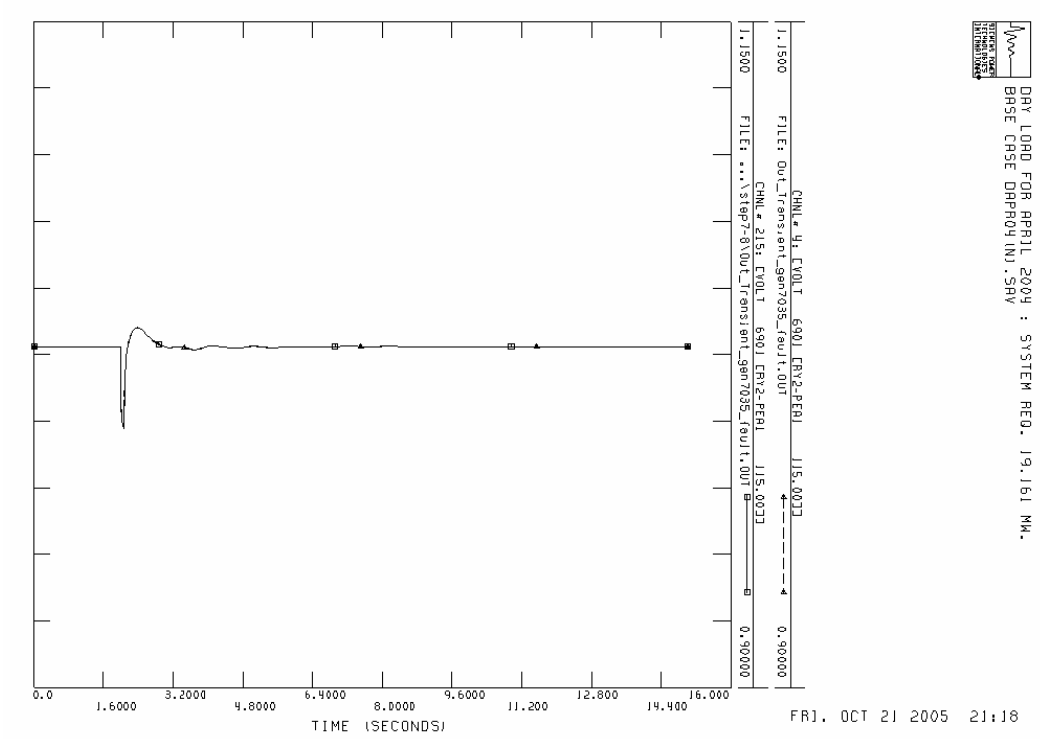


Voltage: Bus 6911




V

Voltage: Bus 6901



Case 2: For 8 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17691 MW   7624 MVA
Total generation of the separated system: 18129 MW   4716 MVA
```

The simulation result is almost the same as the original system.

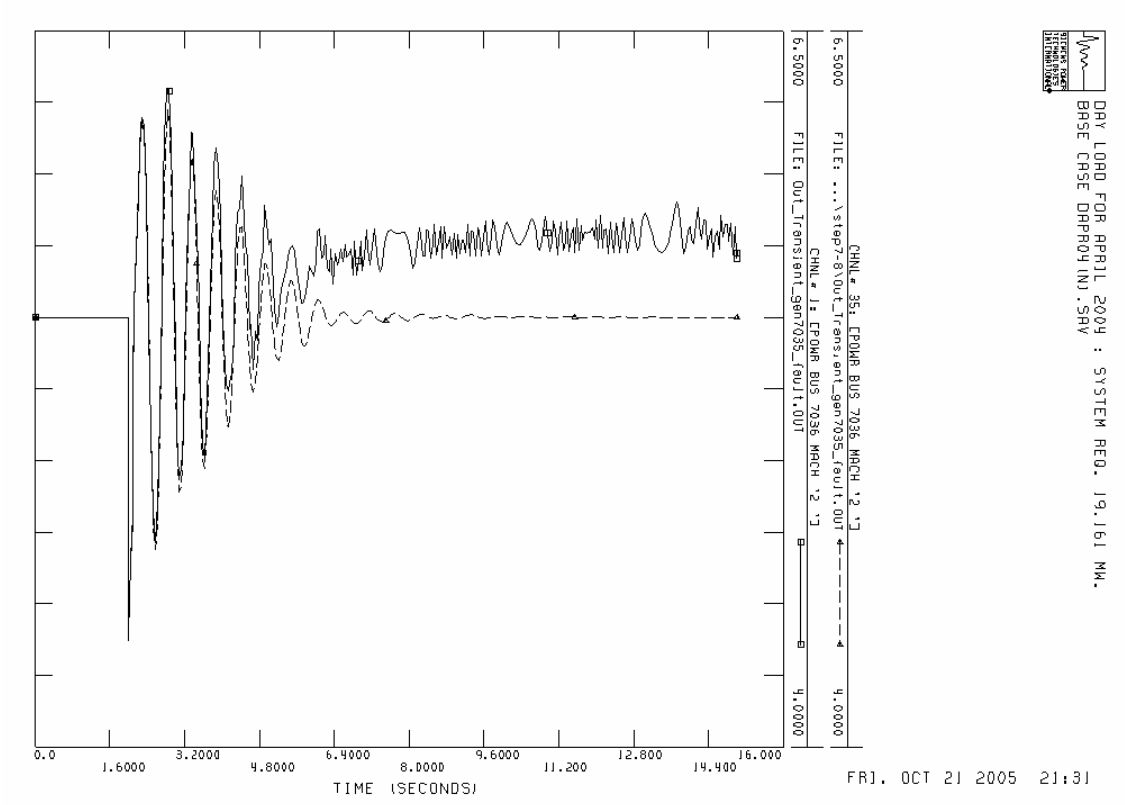
Case 3: For 7 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       16919 MW   7279 MVA
Total generation of the separated system: 17330 MW   4370 MVA
```

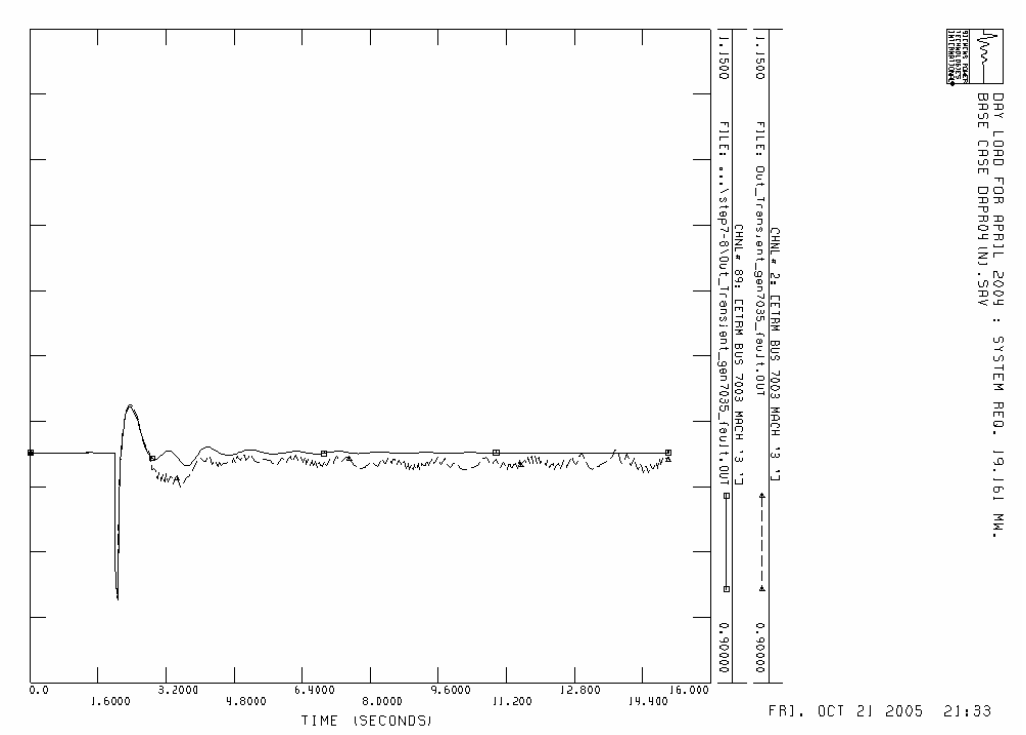
The simulation result is similar with the original system.

The compared graphs list below.

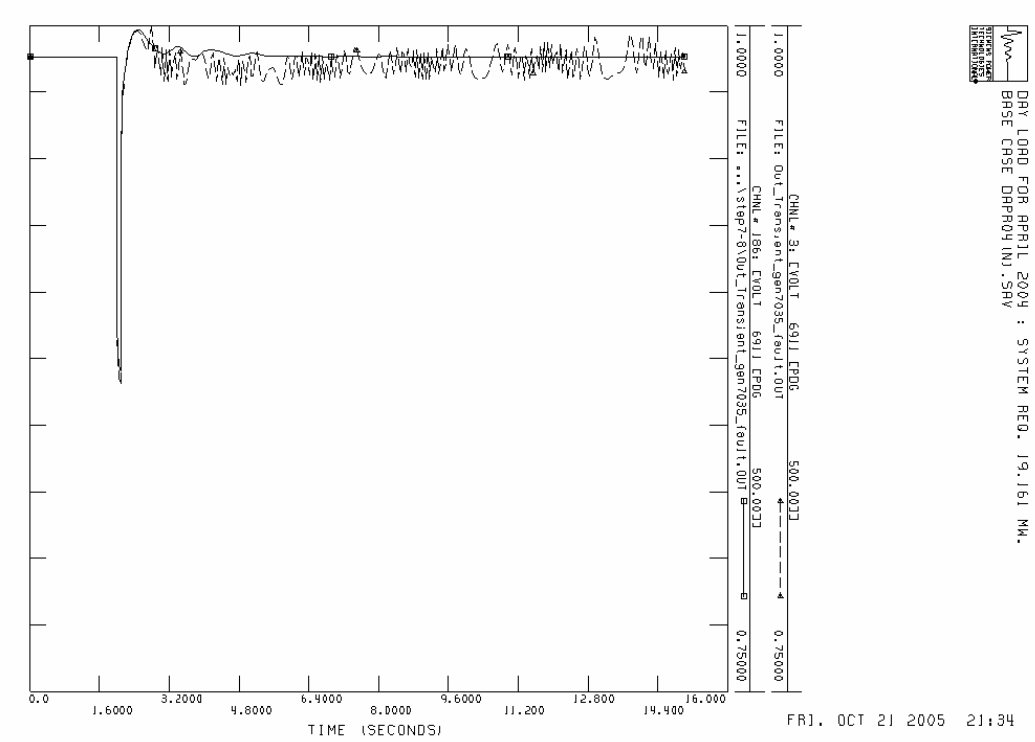
Power: Bus 7036 “2”



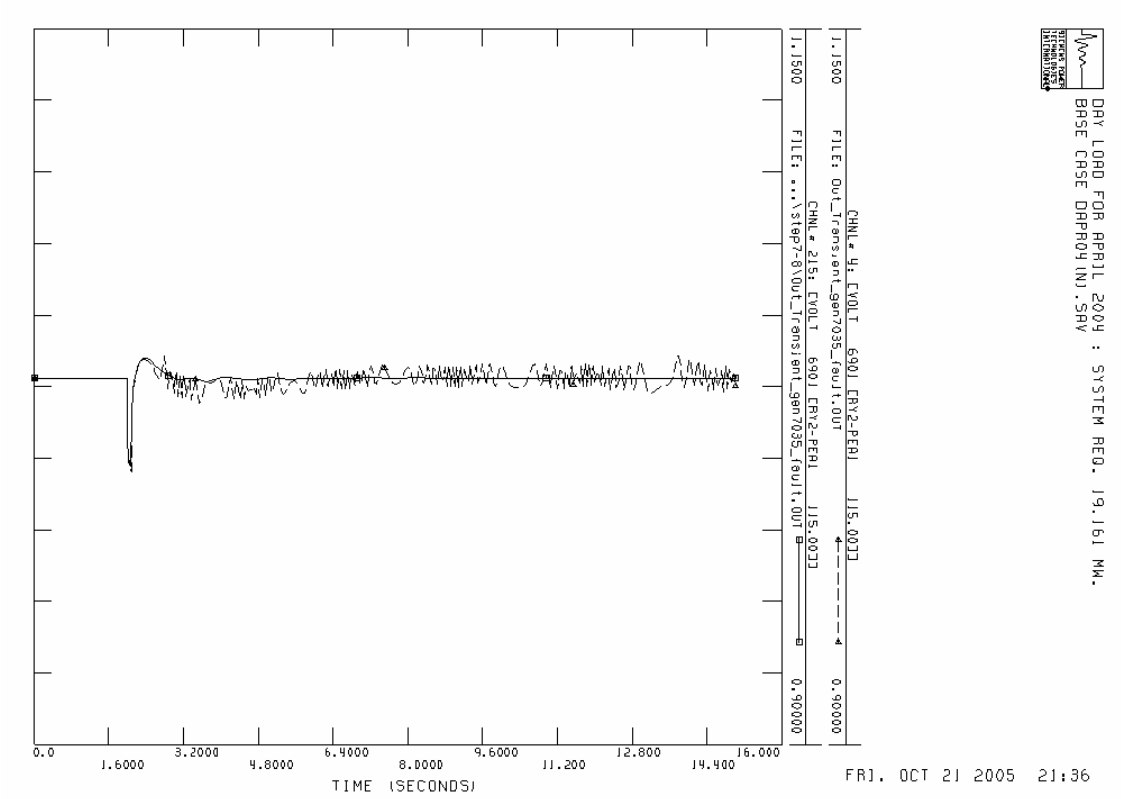

ETRM: Bus 7003 “3”




Voltage: Bus 6911




Voltage: Bus 6901




Case 4: For 6 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       16191 MW   6995 MVA
Total generation of the separated system: 16528 MW   4339 MVA
```

The simulation result is almost the same as the last case.

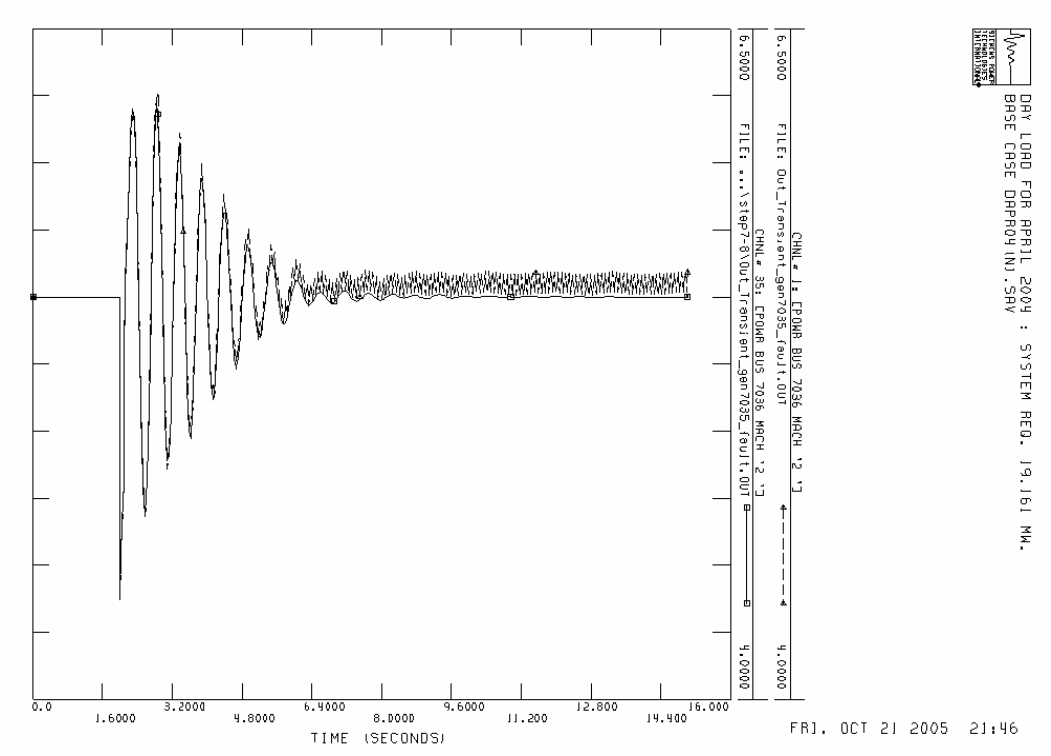
Case 5: For 5 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17297 MW   7349 MVA
Total generation of the separated system: 17670 MW   4632 MVA
```

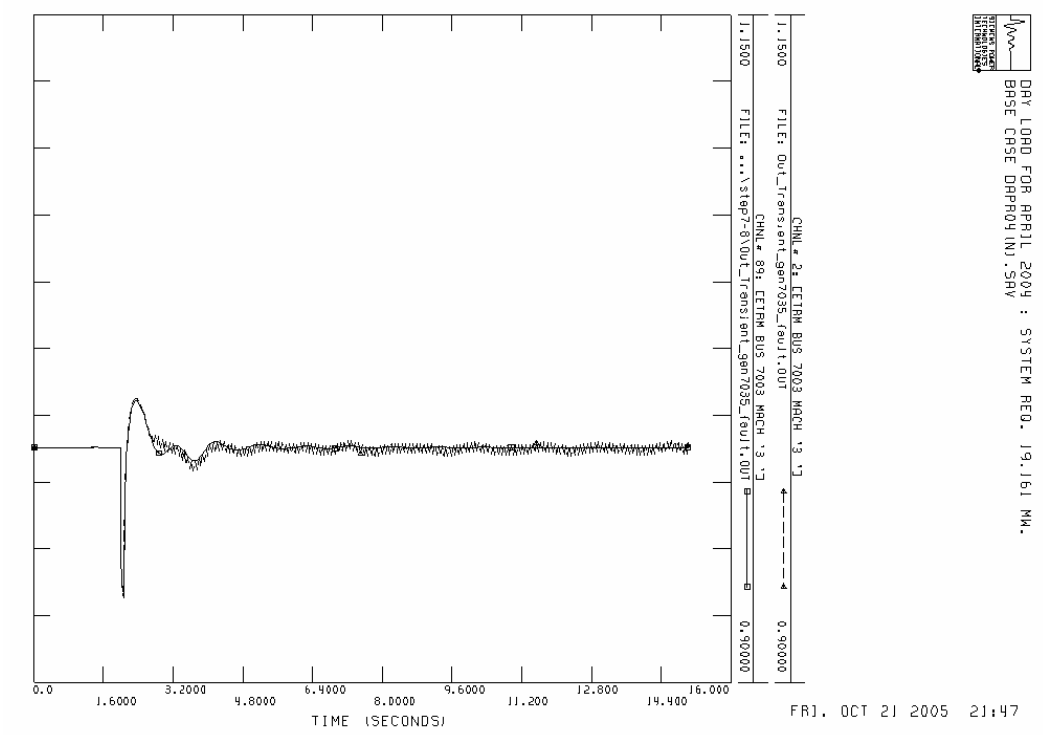
The simulation result is similar with the original system.

The compared graphs list below.

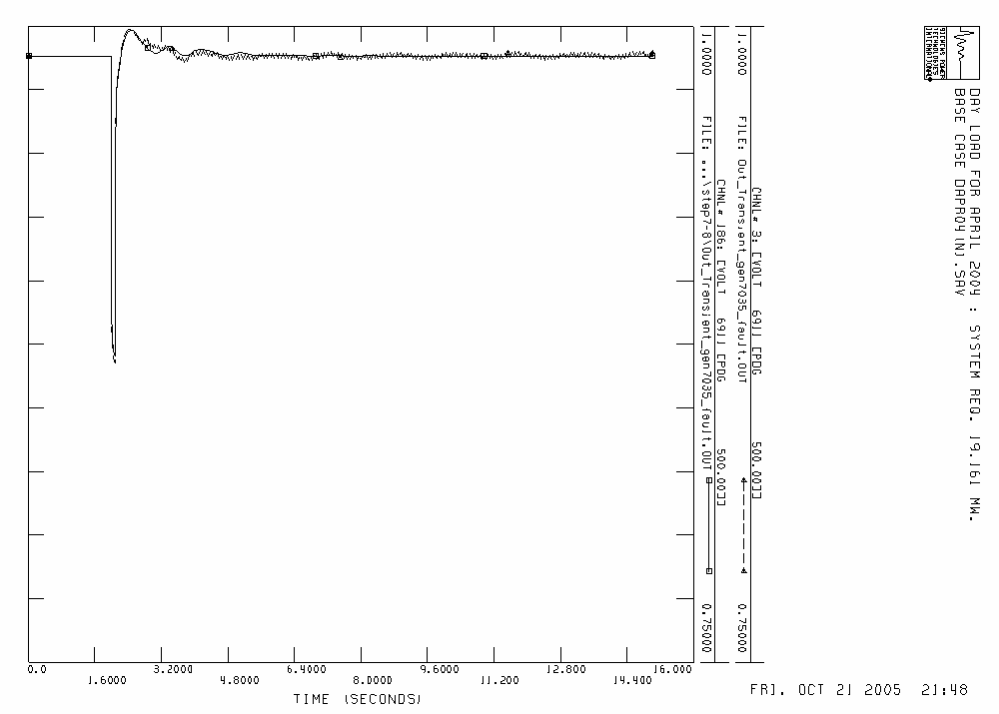
Power: Bus 7036 “2”



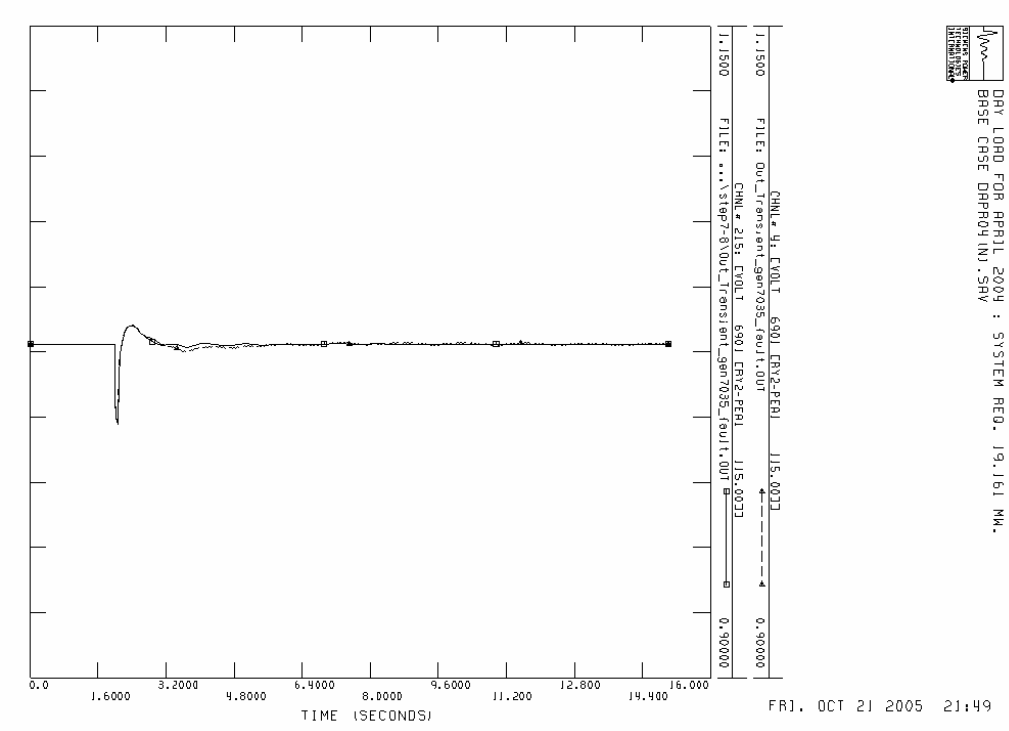

ETRM: Bus 7003 “3”



Voltage: Bus 6911




Voltage: Bus 6901



Step 8

Case 1: For 10 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       18393 MW   7945 MVA
Total generation of the separated system: 18868 MW   4678 MVA
```

The simulation result is almost the same as the original system.

Case 2: For 8 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17691 MW   7624 MVA
Total generation of the separated system: 18129 MW   4716 MVA
```

The simulation result is almost the same as the original system.

Case 3: For 7 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       16919 MW   7279 MVA
Total generation of the separated system: 17330 MW   4370 MVA
```

The simulation result is similar with the original system.

Case 4: For 6 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       16191 MW   6995 MVA
Total generation of the separated system: 16528 MW   4339 MVA
```

The simulation result is similar with the original system.

Case 5: For 5 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17297 MW   7349 MVA
Total generation of the separated system: 17670 MW   4632 MVA
```

The simulation result is similar with the original system.

Step 9

Case 1: For 10 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       18393 MW   7945 MVA
Total generation of the separated system: 18868 MW   4678 MVA
```

The simulation result is almost the same as the original system.

Case 2: For 8 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17487 MW   7647 MVA
Total generation of the separated system: 17925 MW   4648 MVA
```

The simulation result is almost the same as the original system.

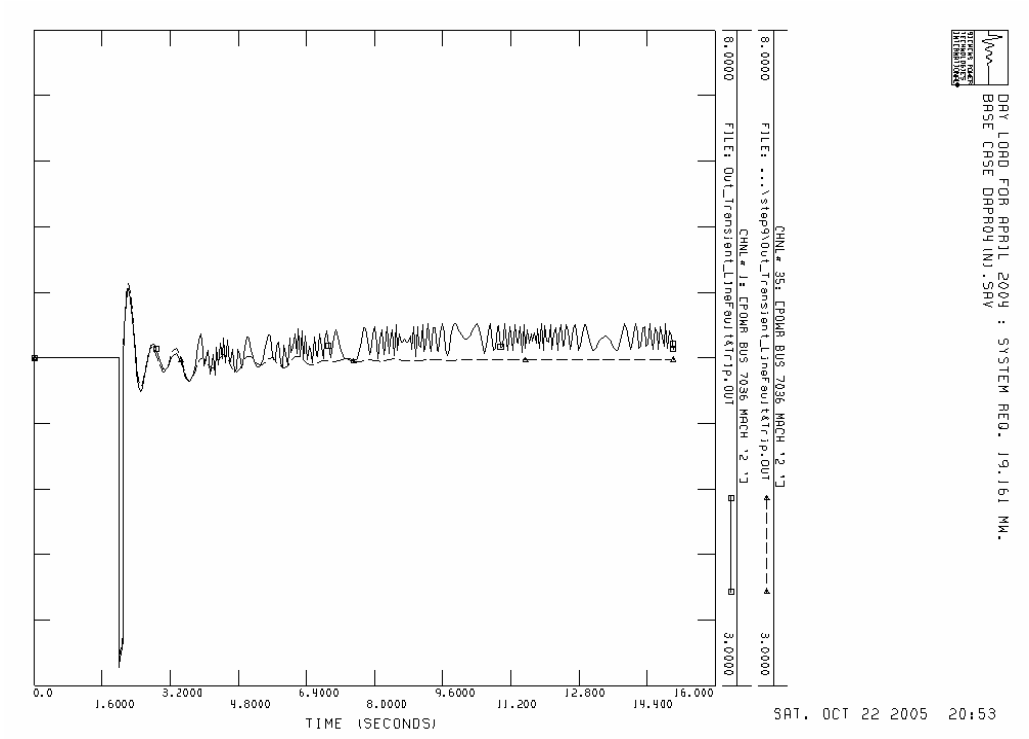
Case 3: For 7 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       17374 MW   7596 MVA
Total generation of the separated system: 17799 MW   4635 MVA
```

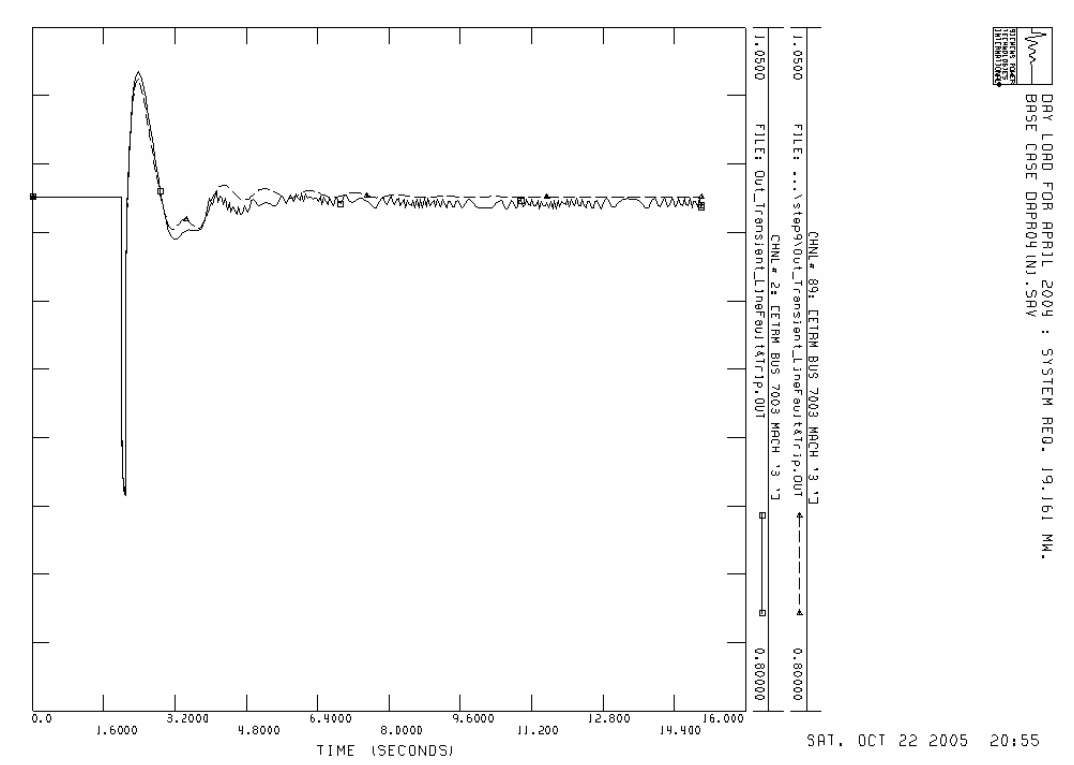
The simulation result is similar with the original system.

The compared graphs list below.

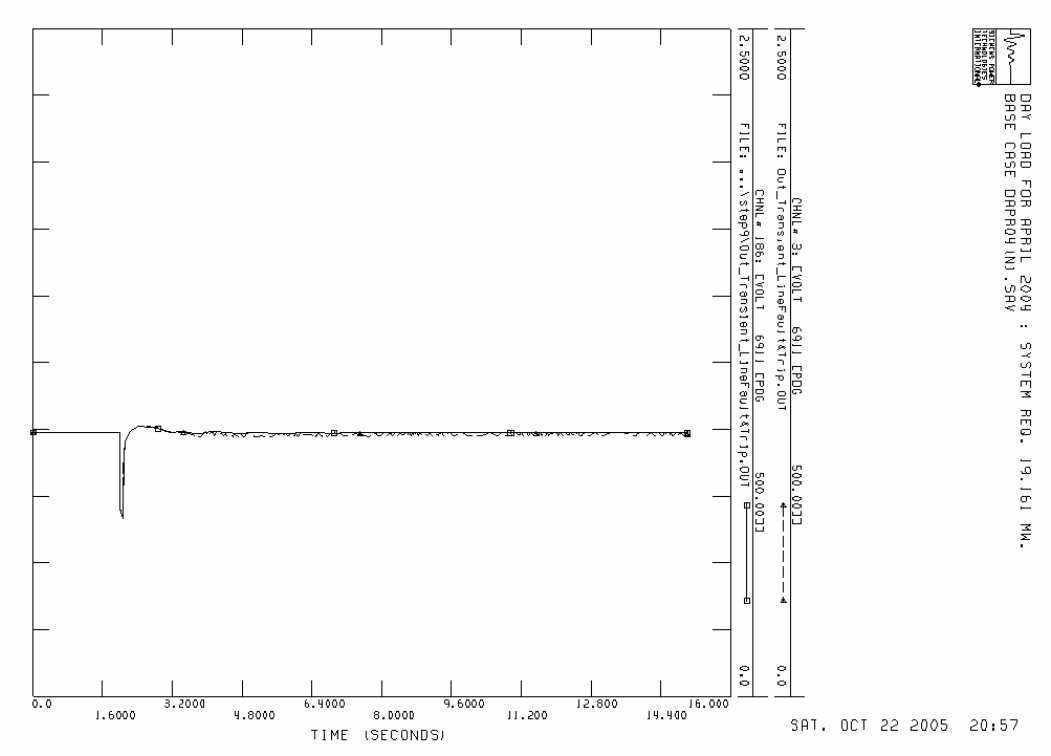
Power: Bus 7036 “2”



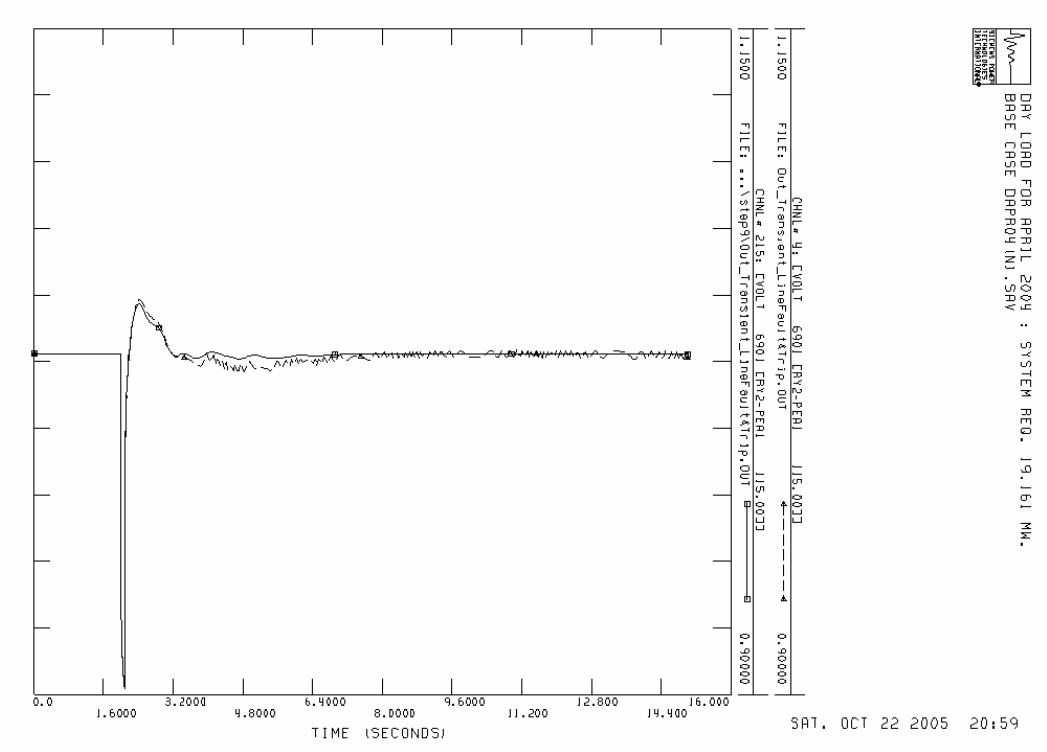

ETRM: Bus 7003 “3”



Voltage: Bus 6911




Voltage: Bus 6901



Case 4: For 6 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       16876 MW   7302 MVA
Total generation of the separated system: 17237 MW   4600 MVA
```

The simulation result is similar with the previous system.

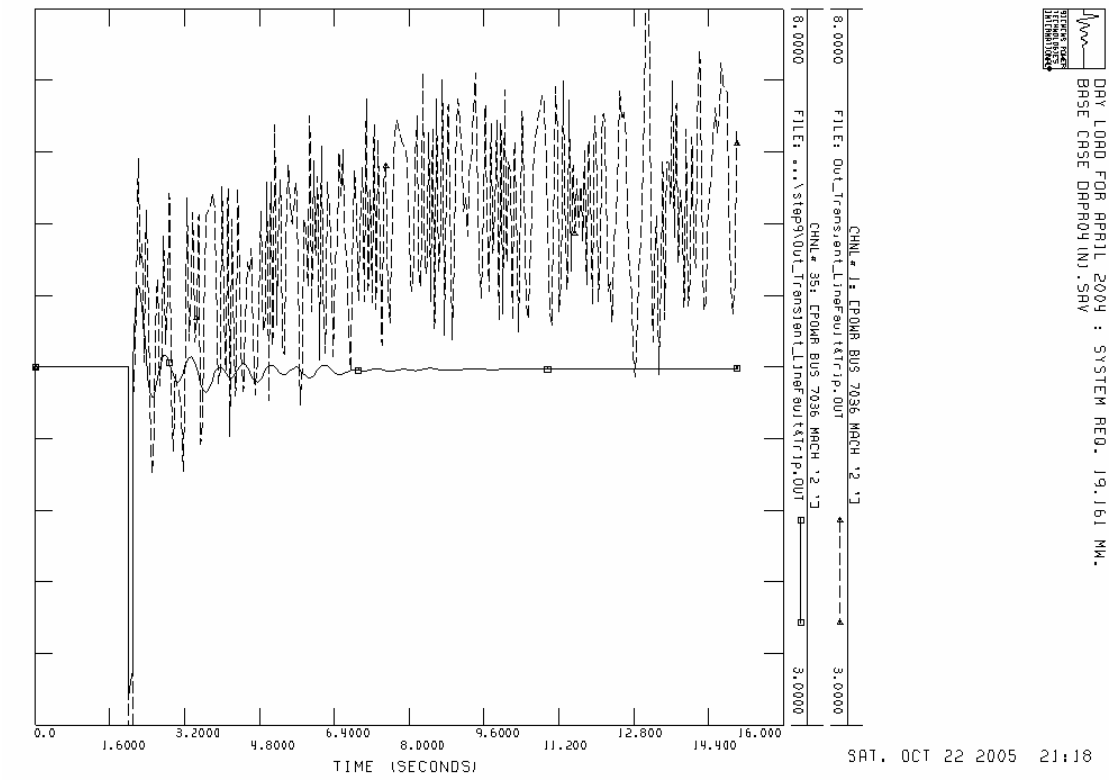
Case 5: For 5 layers of buses near the fault:

```
Total load of the original base system:       18710 MW   8223 MVA
Total generation of the original base system: 19214 MW   4696 MVA

Total load of the separated system:       15427 MW   6730 MVA
Total generation of the separated system: 15711 MW   4506 MVA
```

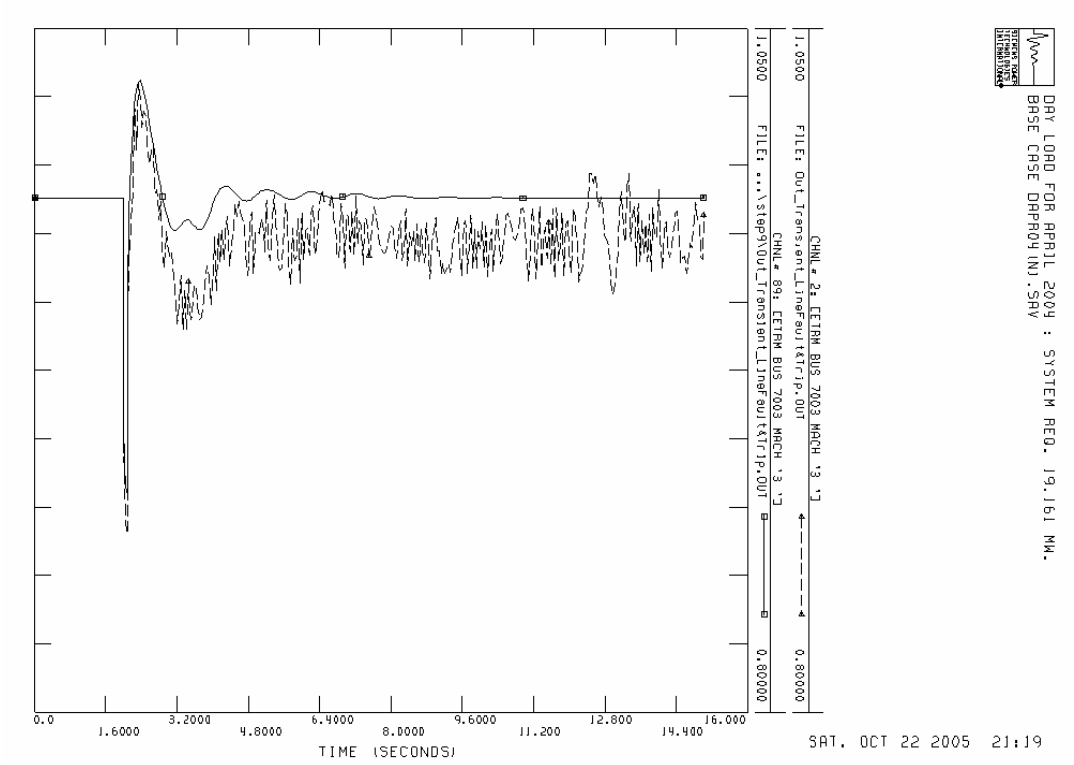
The simulation result is totally different from the original system.

The compared graphs list below.

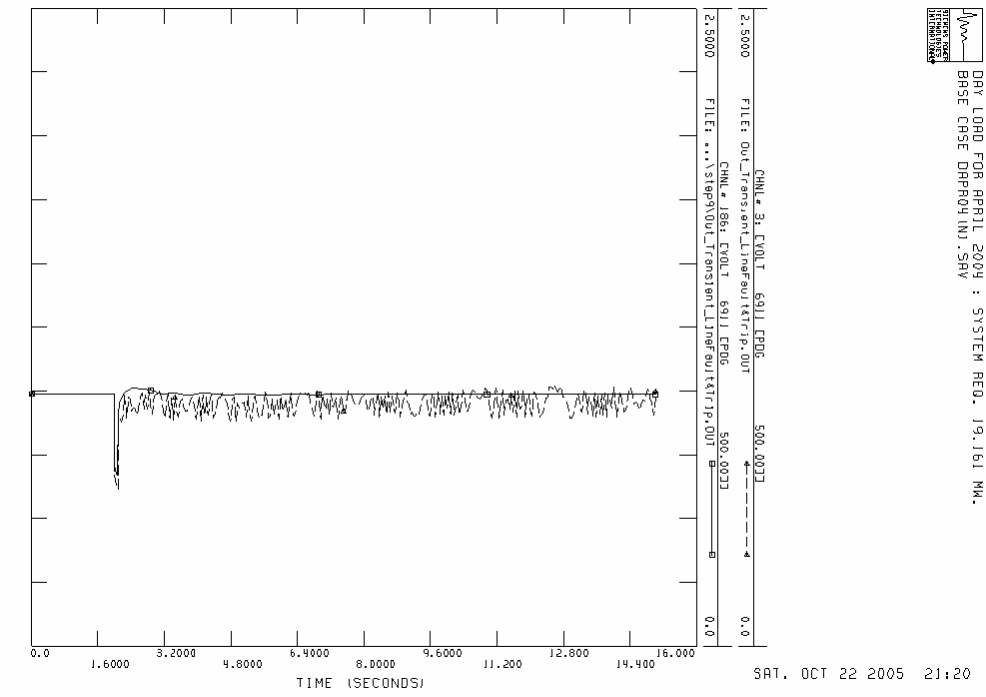
Power: Bus 7036 “2”



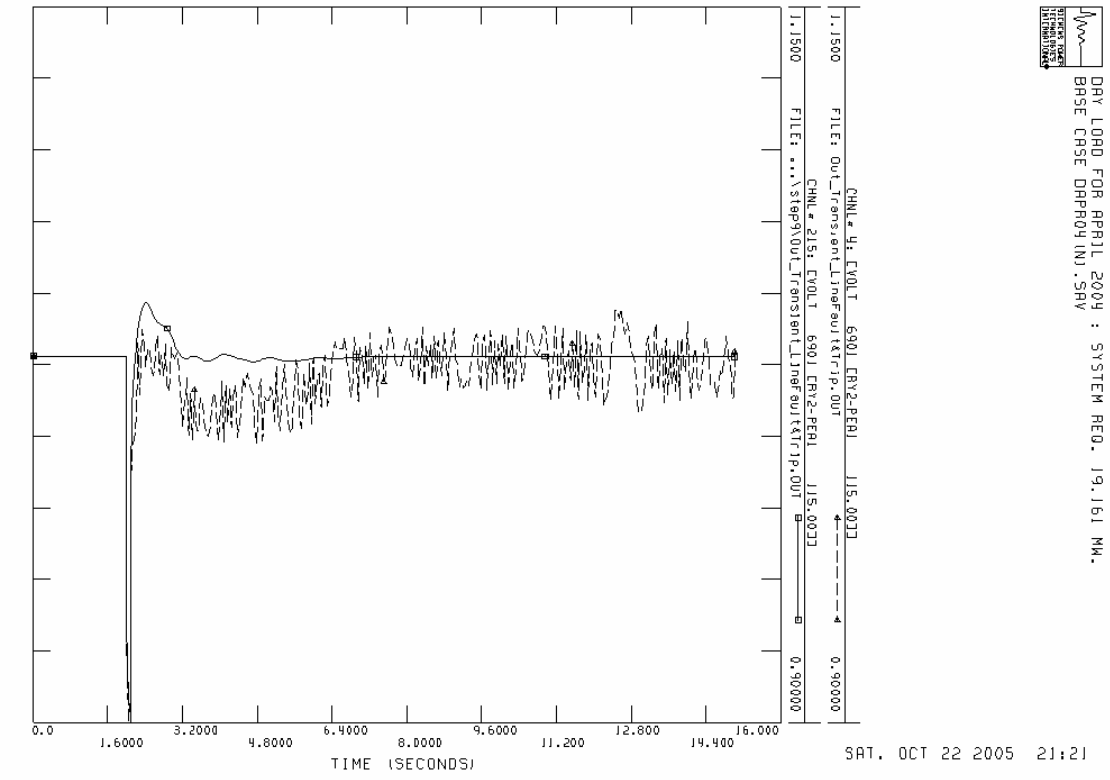
ETRM: Bus 7003 “3”




Voltage: Bus 6911



Voltage: Bus 6901



# APPENDIX B

# THE DETAIL SIMULATION RESULTS OF SPP SYSTEM

The detail stability simulation step for SPP system shows below.

1_ts05s2x_ori_solve.sav: A reasonable system base case raw data.

2_nerc-all-NO-SPP.dyr, 2_Ts-05SP.dyr: The system dynamic model.

3_PowerFlow_CONLG.py , 3_CONVLOADS-GENS.idv: Use python program to divide the system. Then use idv file to run power flow in dynamic environment.

4_SPPsystem_initial.idv: No fault test to check the integrity of the data.

5_SPPsystem_F54121.idv: A three-phase short circuit fault on the load bus 54121.

6_SPPsystem_F87456.idv: A three-phase short circuit fault on the generator bus 87456.

7_SPPsystem_F64895.idv: A three-phase short circuit fault on the load bus 64895.

8_SPPsystem_F31426.idv: A three-phase short circuit fault on the load bus 31426.

9_SPPsystem_F25044.idv: A three-phase short circuit fault on the generator bus 25044.

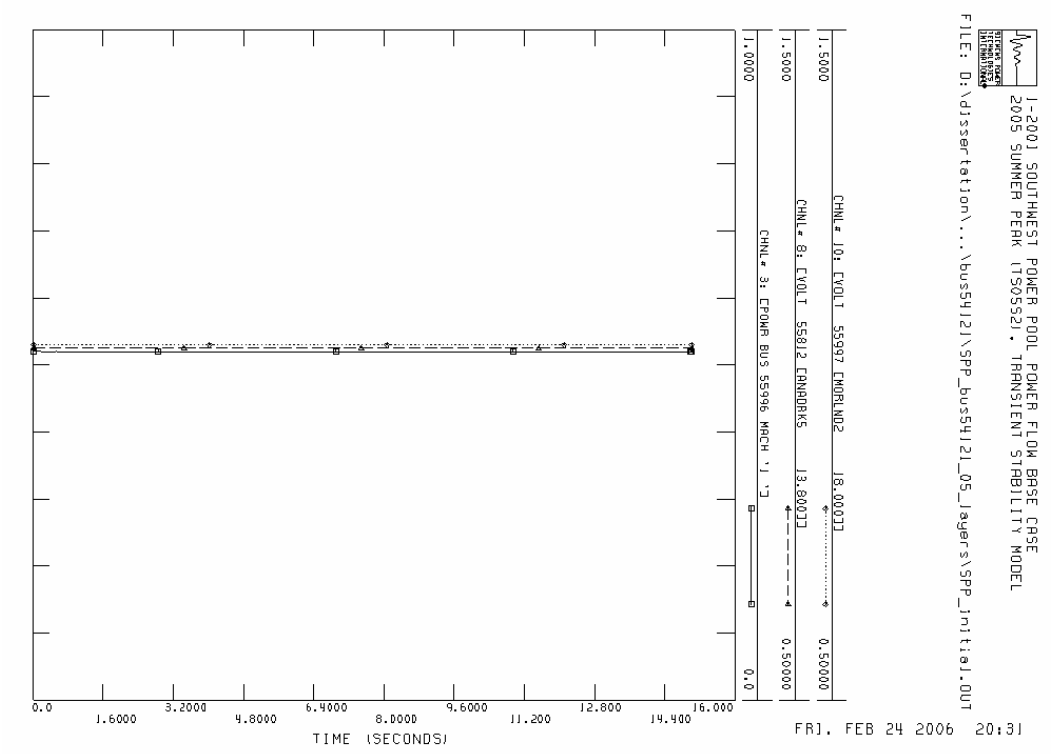
10_SPPsystem_F83904.idv: A three-phase short circuit fault on the load bus 83904.

After simulated all stability process, compare the whole process result below.

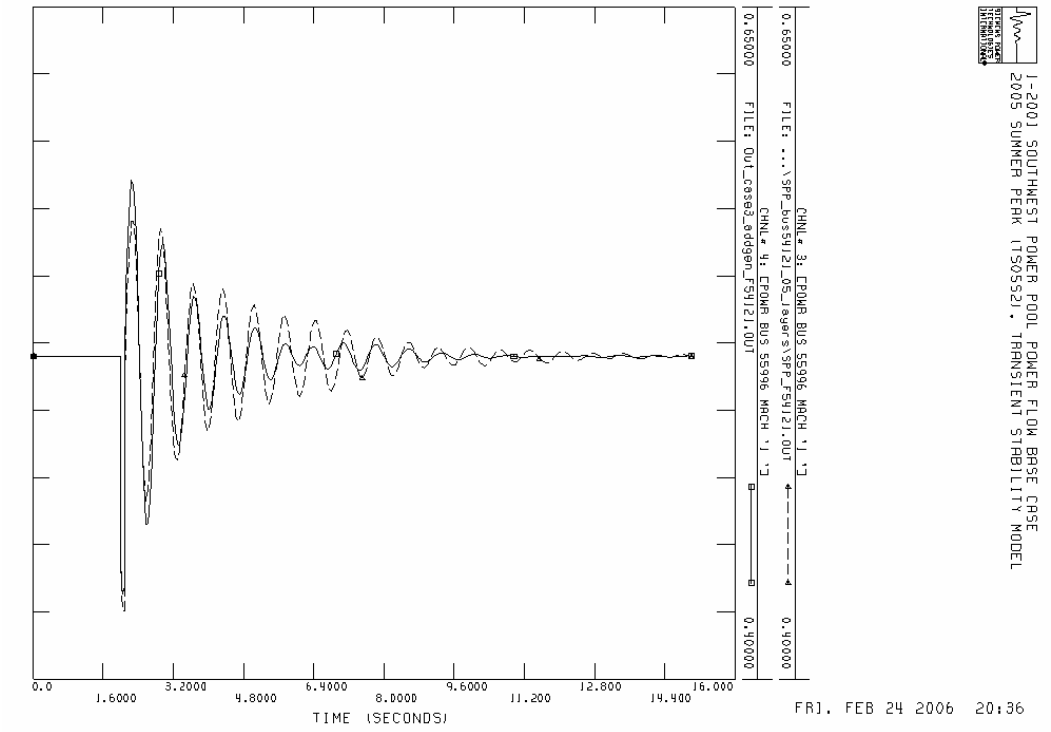
| How many layer of buses near the fault? | 30 | 15 | 8 | 5 |
|---|---|---|---|---|
| Step 5 | O | O | V | V |
| Step 6 | O | V | X | X |
| Step 7 | O | O | V | V |
| Step 8 | O | O | O | O |
| Step 9 | O | O | O | O |
| Step 10 | O | V | V | X |

Step 5: SPP_bus54121_5_layers

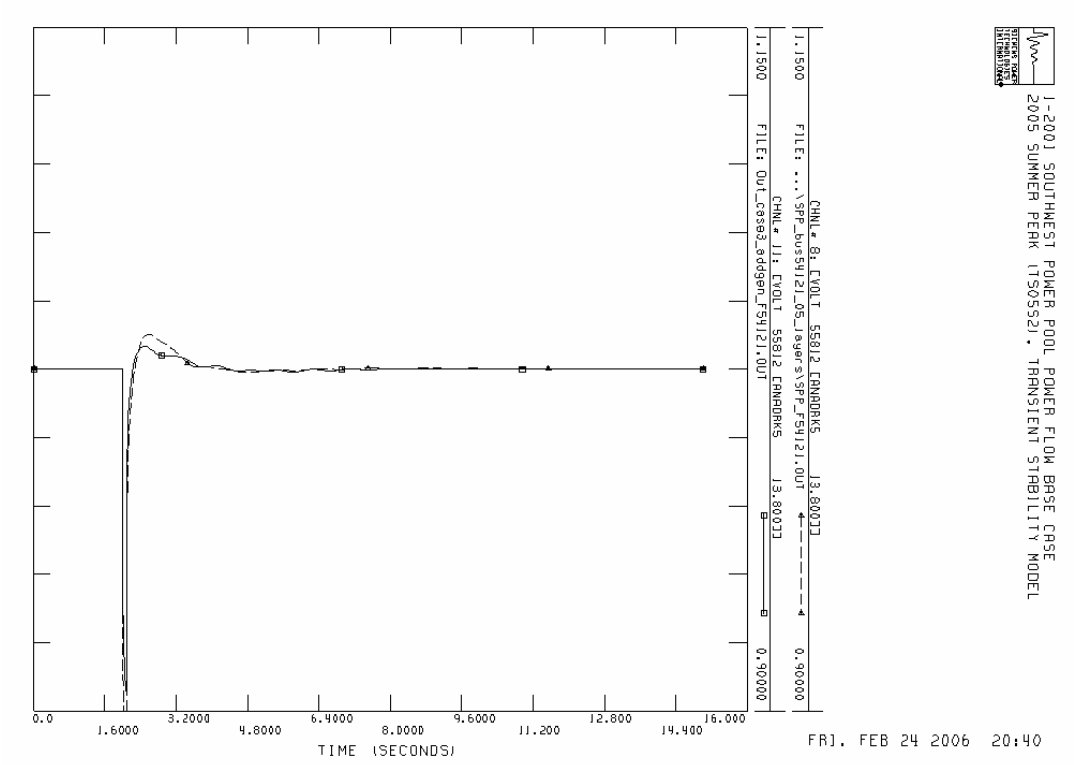
Initial condition for channel 3,8,10



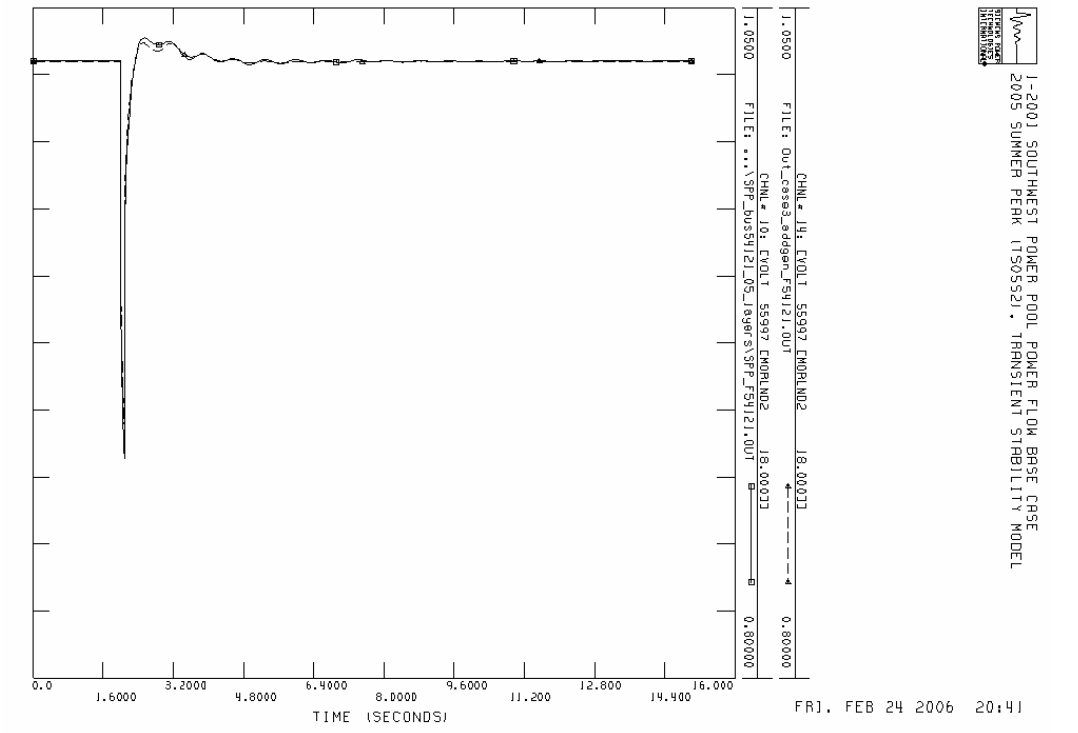

Channel 3: Power bus 55996 MACH '1'



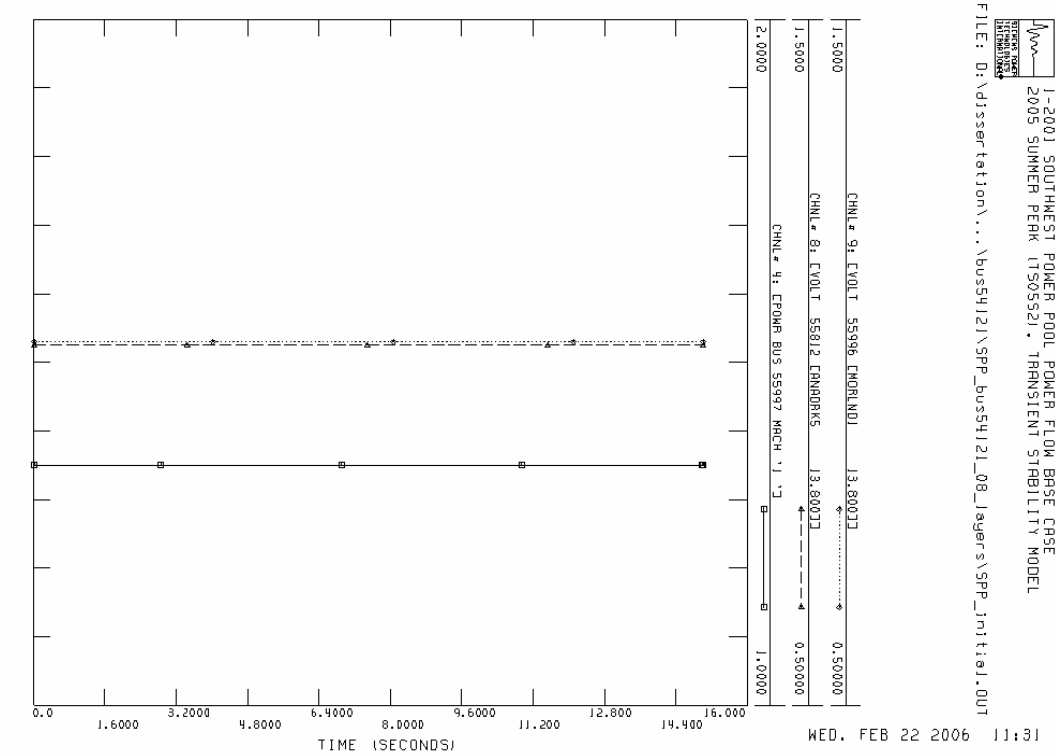
Channel 8: Voltage 55812 (13.8 kV)



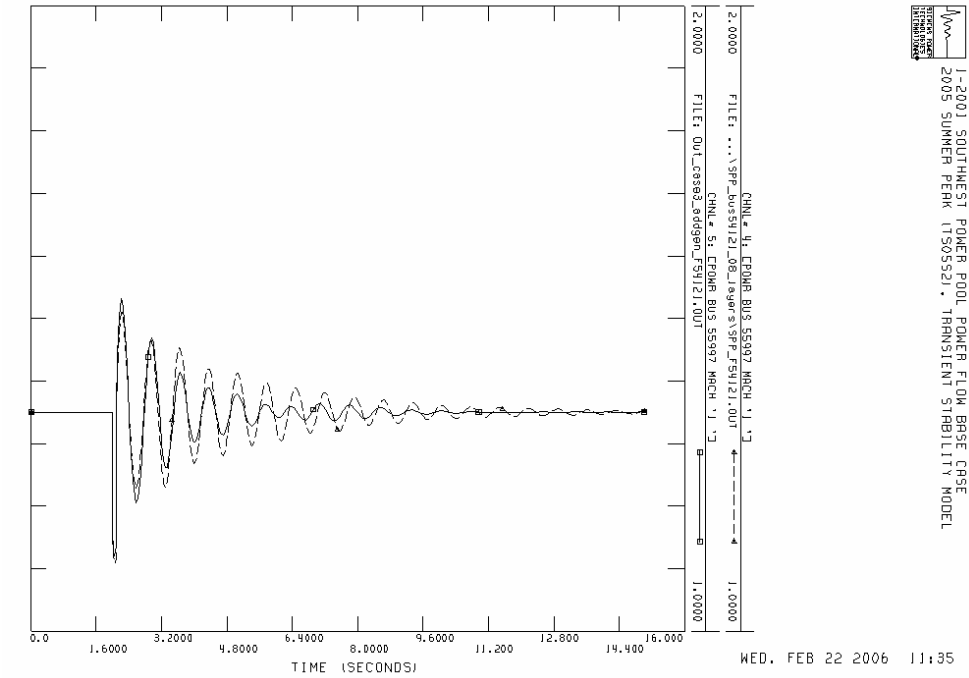

Channel 10: Voltage 55997 (18 kV)



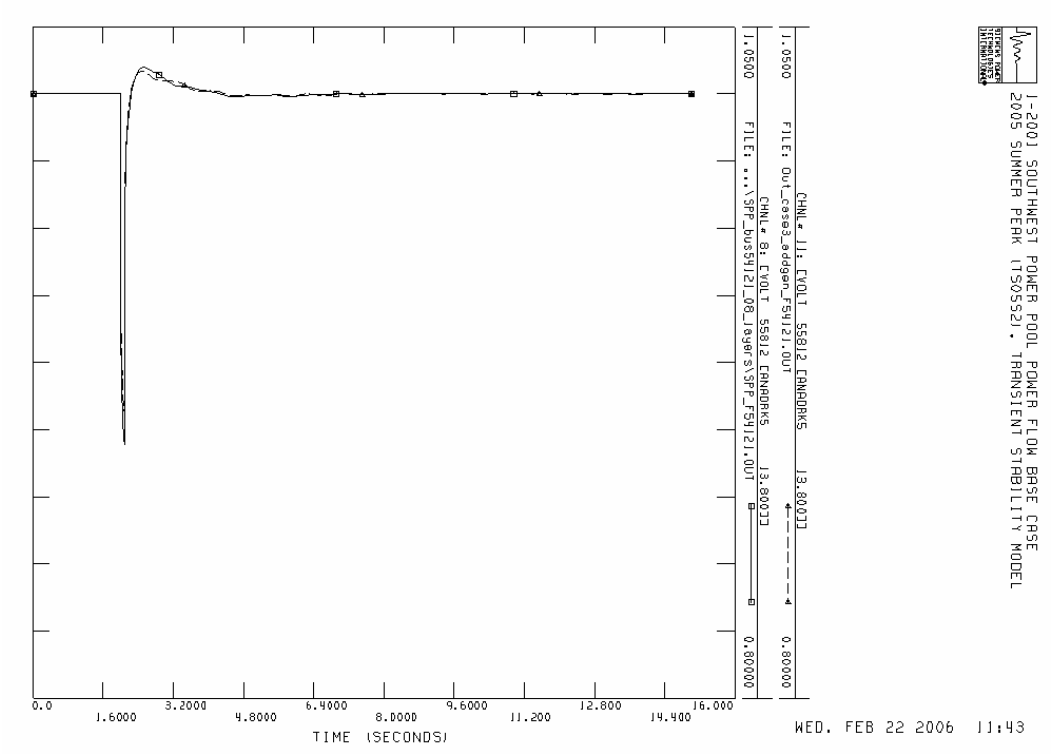
SPP_bus54121_8_layers
Initial condition for channel 4,8,9



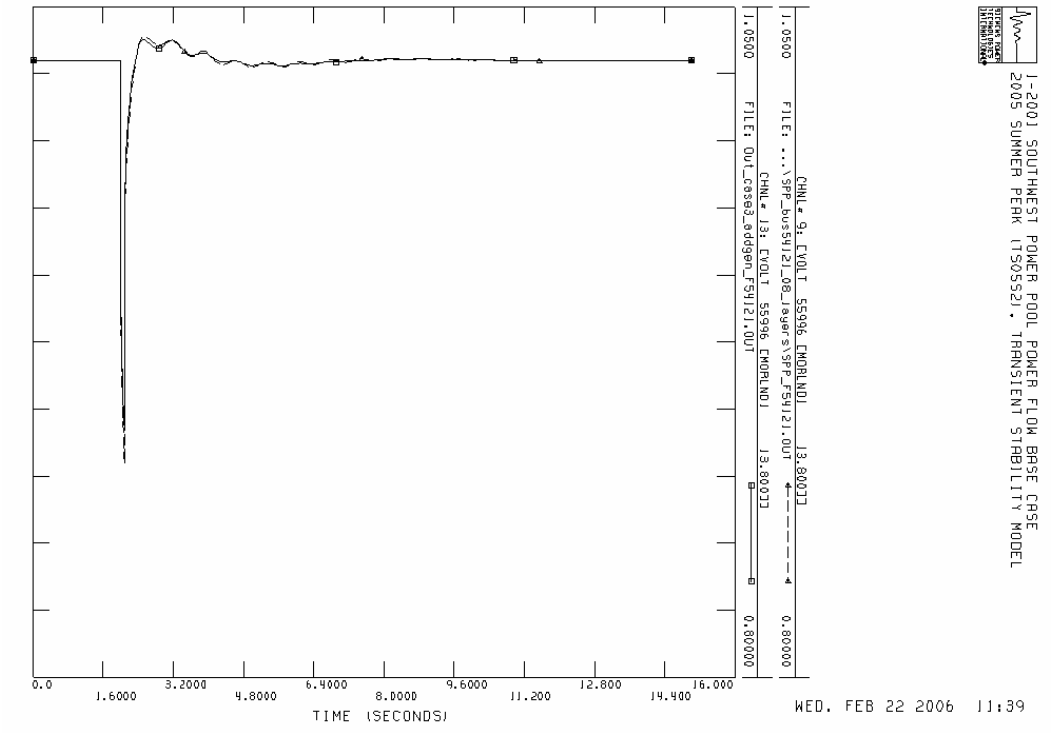

Channel 4: Power bus 55997 MACH '1'



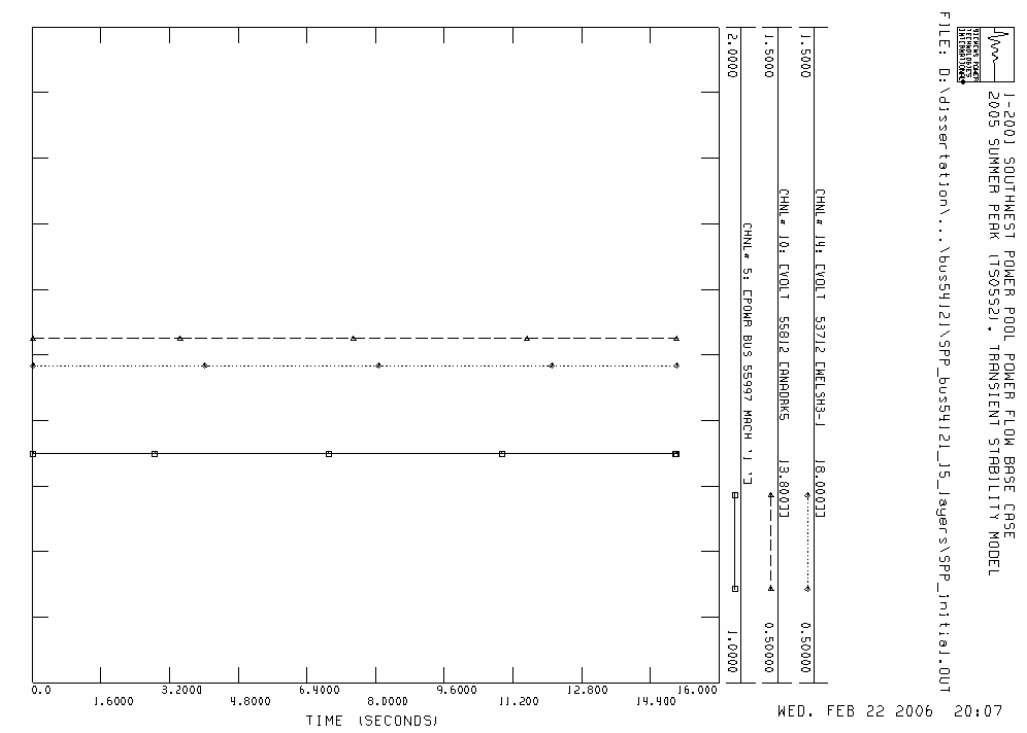
Channel 8: Voltage 55812 (13.8 kV)



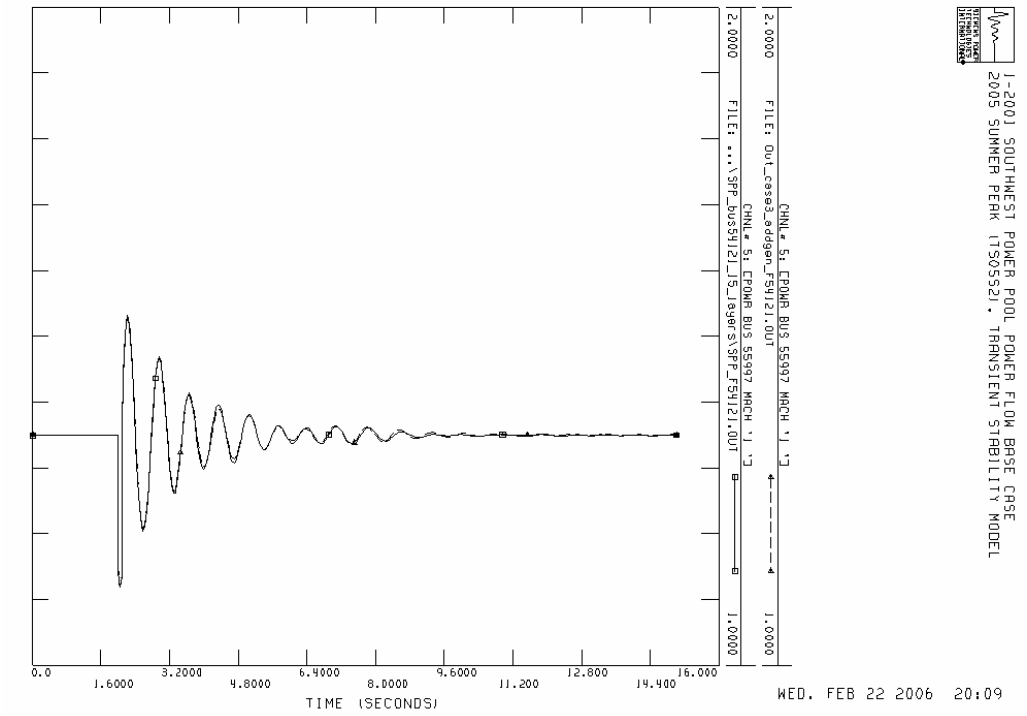

Channel 9: Voltage 55996 (18 kV)



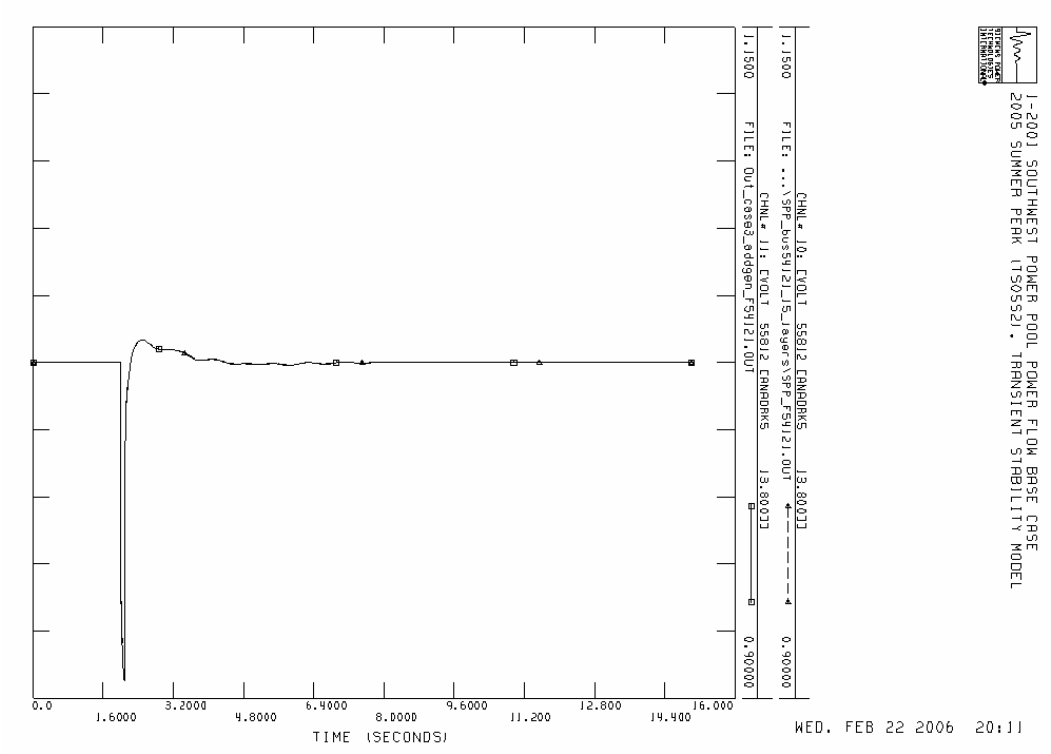
SPP_bus54121_15_layers
Initial condition for channel 5,10,14



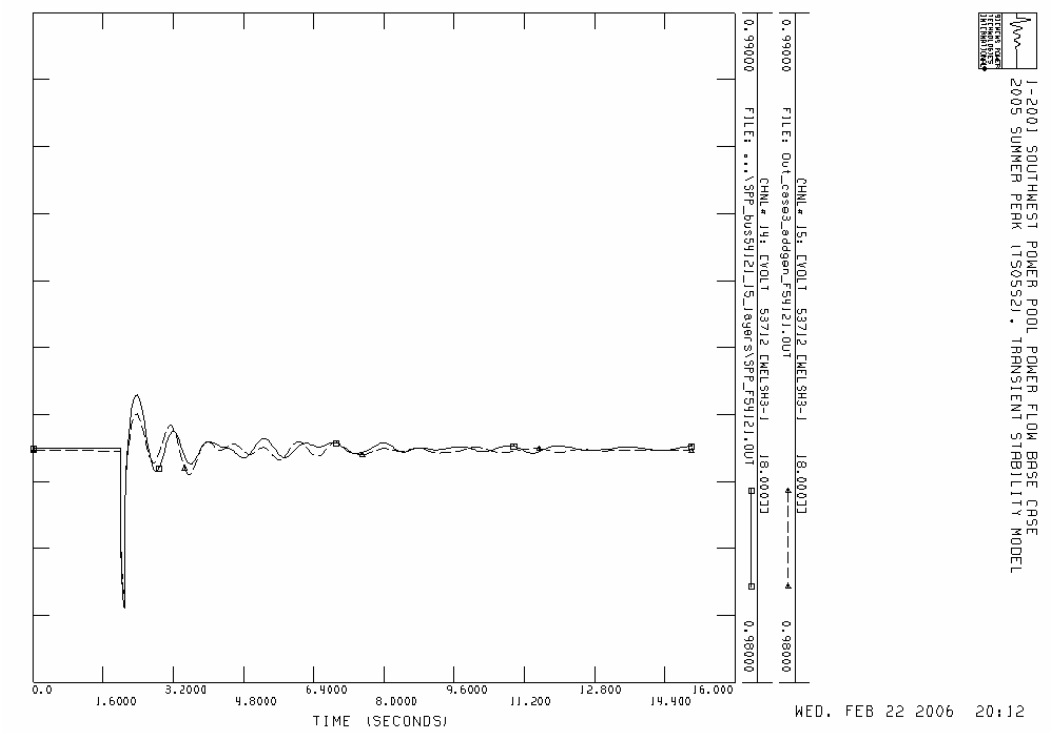

Channel 5: Power bus 55997 MACH '1'



Channel 10: Voltage 55812 (13.8 kV)

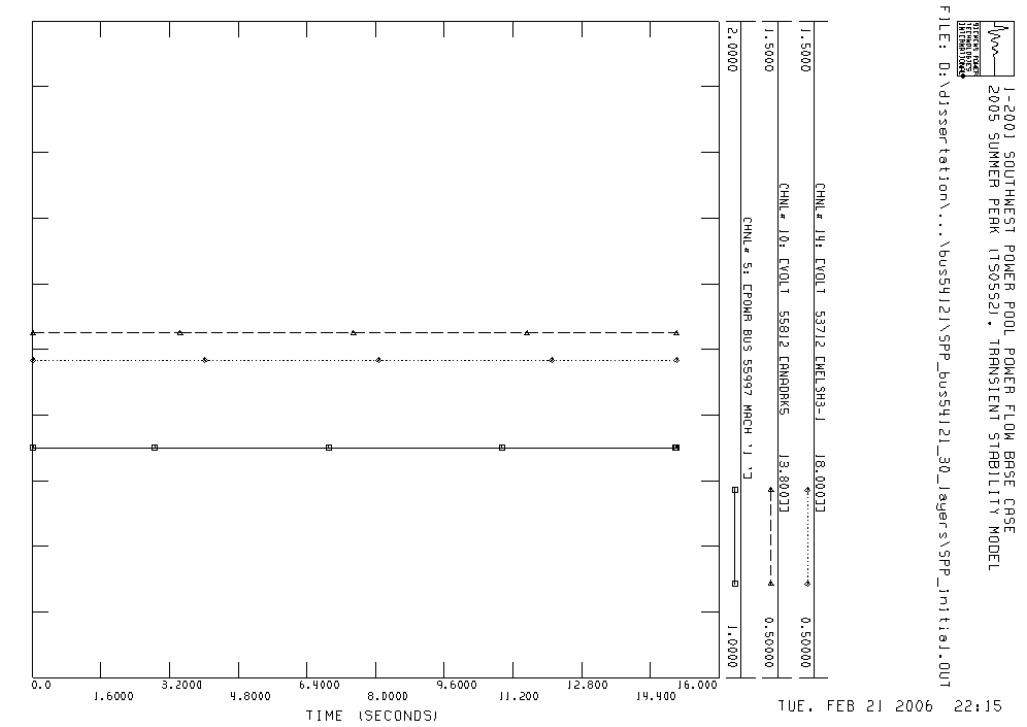



Channel 14: Voltage 53712 (18 kV)

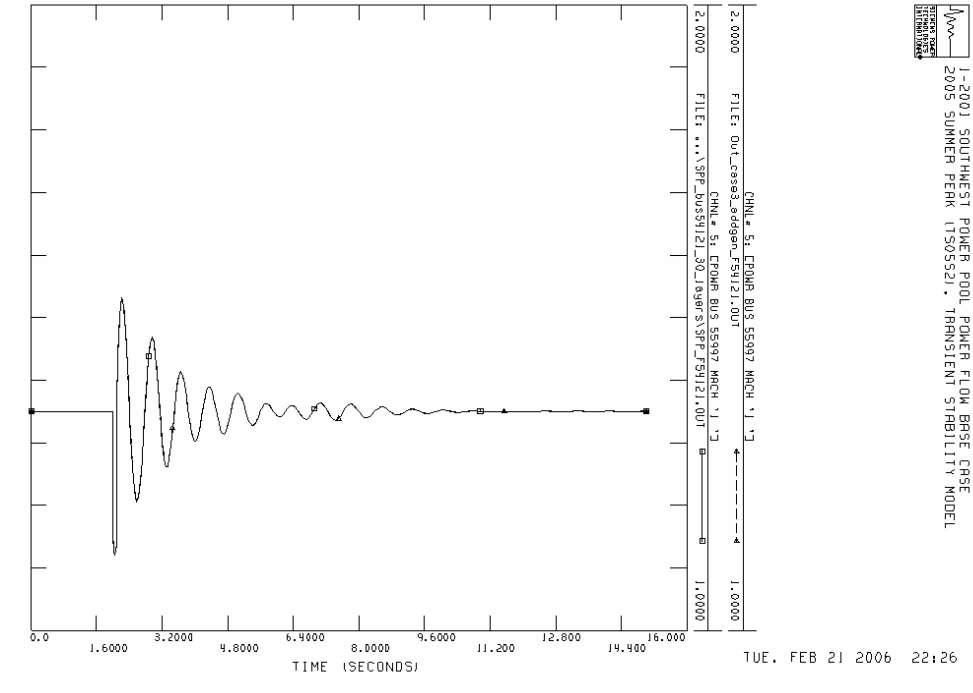


SPP_bus54121_30_layers

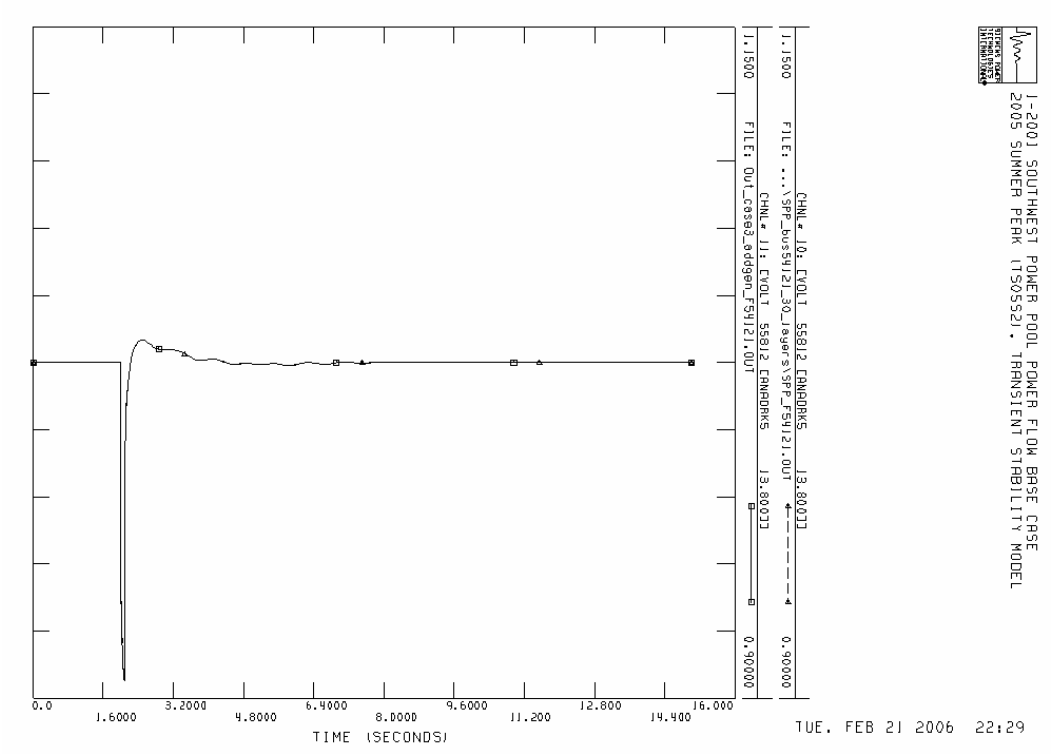
Initial condition for channel 5,10,14



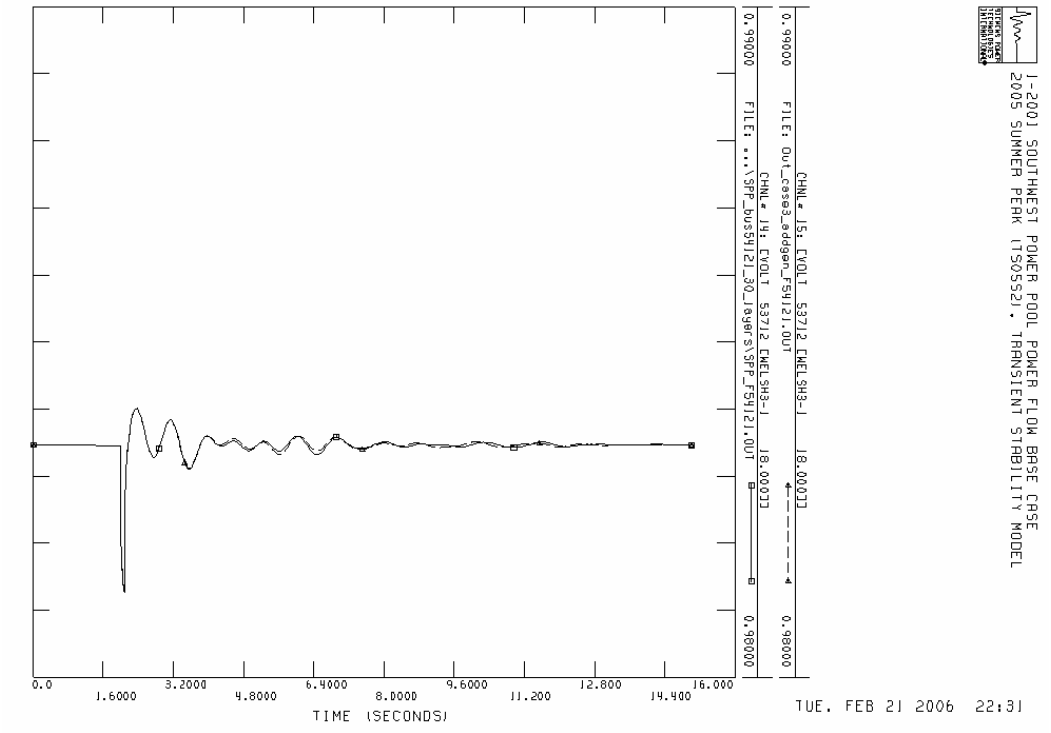

Channel 5: Power bus 55997 MACH ‘1’



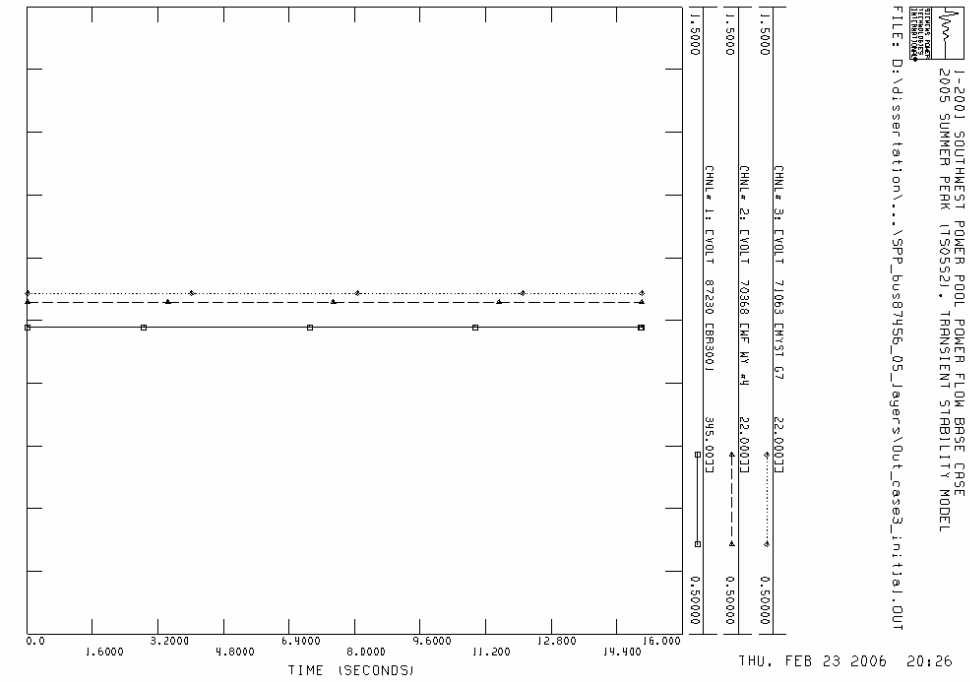
Channel 10: Voltage 55812 (13.8 kV)



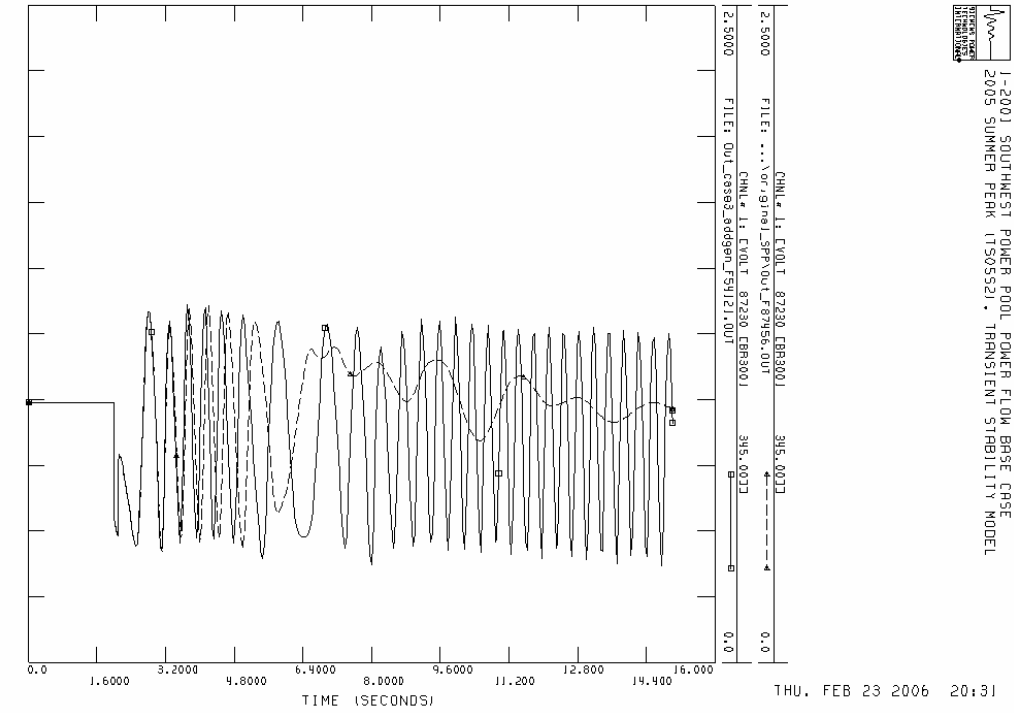

Channel 14: Voltage 53712 (18 kV)



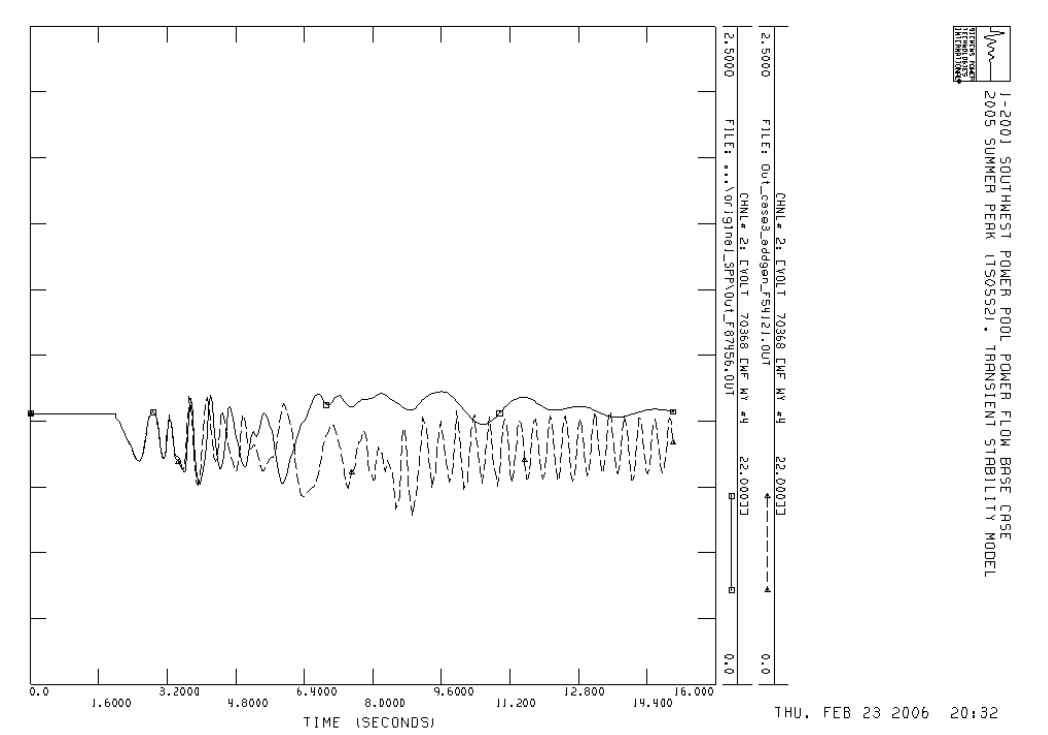
SPP_bus87456_05_layers
Initial condition for channel 1,2,3



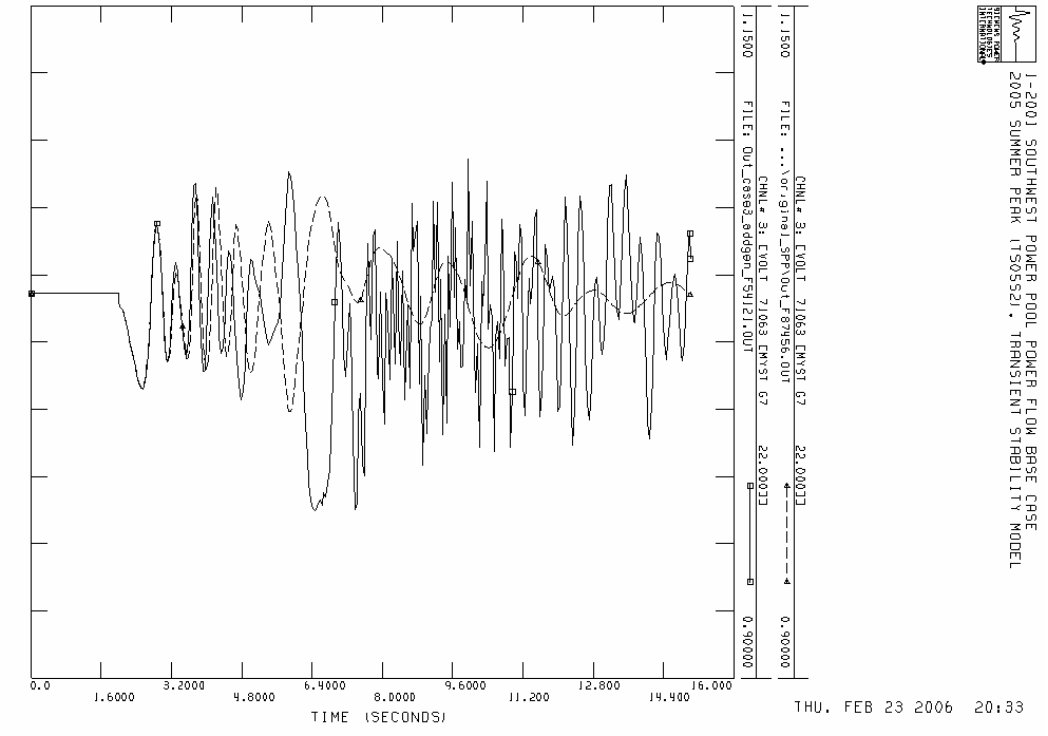

Channel 1: Voltage 87230 (345 kV)



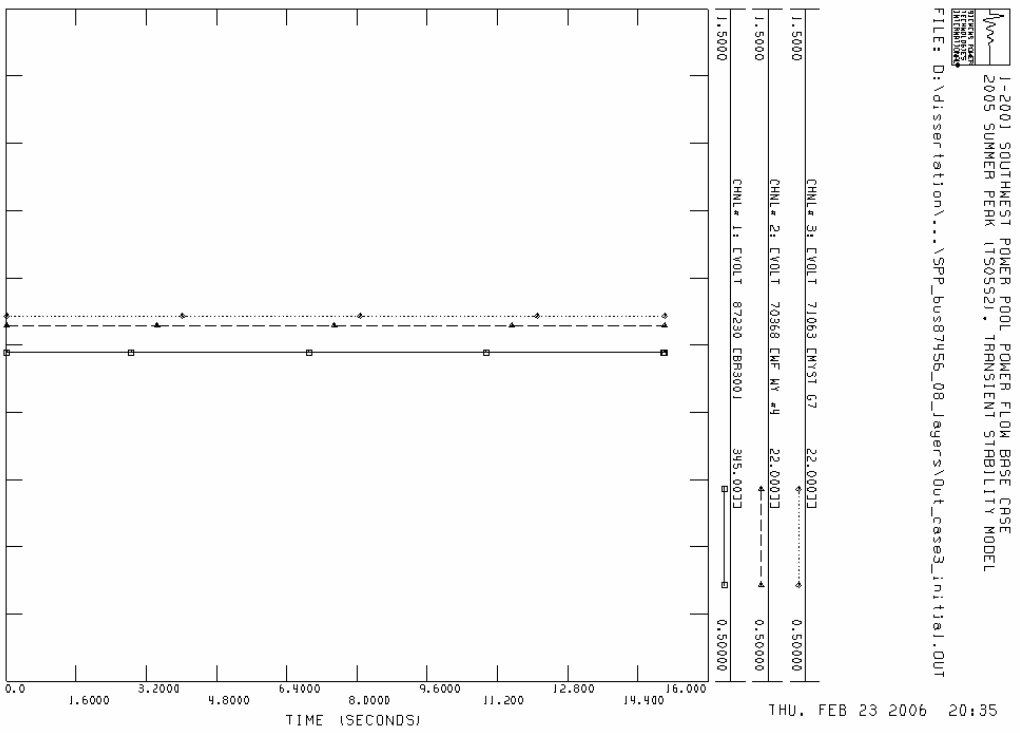
Channel 2: Voltage 70368 (22 kV)



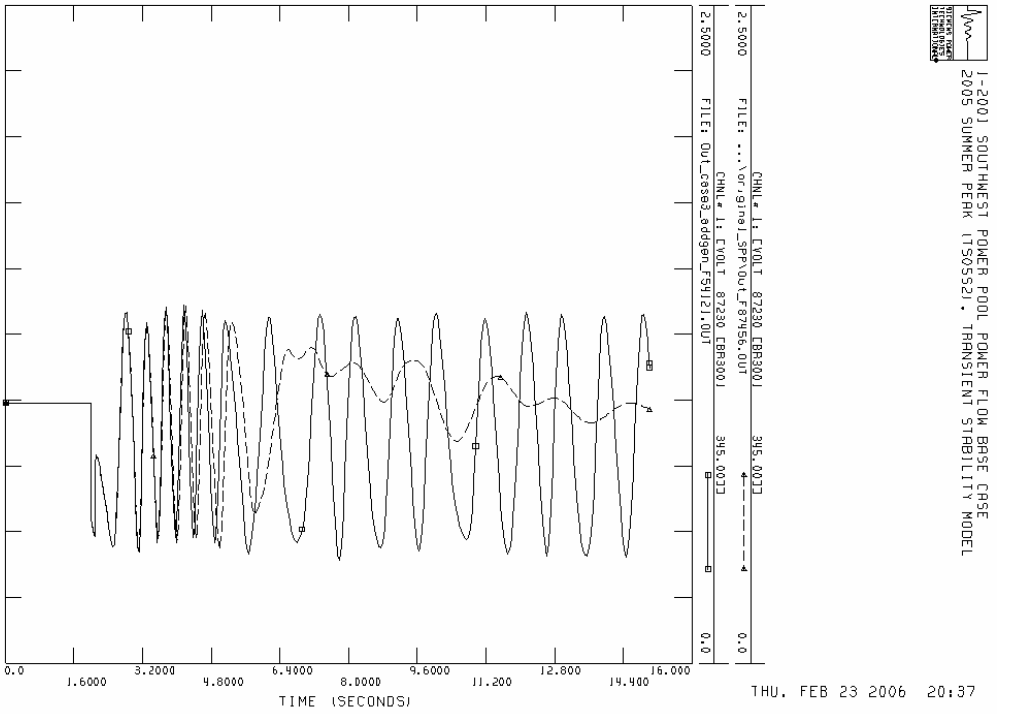

Channel 3: Voltage 71063 (22 kV)



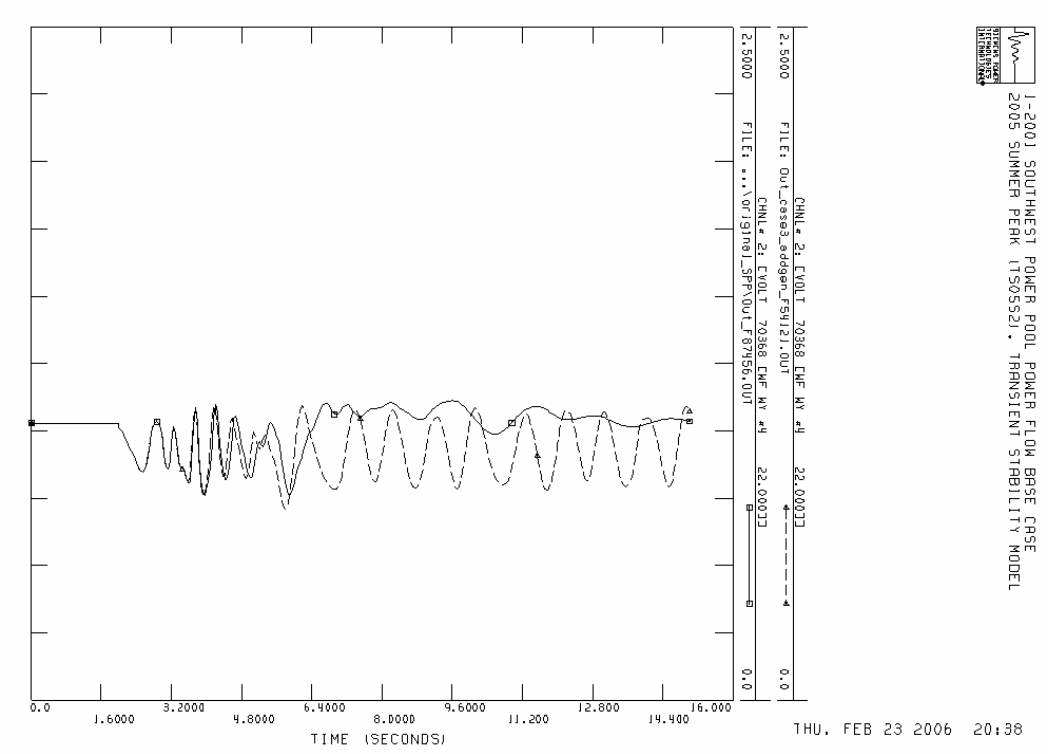
SPP_bus87456_08_layers
Initial condition for channel 1,2,3



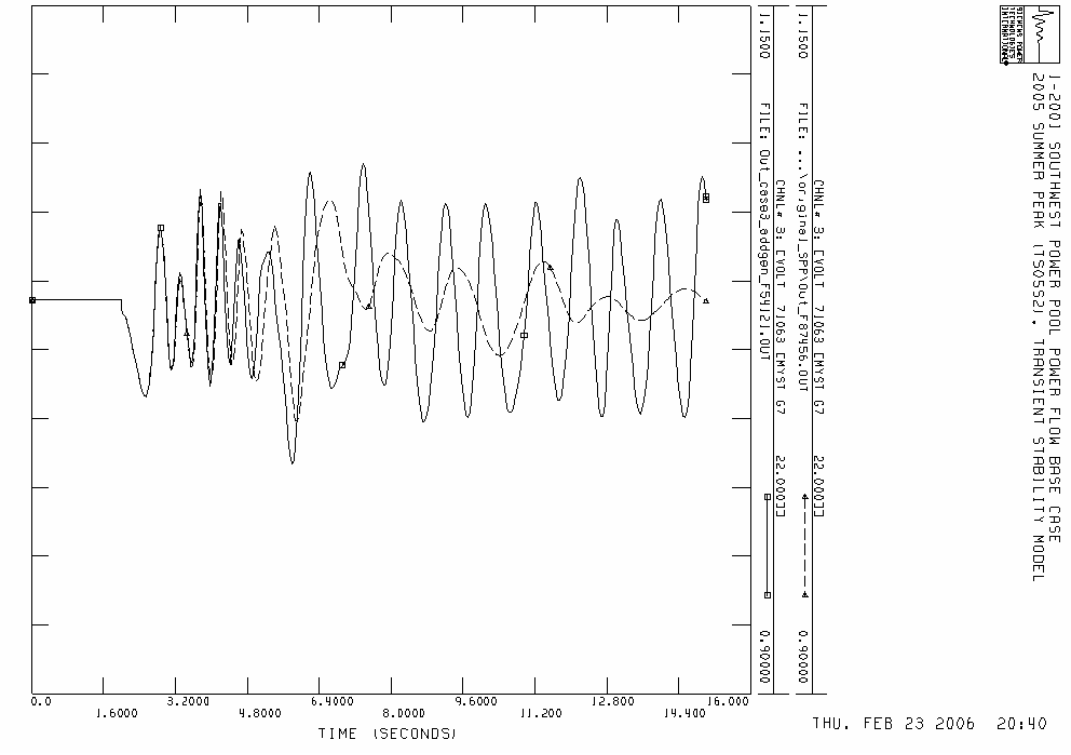

Channel 1: Voltage 87230 (345 kV)



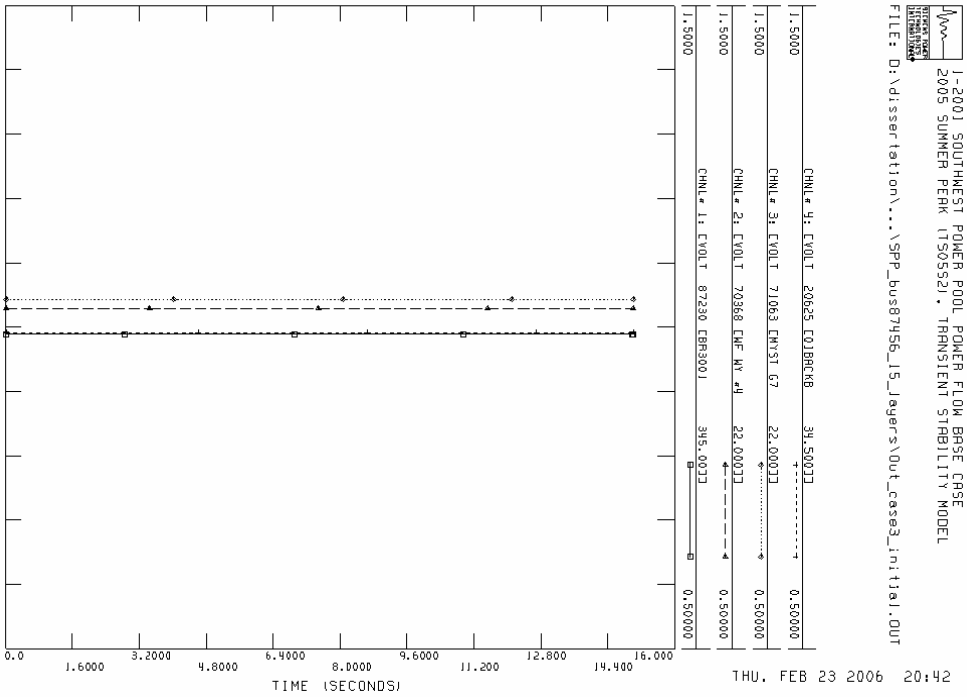
Channel 2: Voltage 70368 (22 kV)



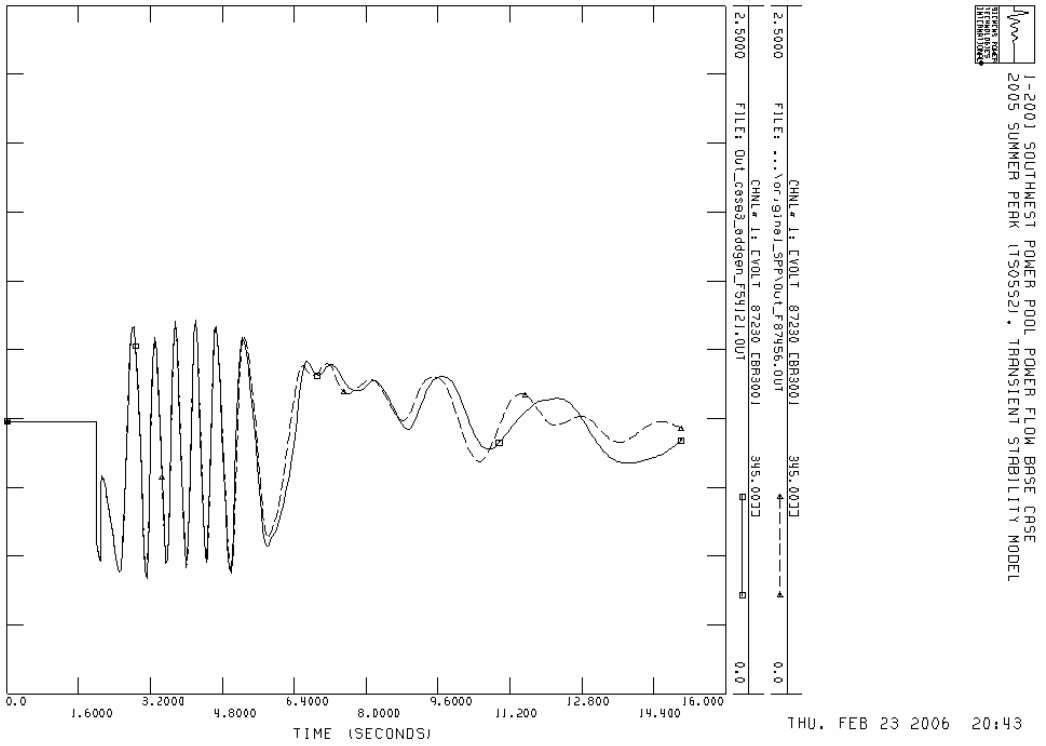

Channel 3: Voltage 71063 (22 kV)



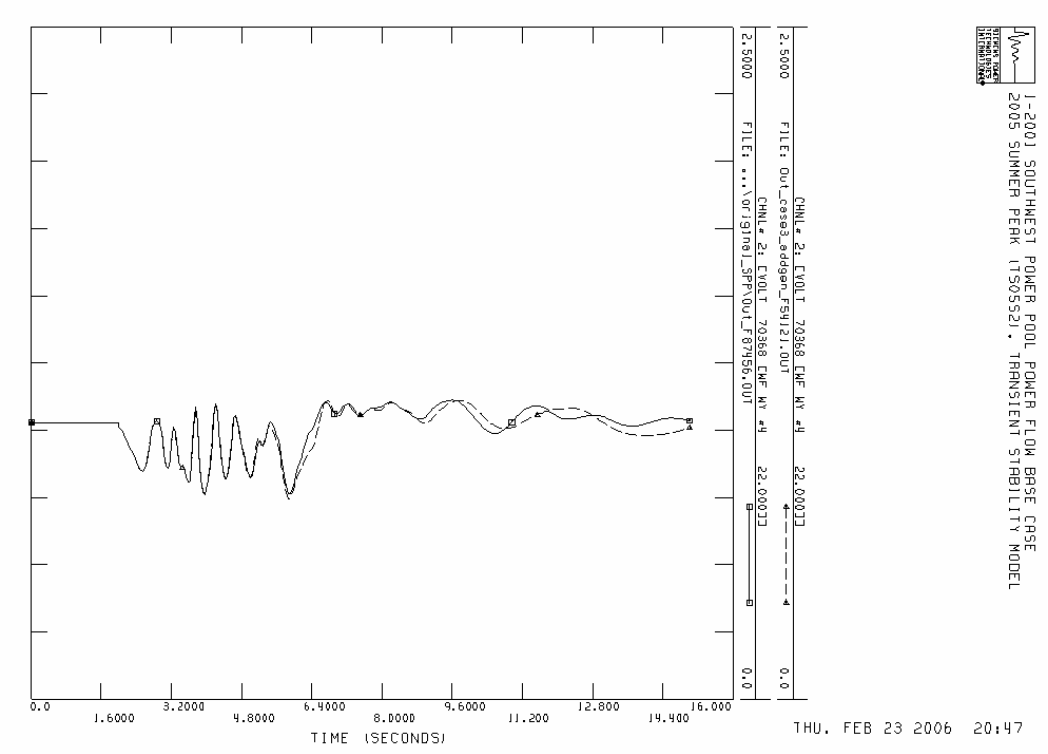
SPP_bus87456_15_layers
Initial condition for channel 1,2,3,4



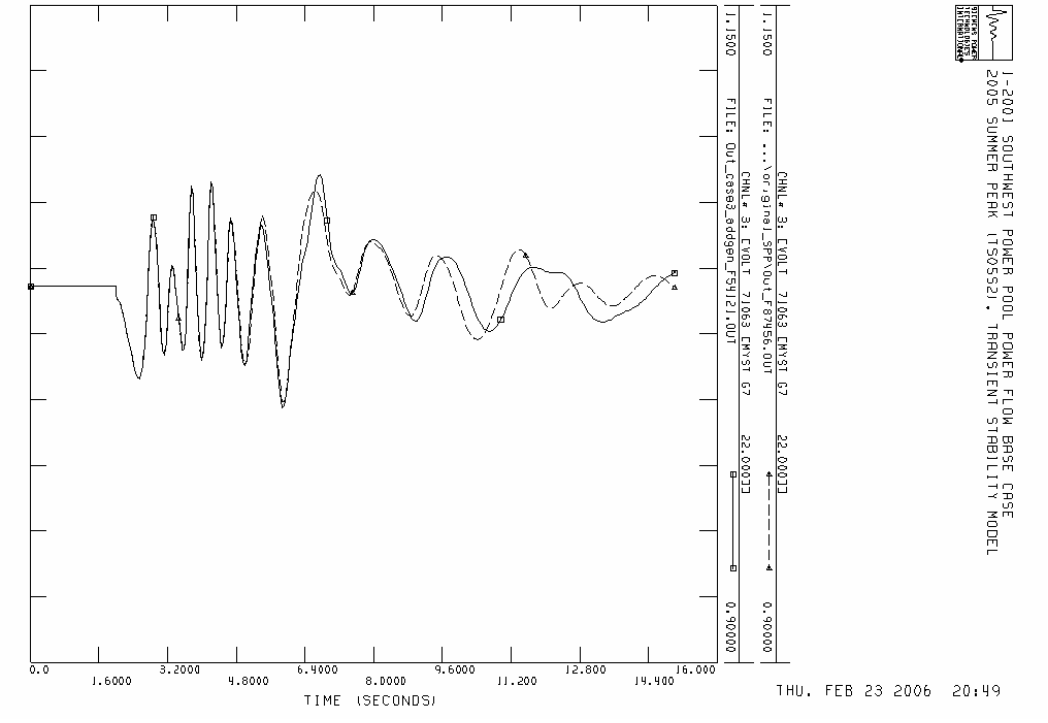

Channel 1: Voltage 87230 (345 kV)



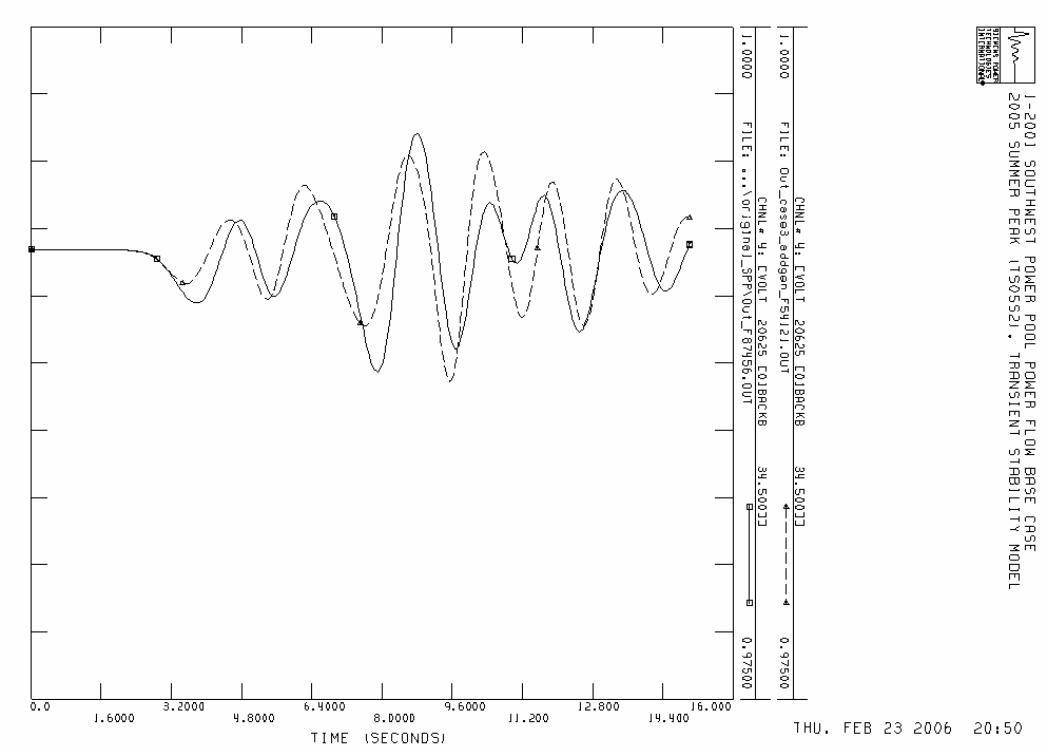
Channel 2: Voltage 70368 (22 kV)



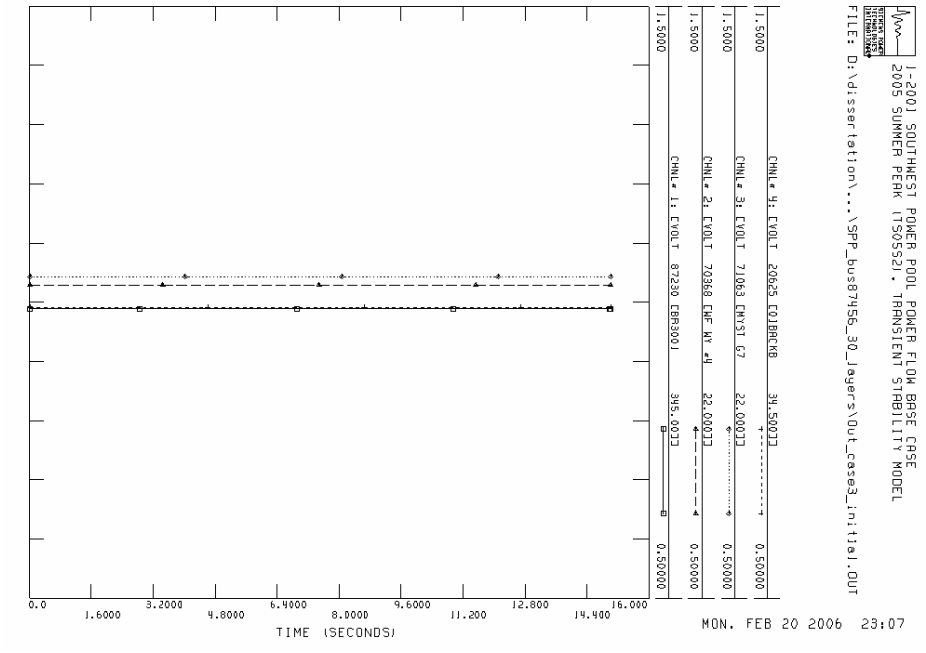

Channel 3: Voltage 71063 (22 kV)



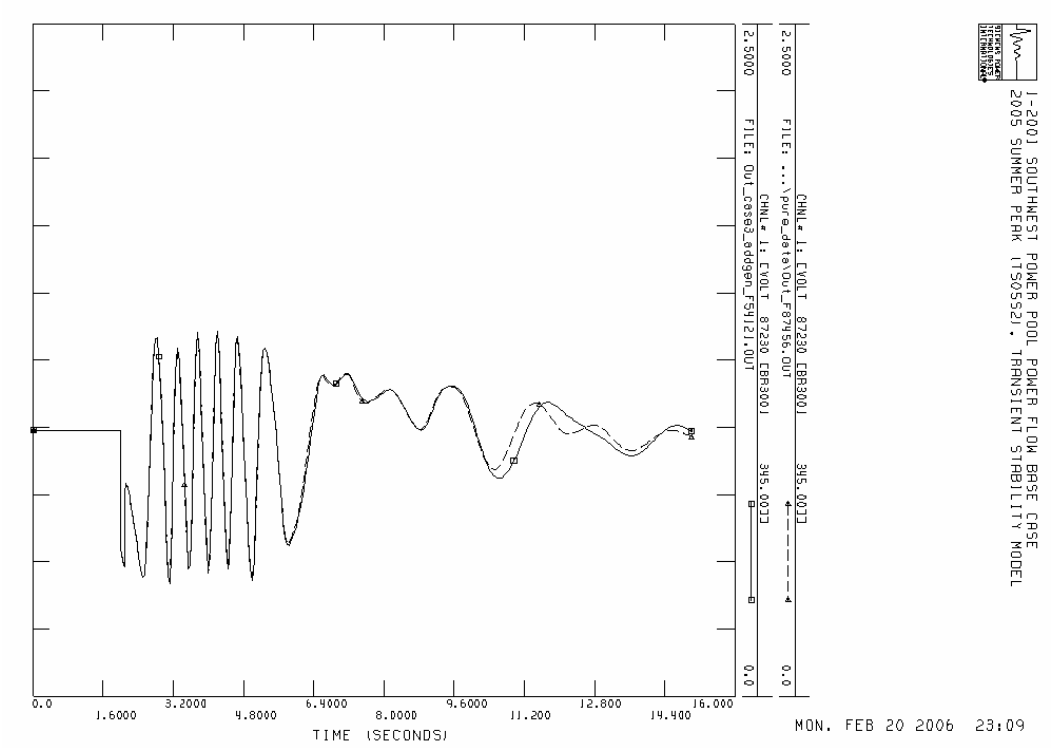
Channel 4: Voltage 20625 (34.5 kV)



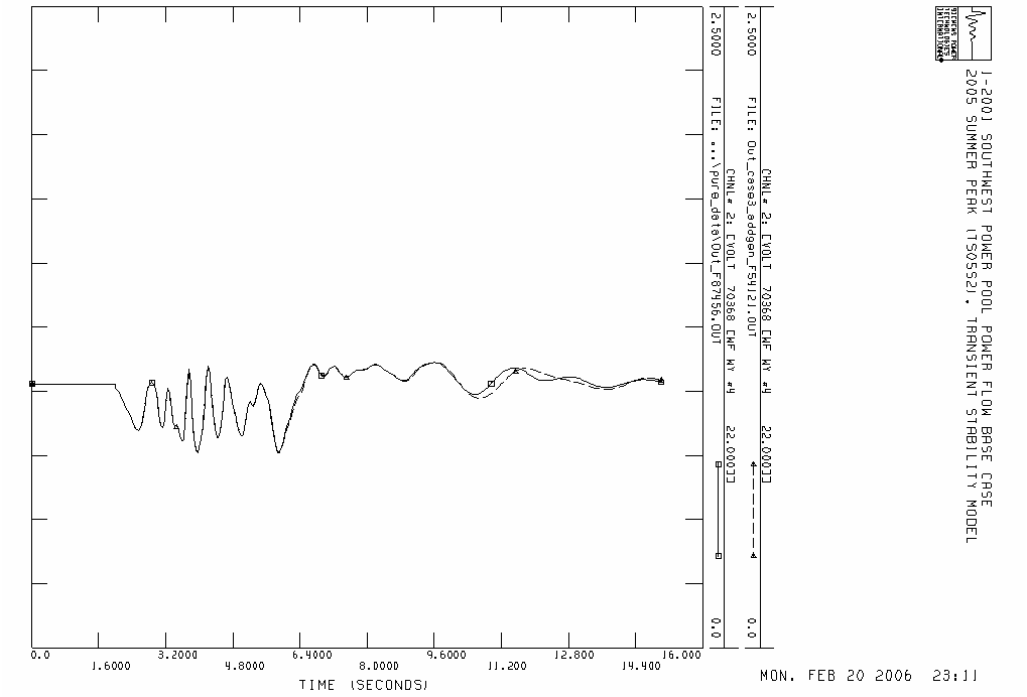

SPP_bus87456_30_layers
Initial condition for channel 1,2,3,4



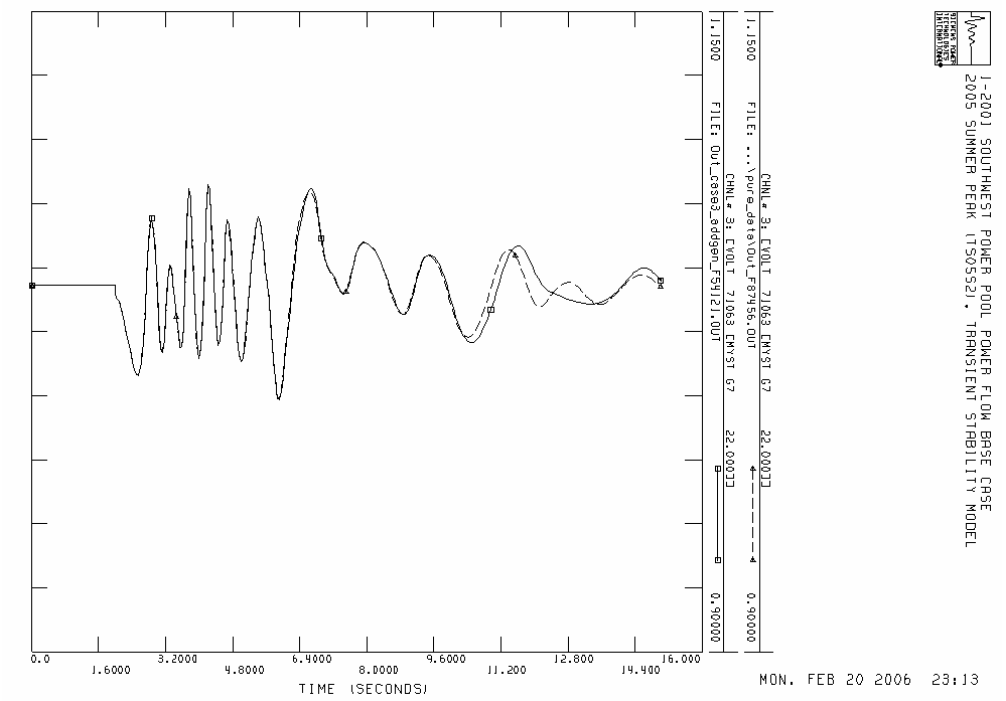
Channel 1: Voltage 87230 (345 kV)



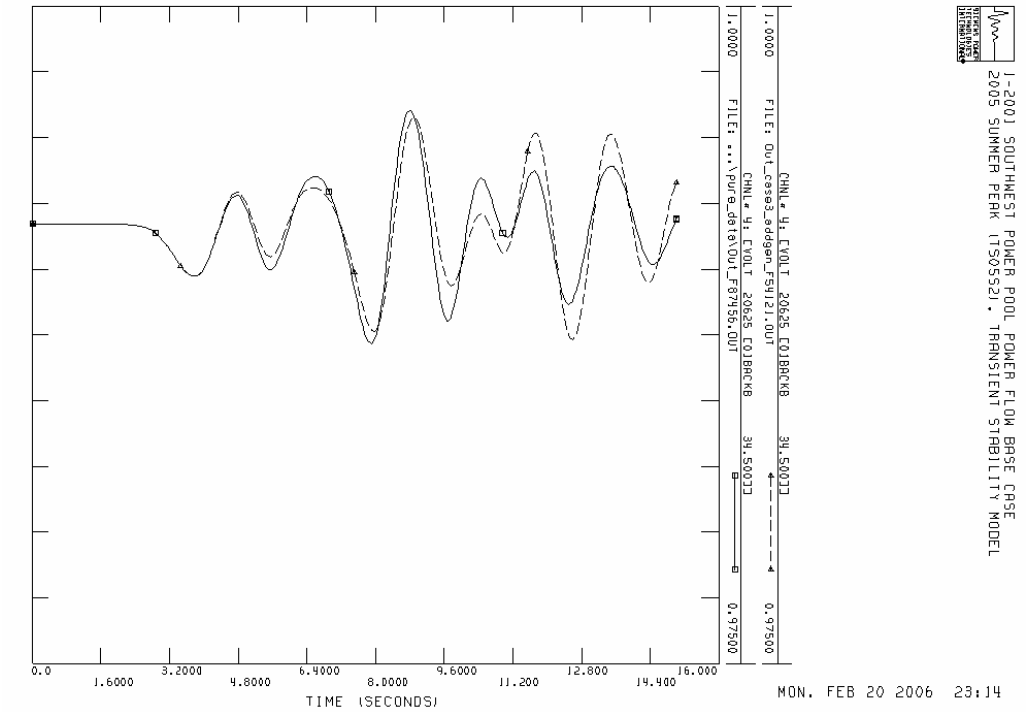

Channel 2: Voltage 70368 (22 kV)



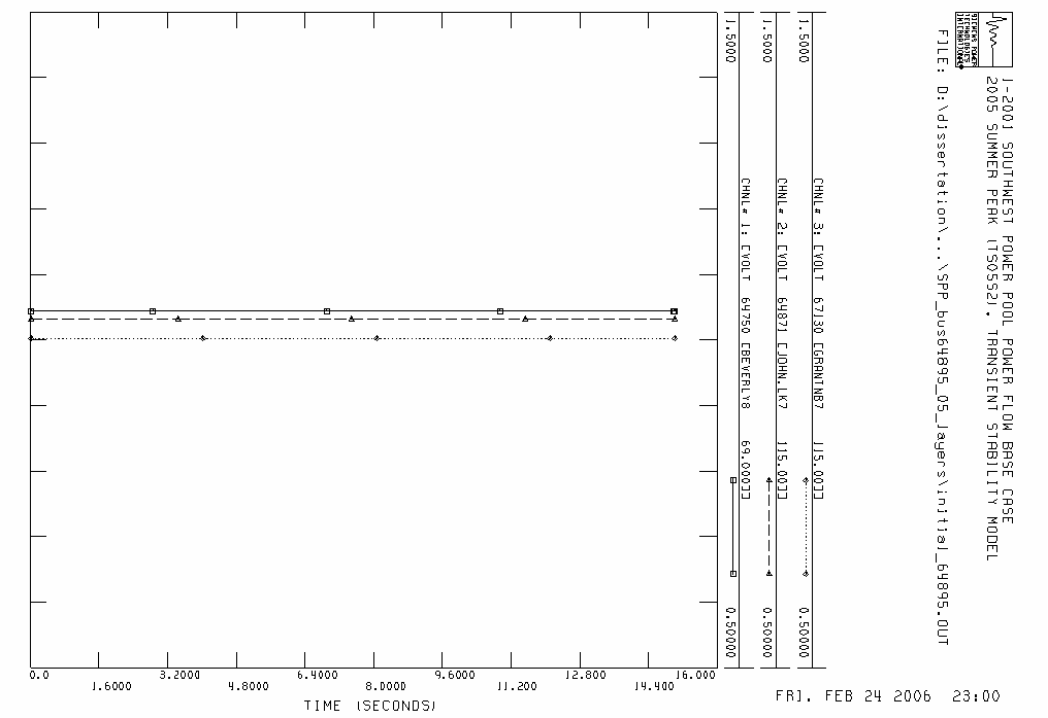
Channel 3: Voltage 71063 (22 kV)



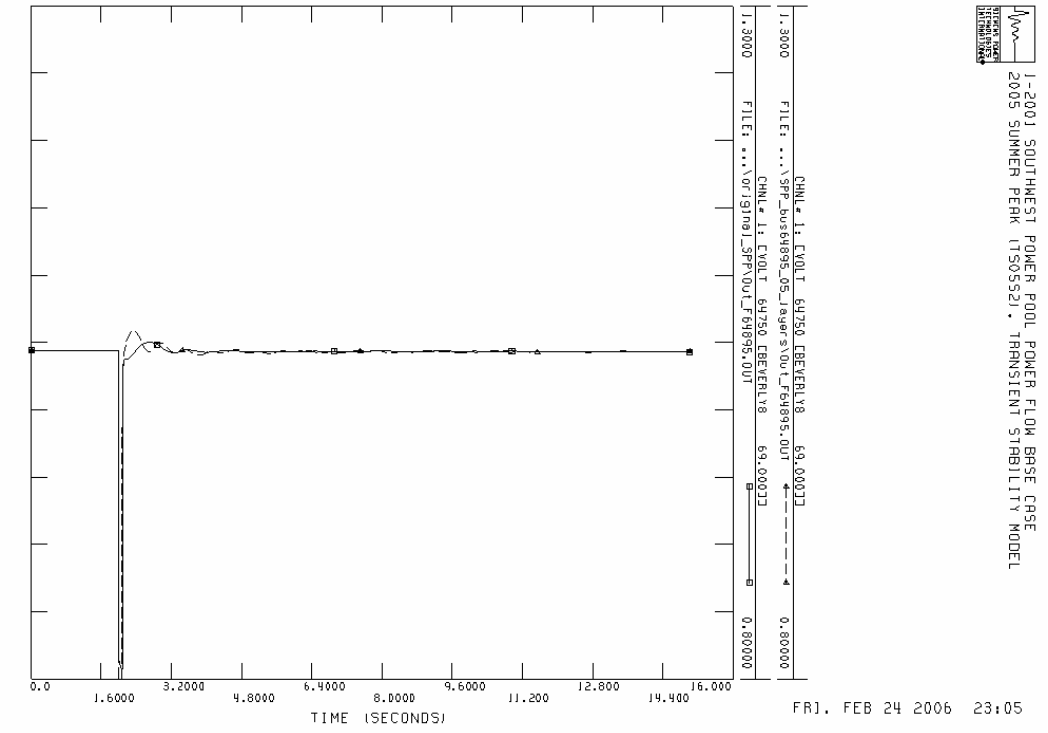

Channel 4: Voltage 20625 (34.5 kV)



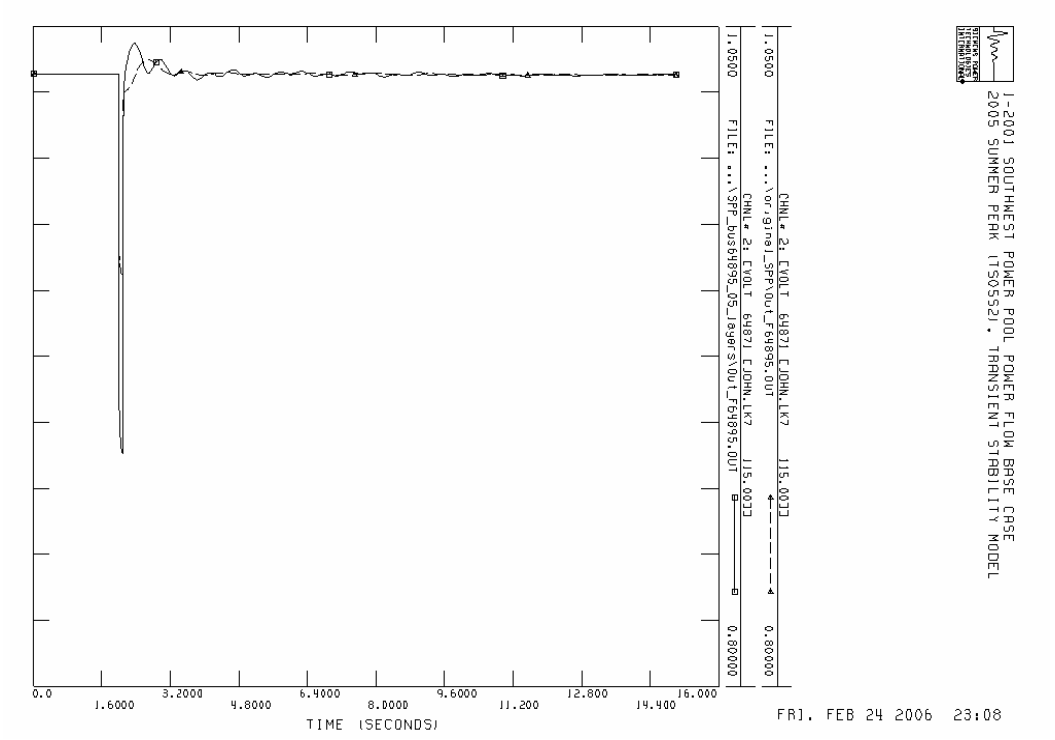
SPP_bus64895_5_layers
Initial condition for channel 1,2,3



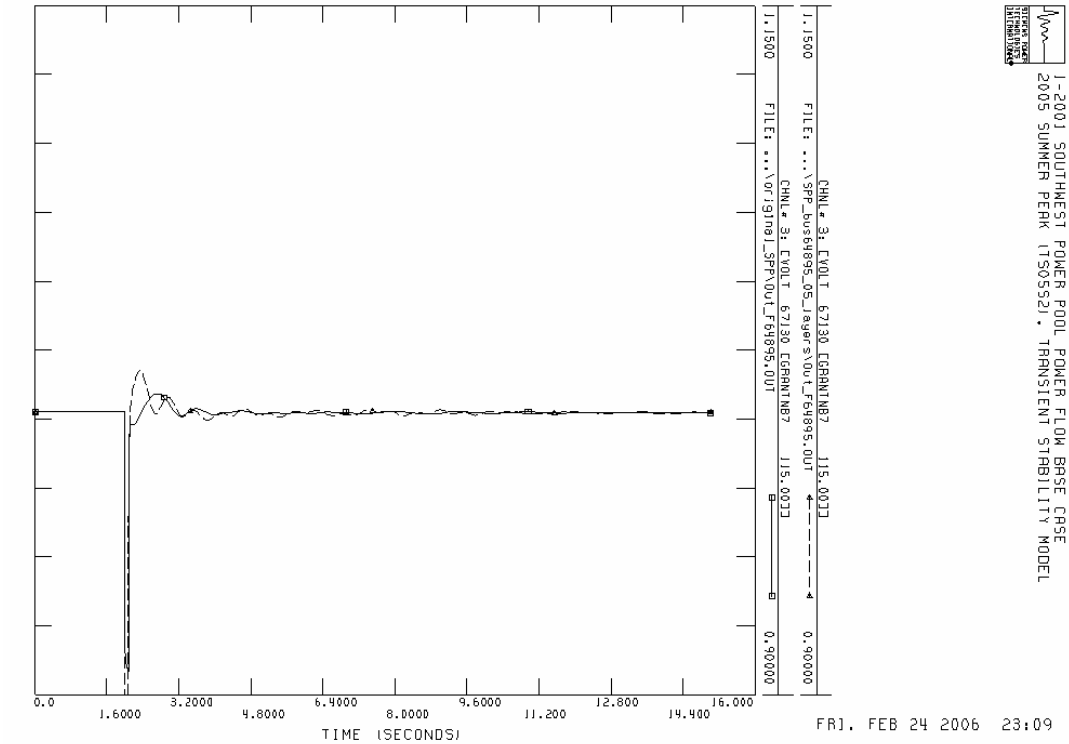

Channel 1: Power bus 64750 (69 kV)



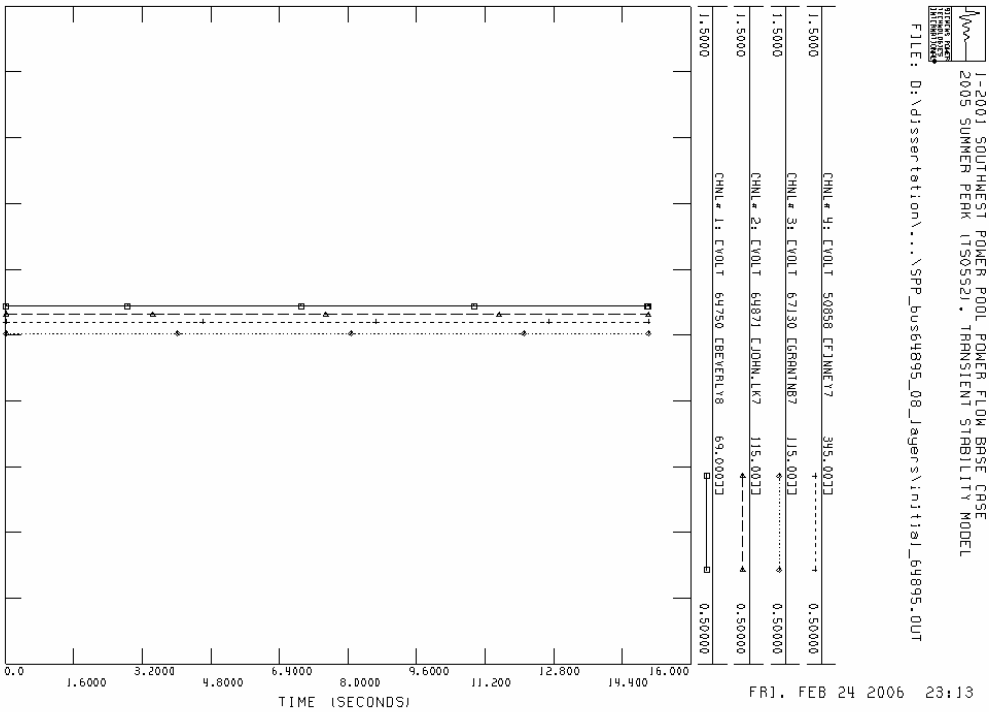
Channel 2: Power bus 64871 (115 kV)



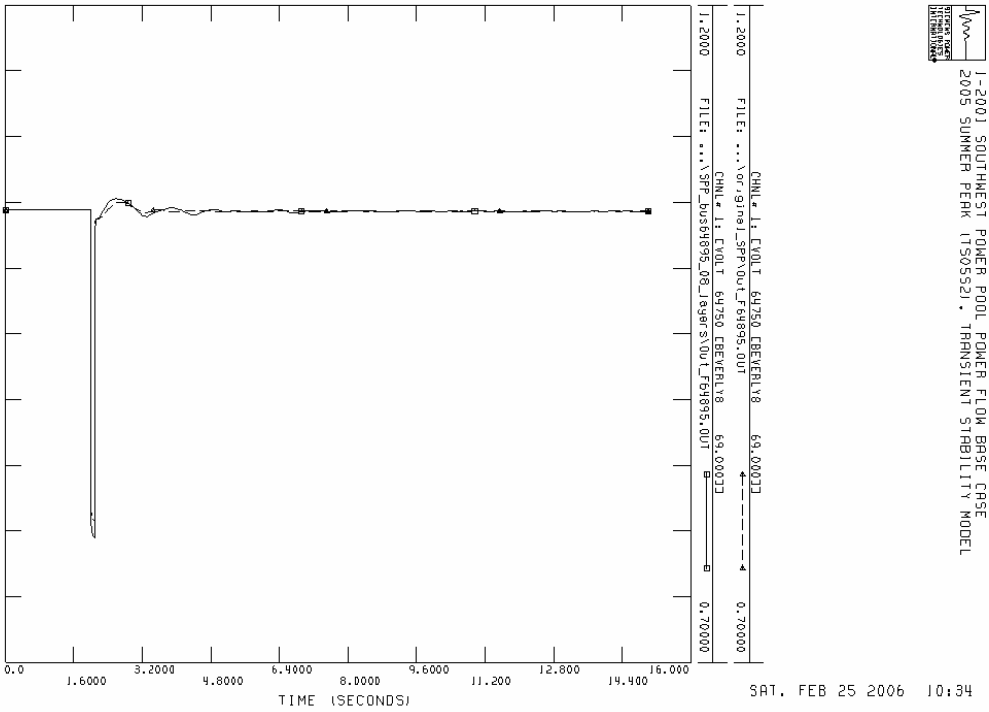

Channel 3: Power bus 67130 (115 kV)



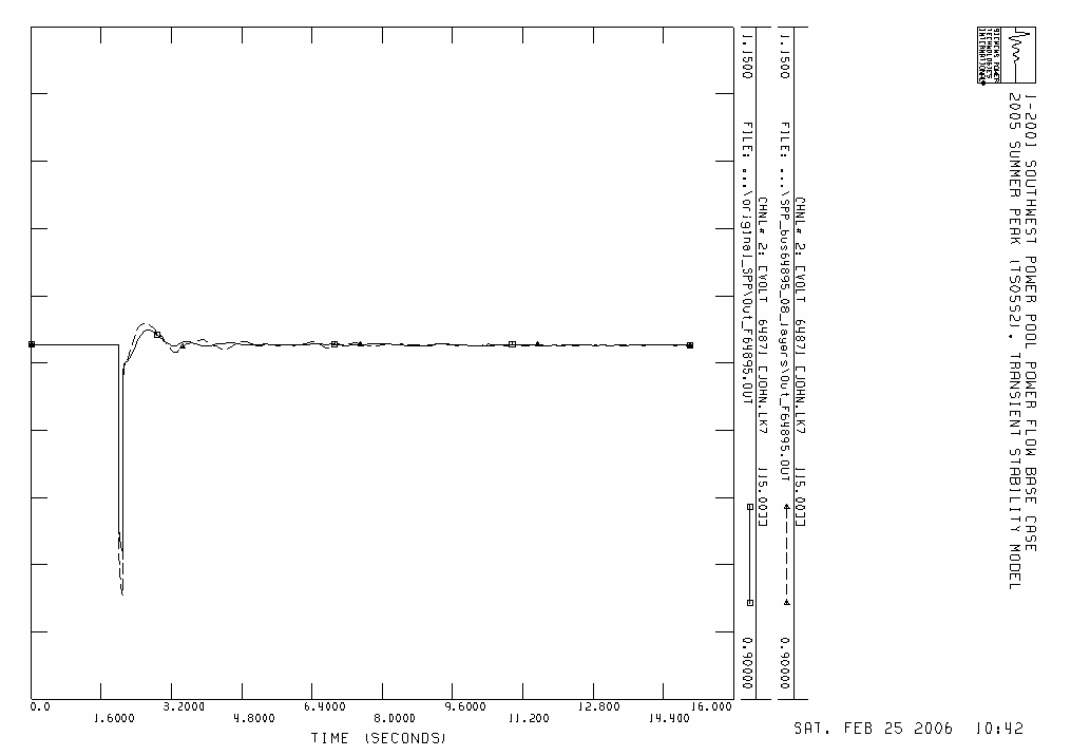
SPP_bus64895_8_layers
Initial condition for channel 1,2,3,4



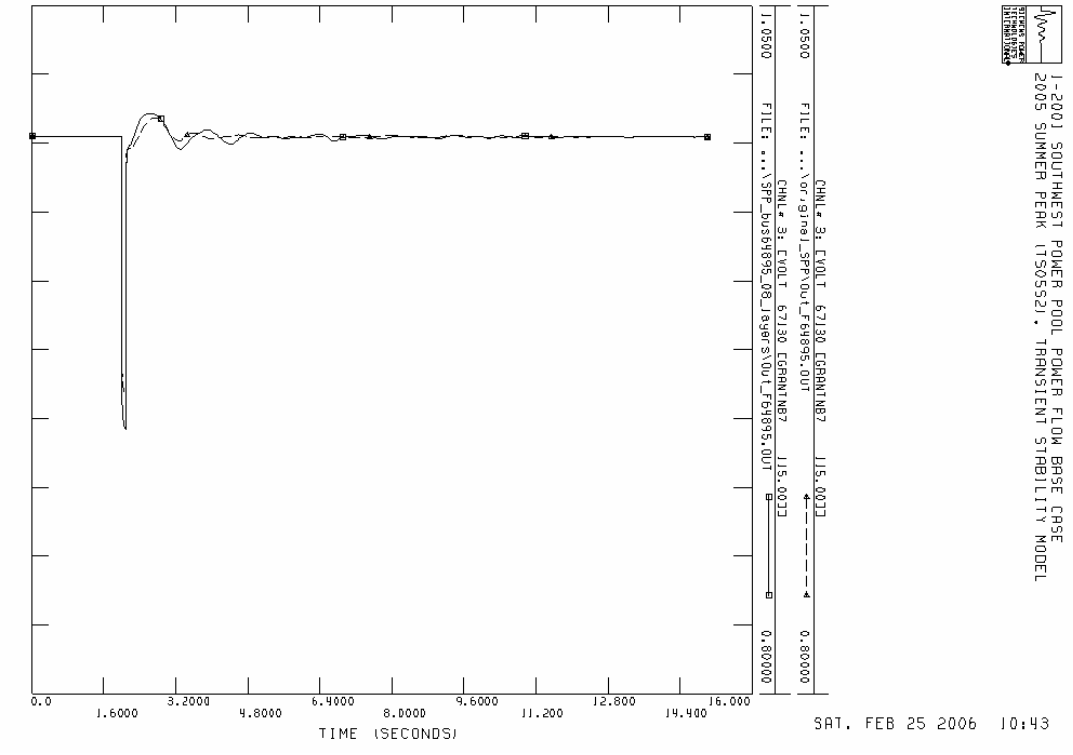

Channel 1: Power bus 64750 (69 kV)



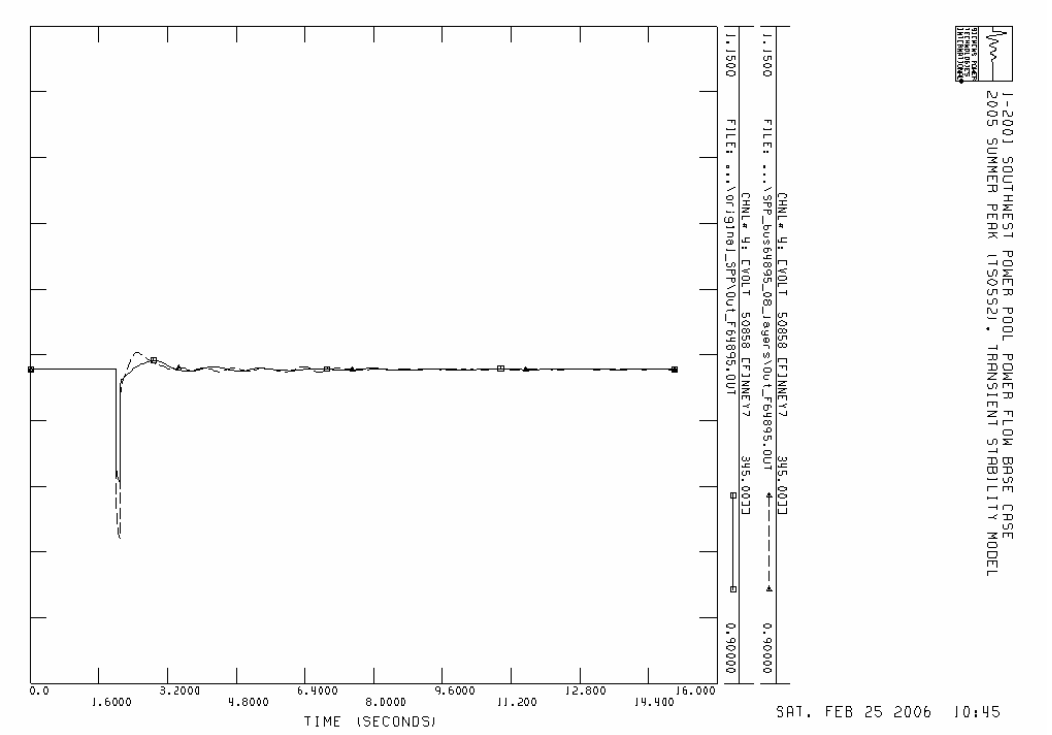
Channel 2: Power bus 64871 (115 kV)



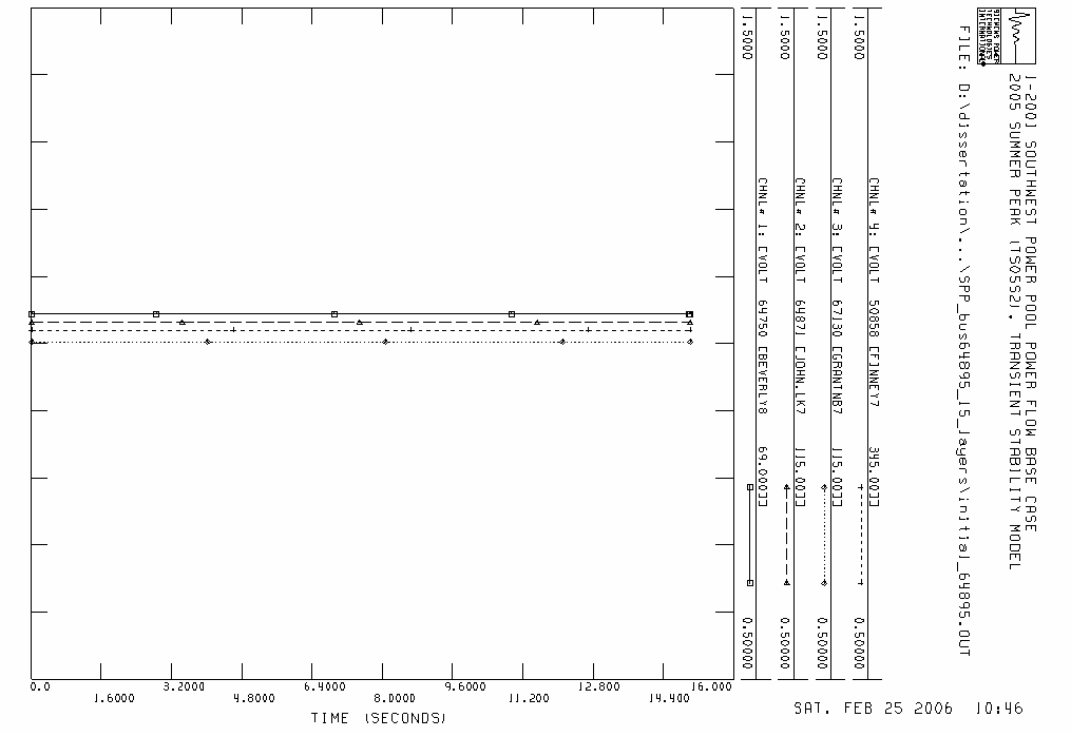

Channel 3: Power bus 67130 (115 kV)



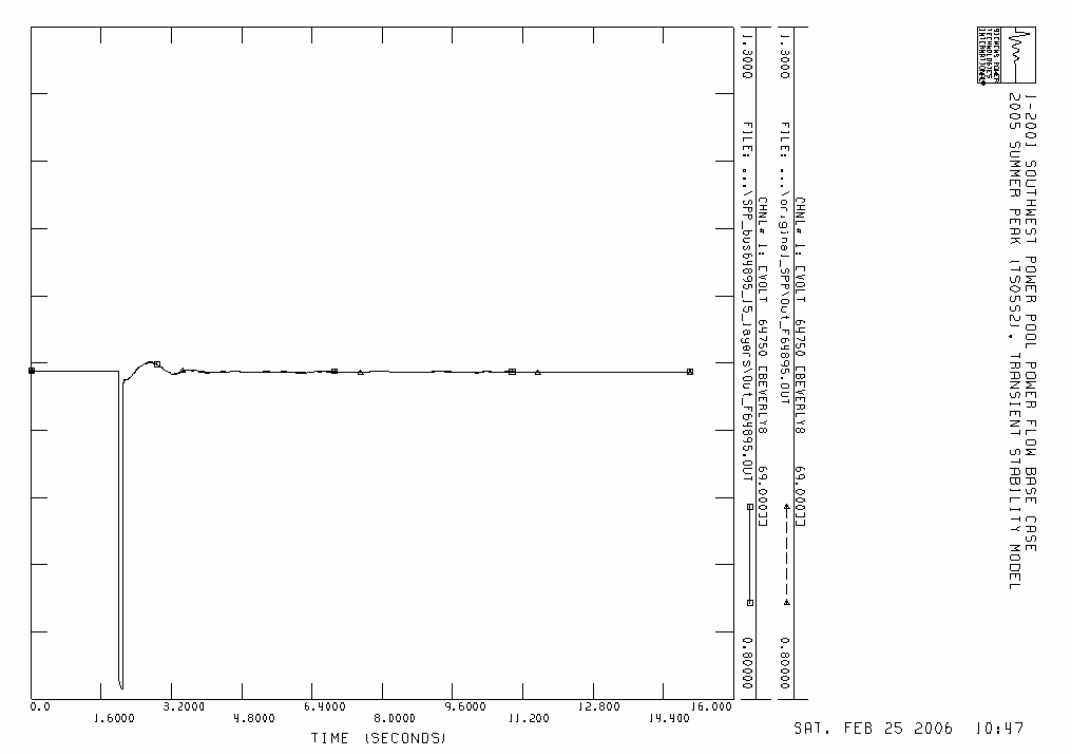
Channel 4: Power bus 50858 (345 kV)



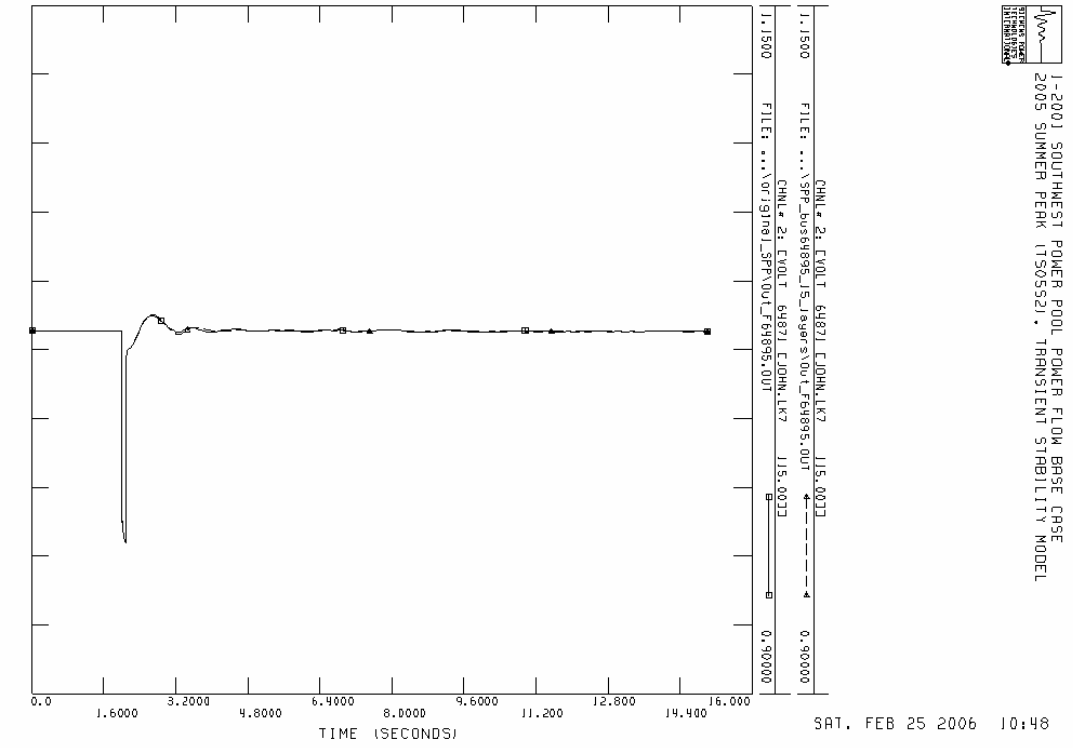

SPP_bus64895_15_layers
Initial condition for channel 1,2,3,4



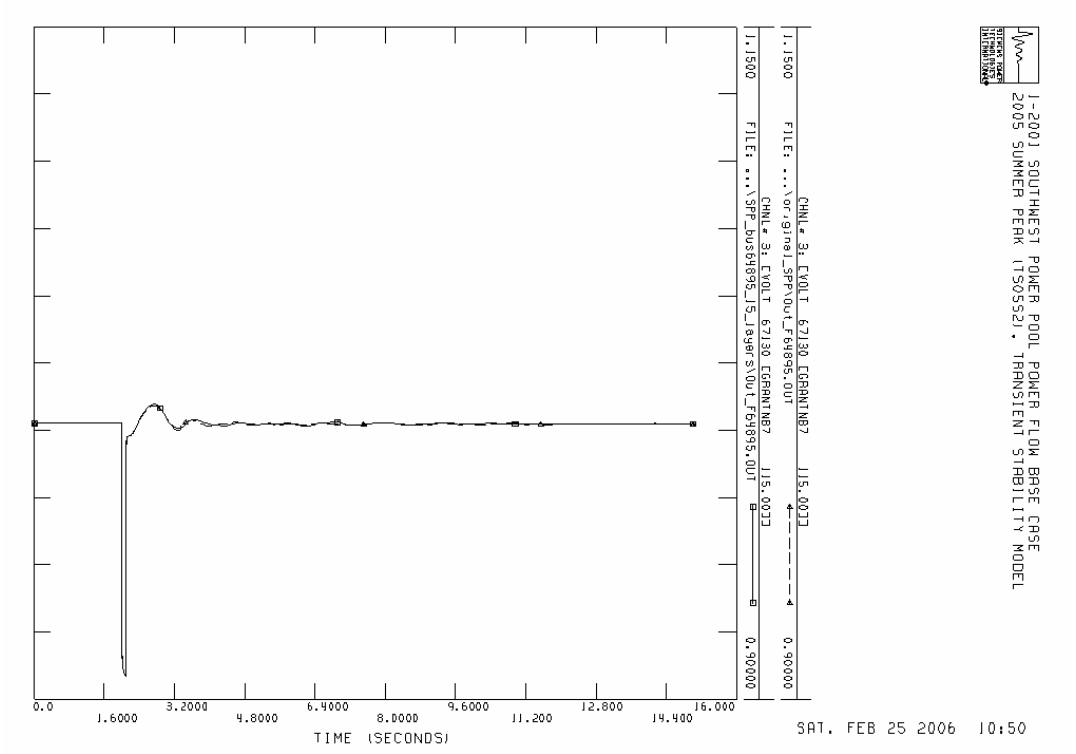
Channel 1: Power bus 64750 (69 kV)



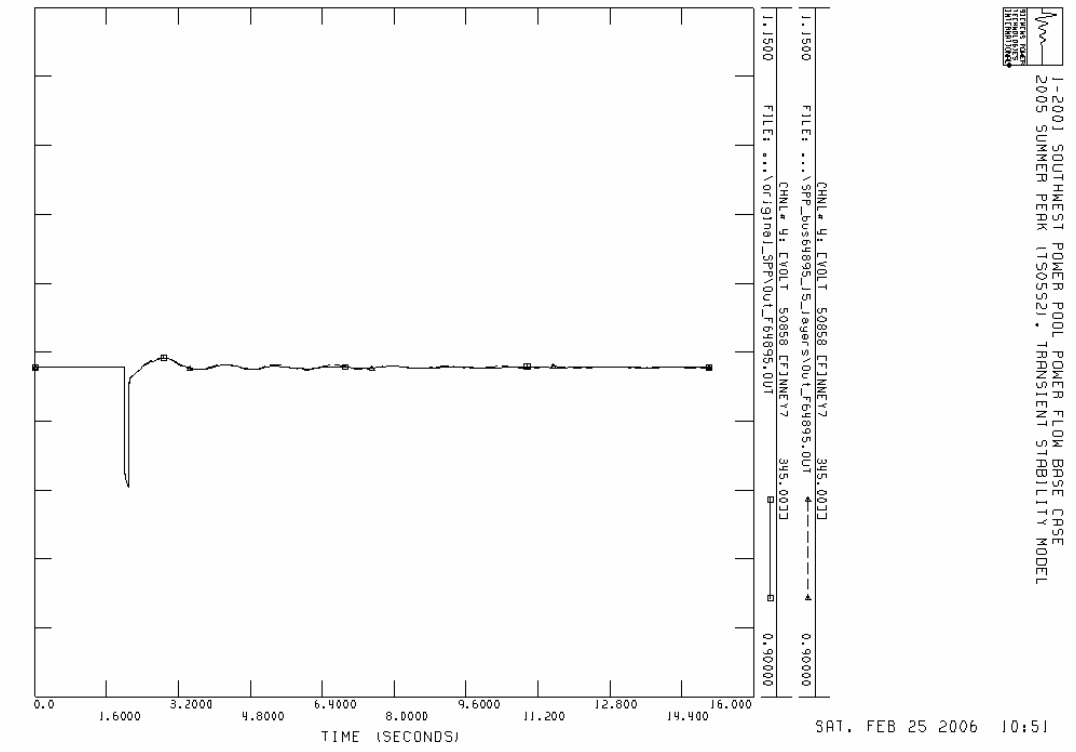

Channel 2: Power bus 64871 (115 kV)



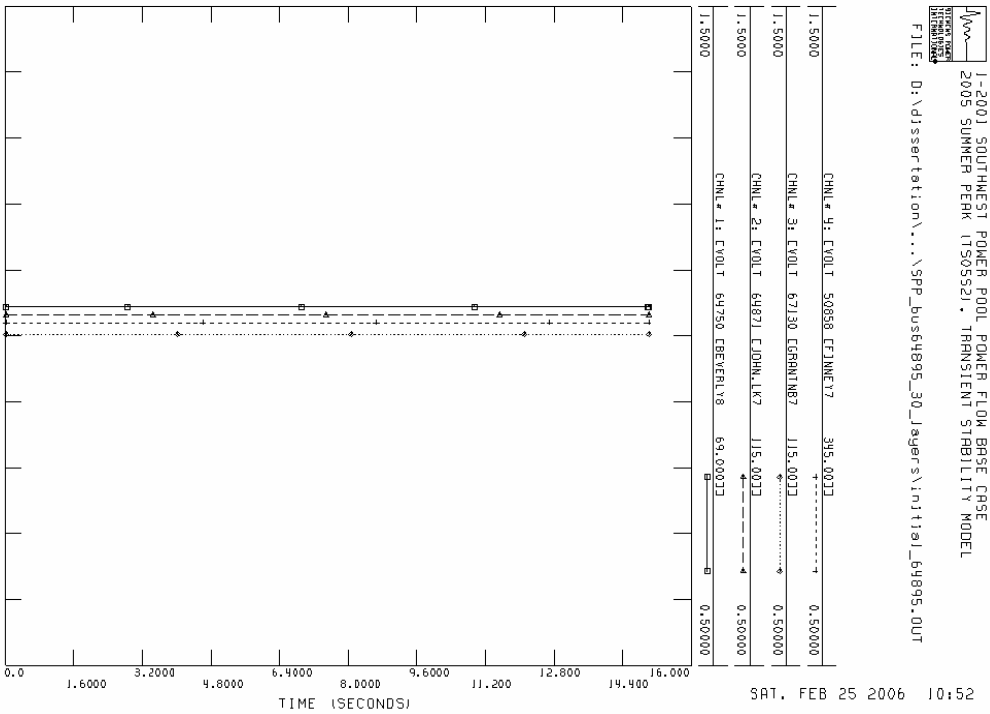
Channel 3: Power bus 67130 (115 kV)



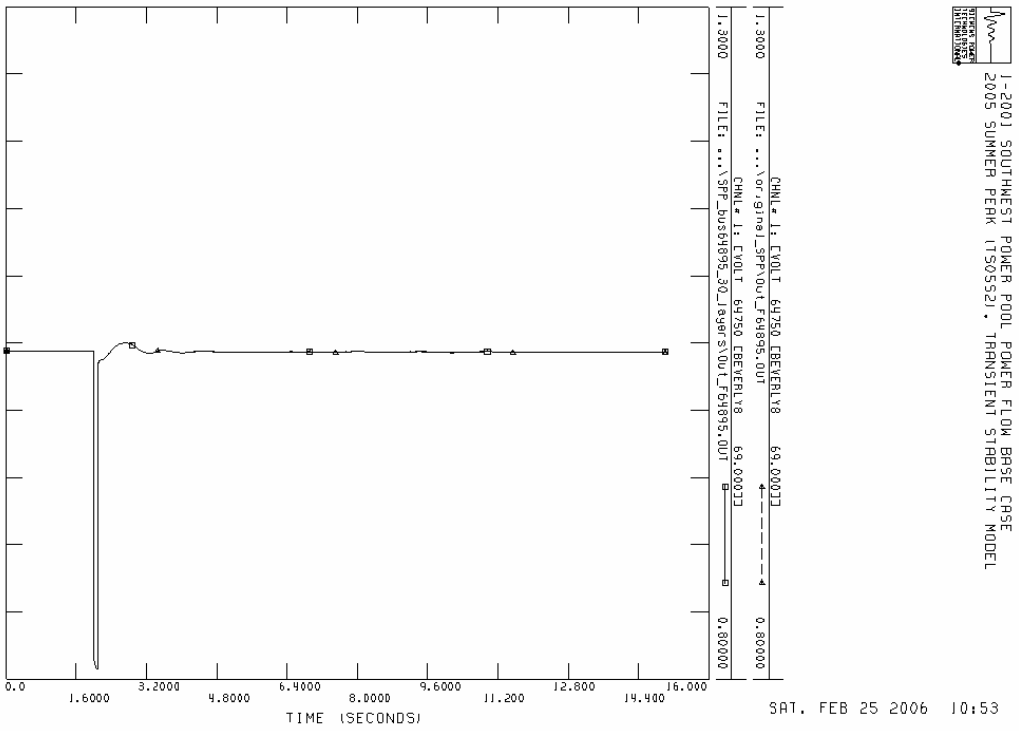

Channel 4: Power bus 50858 (345 kV)



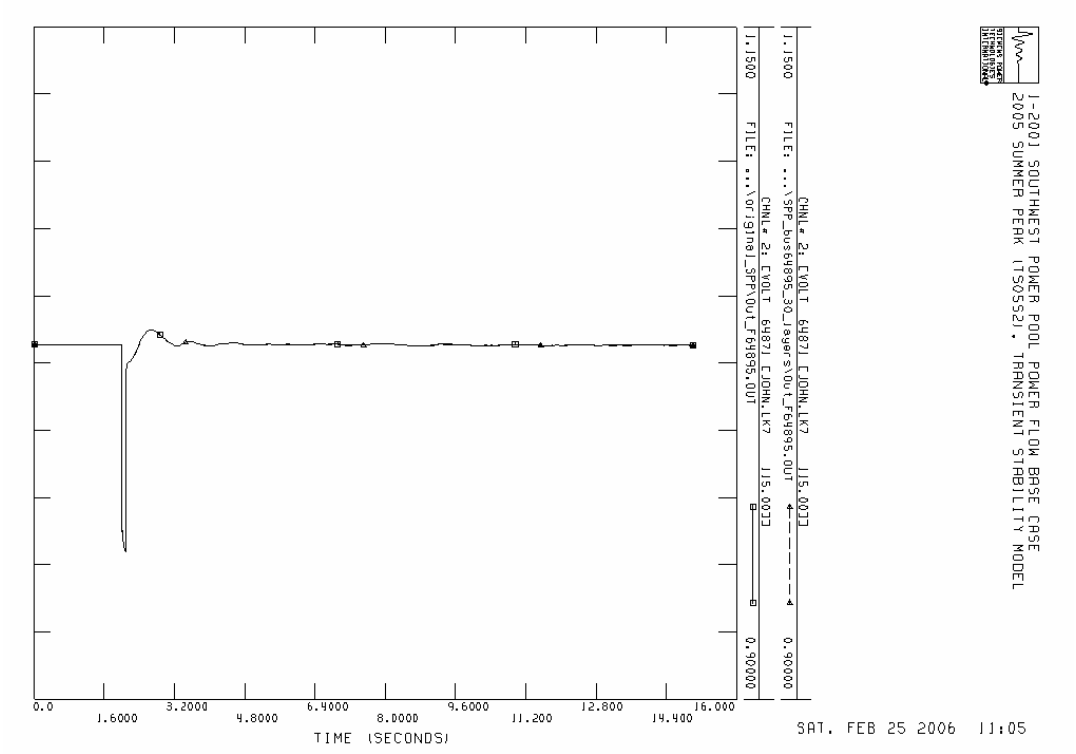
SPP_bus64895_30_layers
Initial condition for channel 1,2,3,4



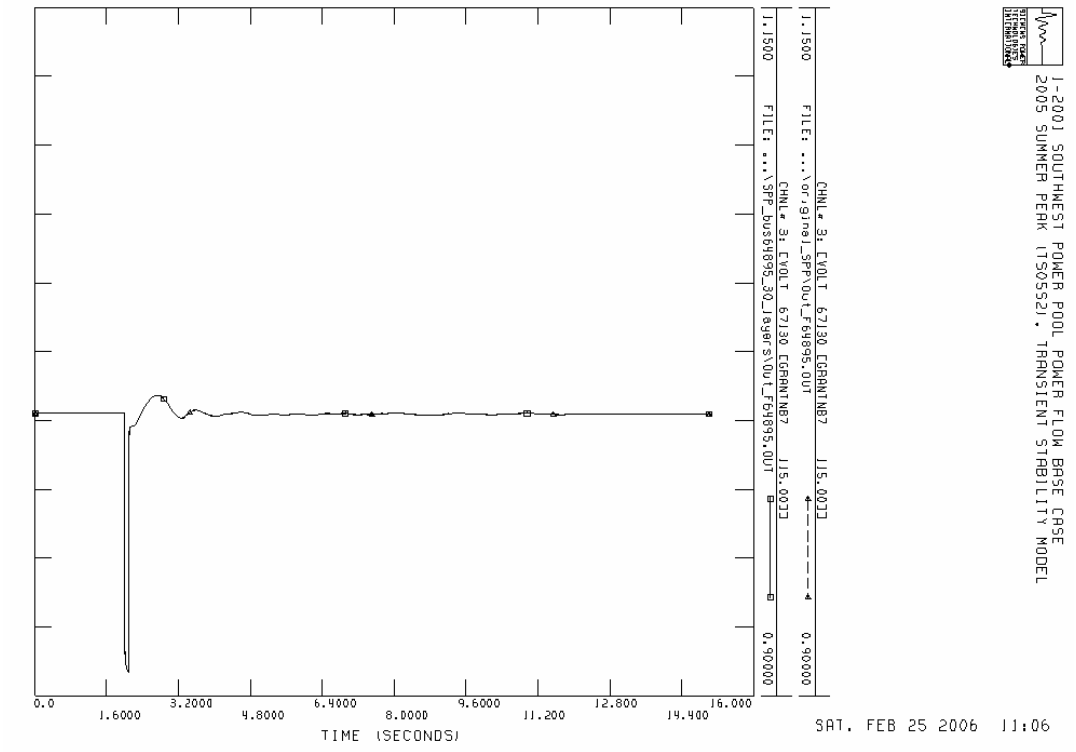

Channel 1: Power bus 64750 (69 kV)



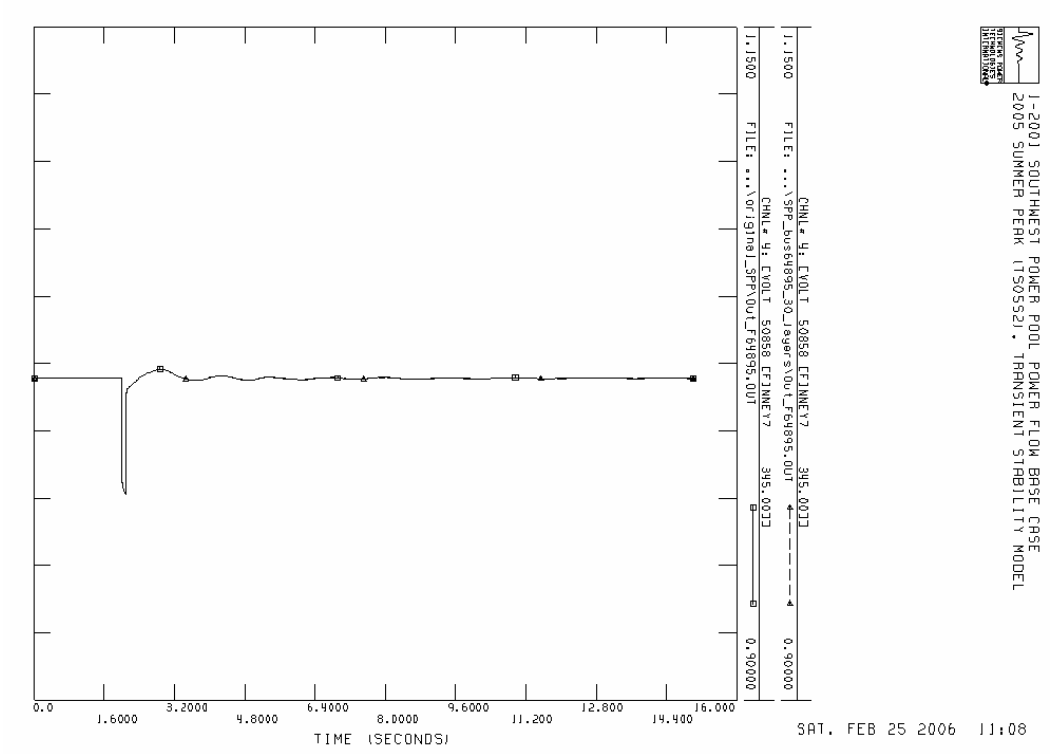
Channel 2: Power bus 64871 (115 kV)



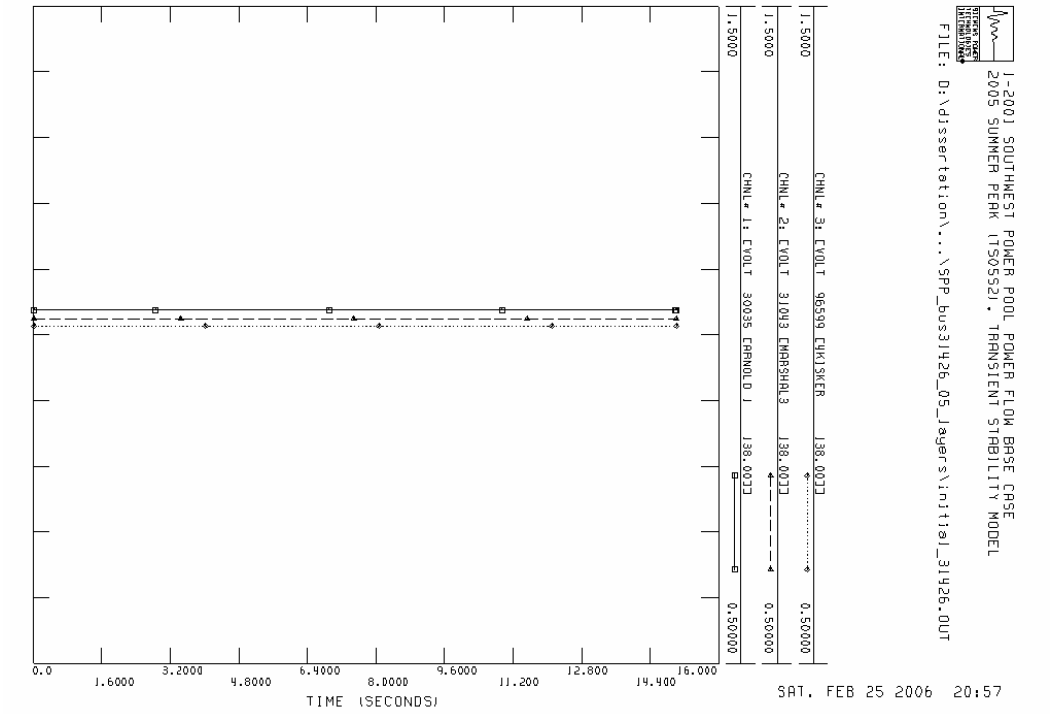

Channel 3: Power bus 67130 (115 kV)



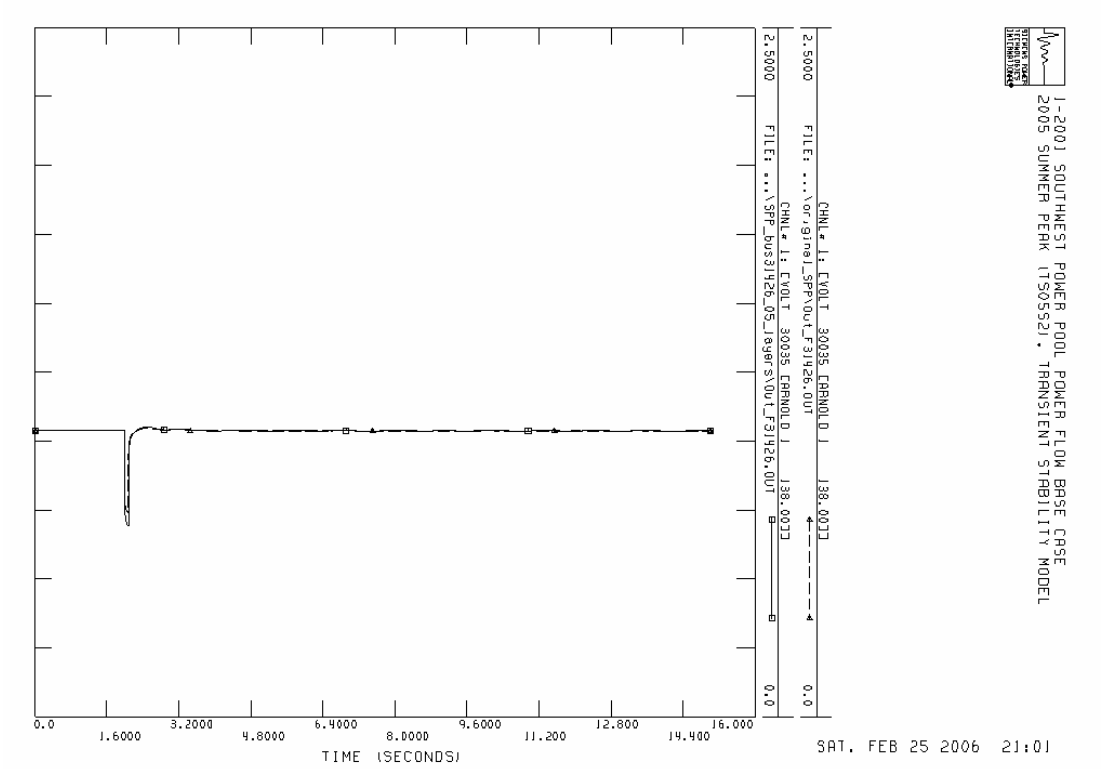
Channel 4: Power bus 50858 (345 kV)



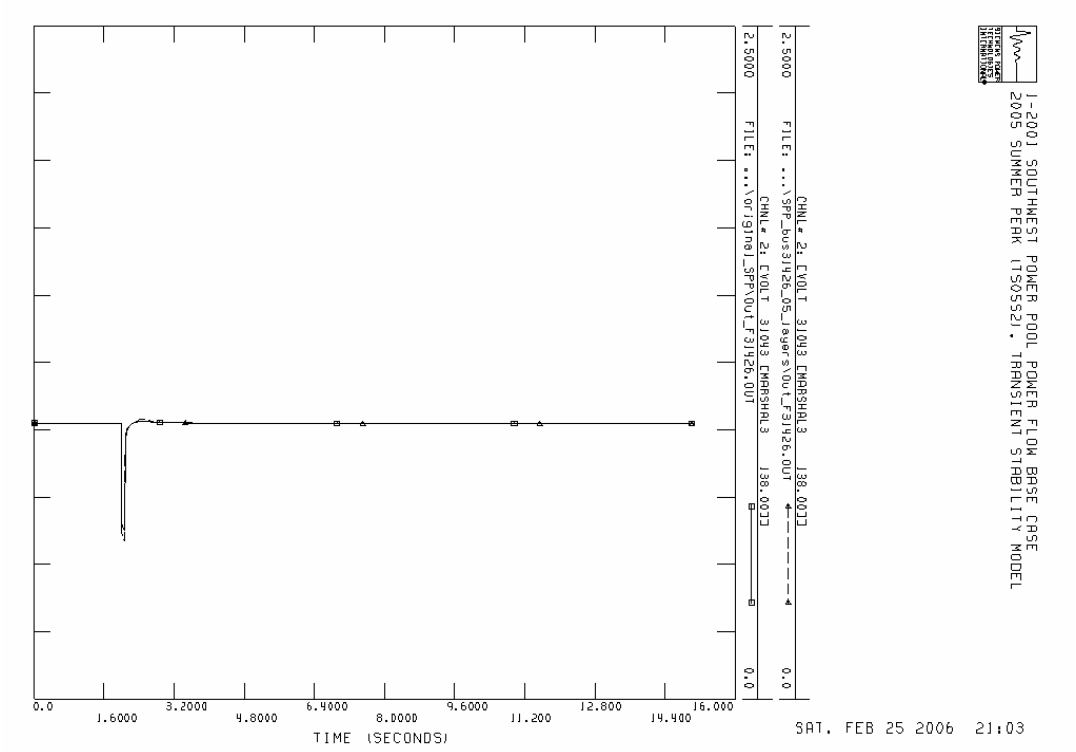

SPP_bus31426_5_layers
Initial condition for channel 1,2,3



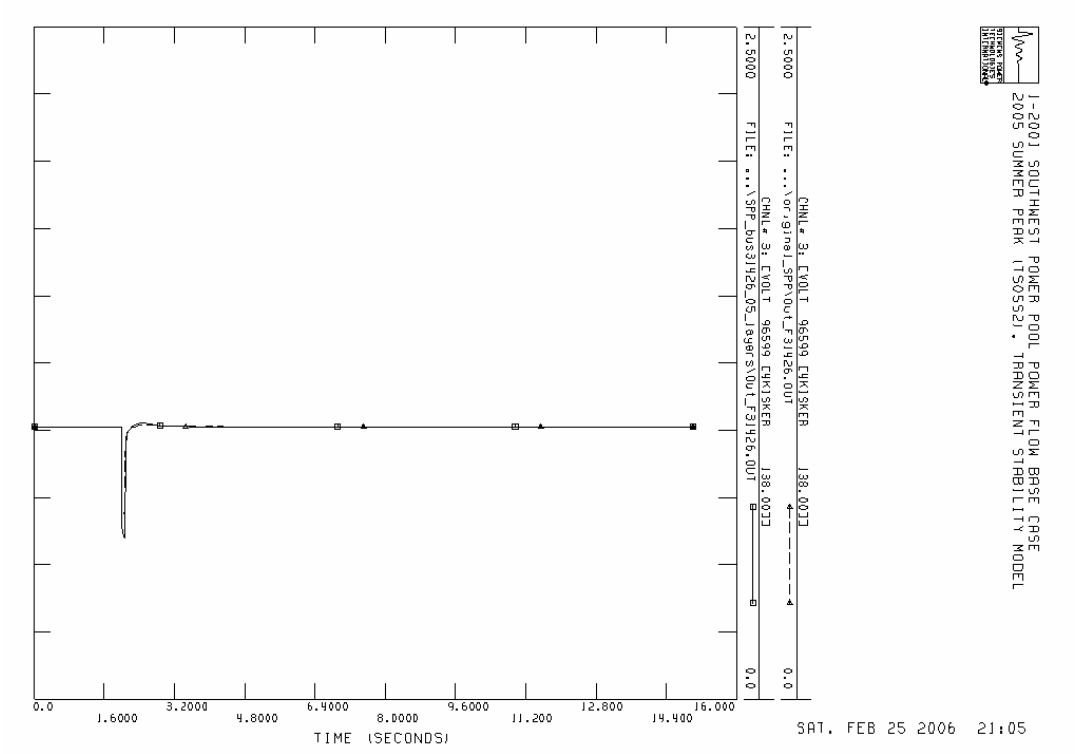
Channel 1: Power bus 30035 (138 kV)




Channel 2: Power bus 31043 (138 kV)

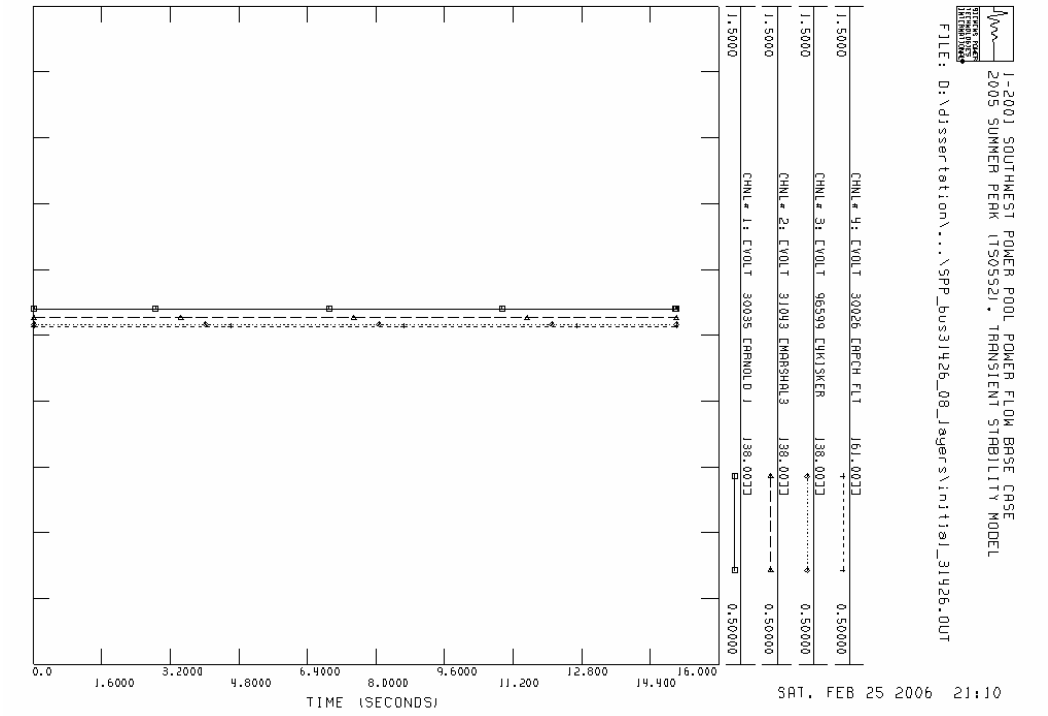


Channel 3: Power bus 96599 (138 kV)

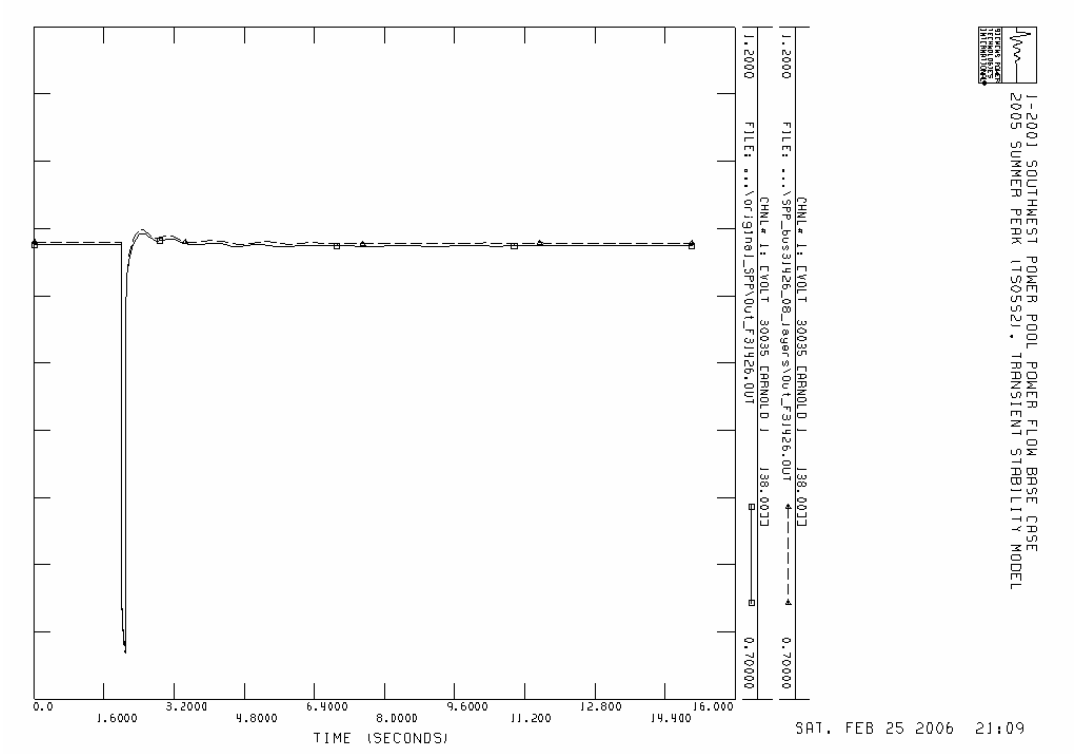



SPP_bus31426_8_layers

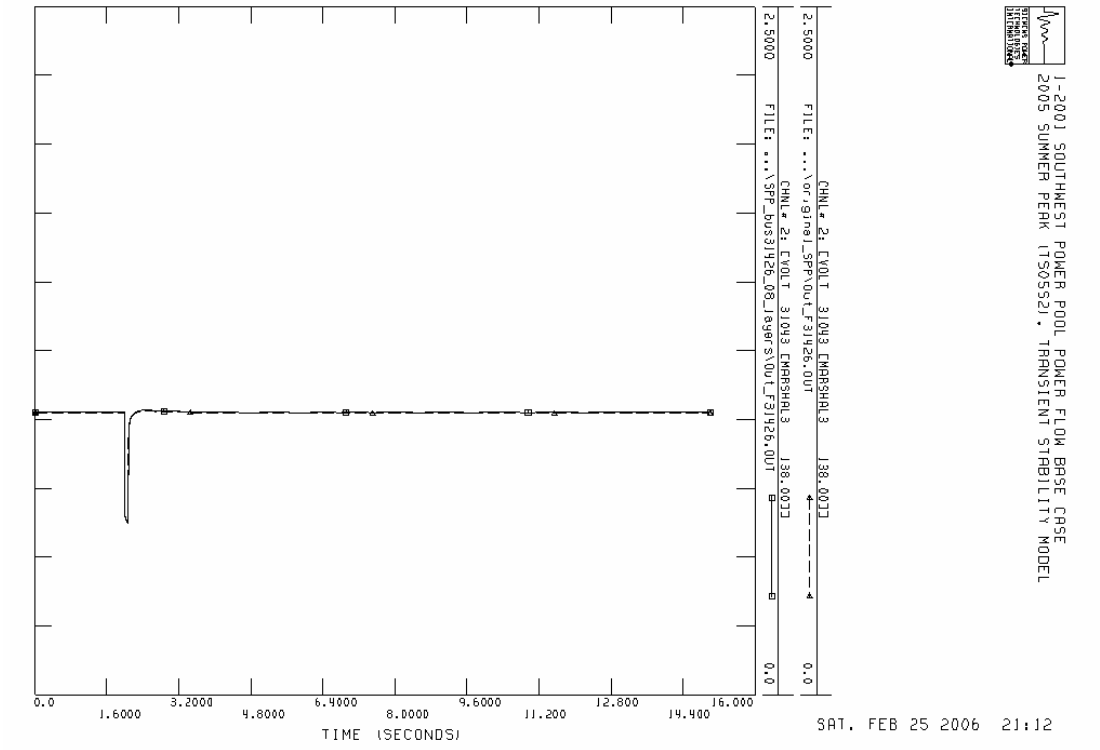
Initial condition for channel 1,2,3,4



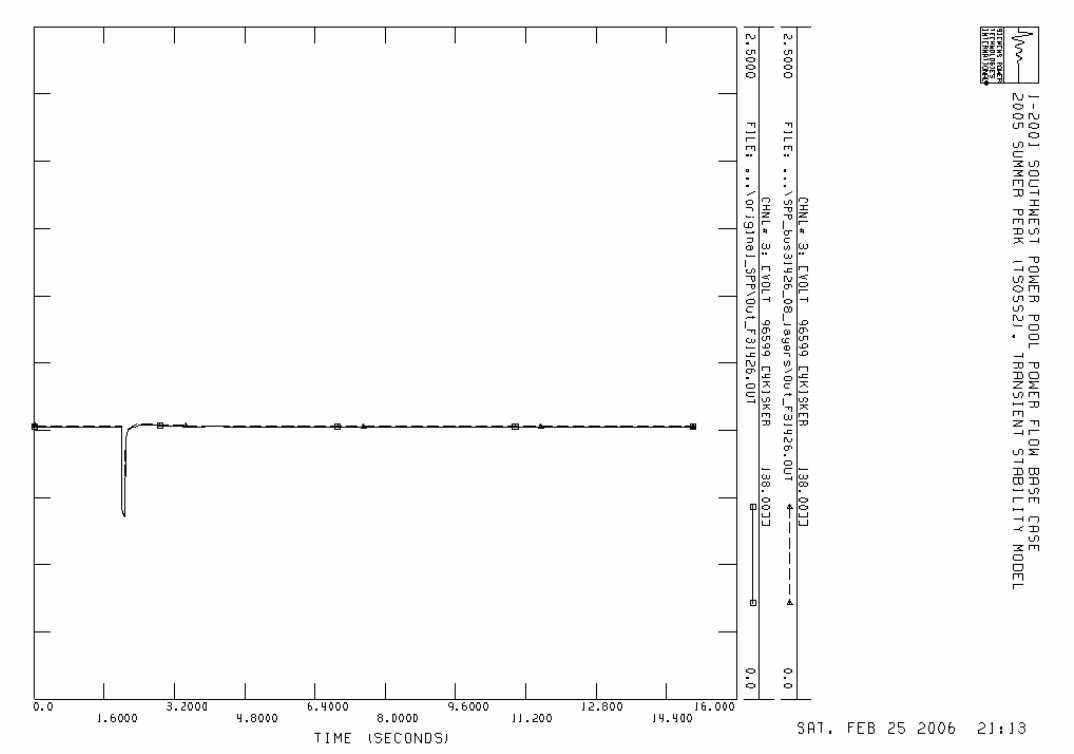
Channel 1: Power bus 30035 (138 kV)



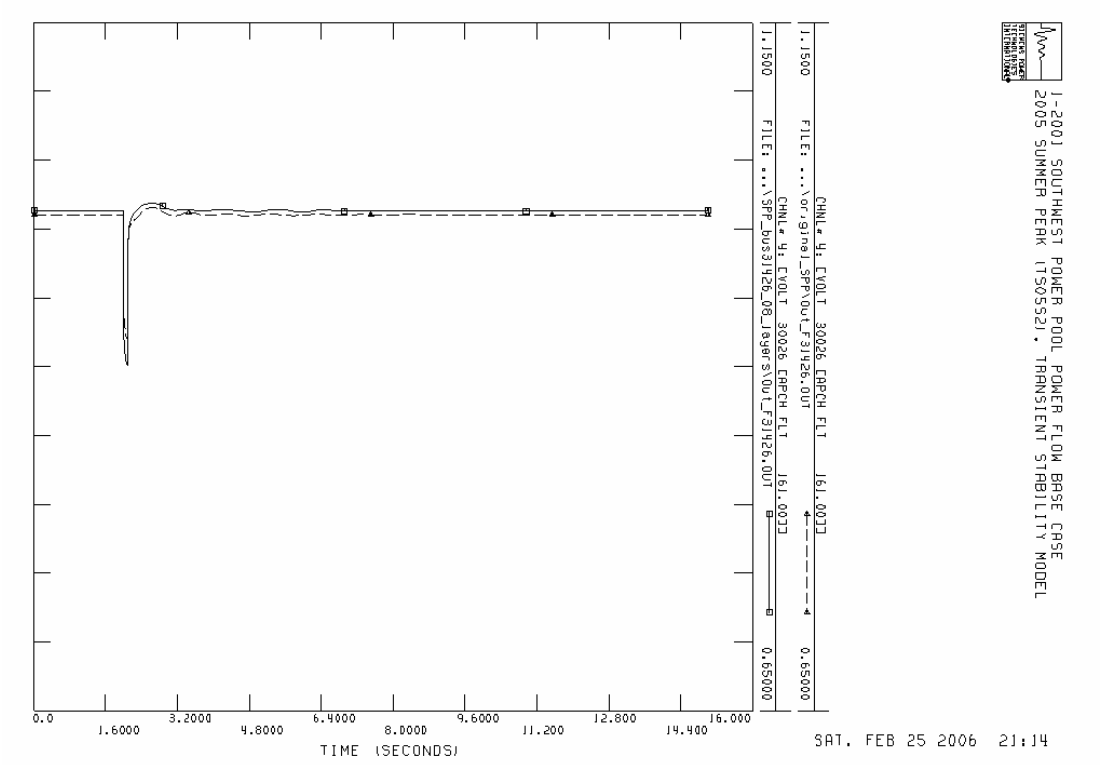

Channel 2: Power bus 31043 (138 kV)



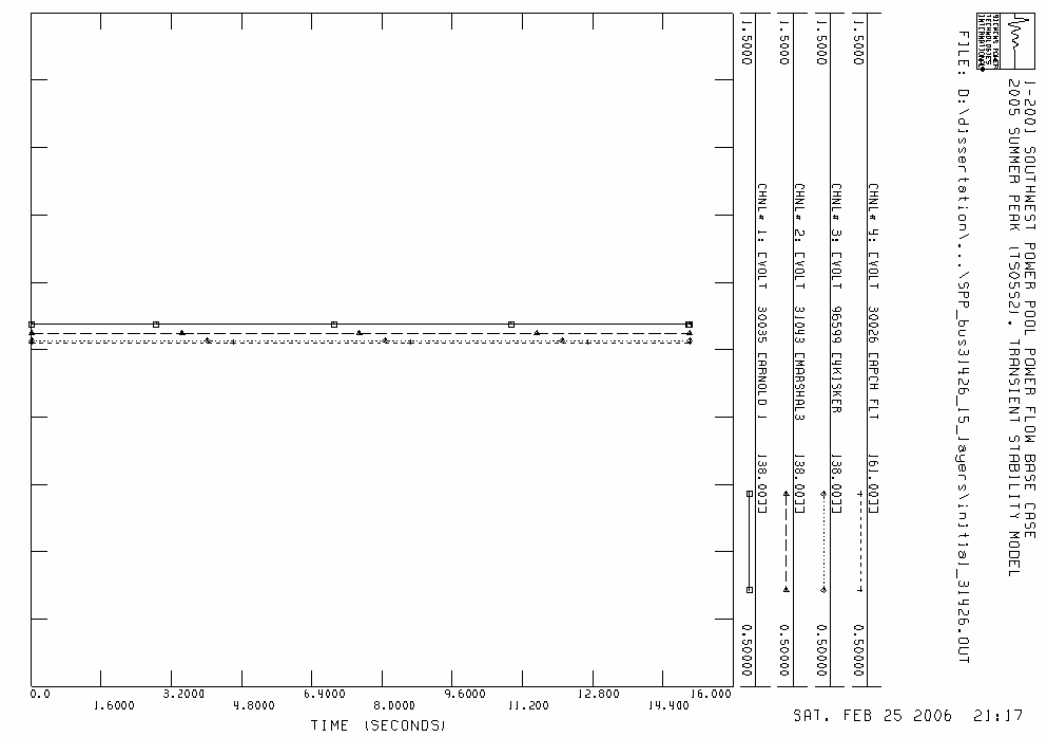
Channel 3: Power bus 96599 (138 kV)



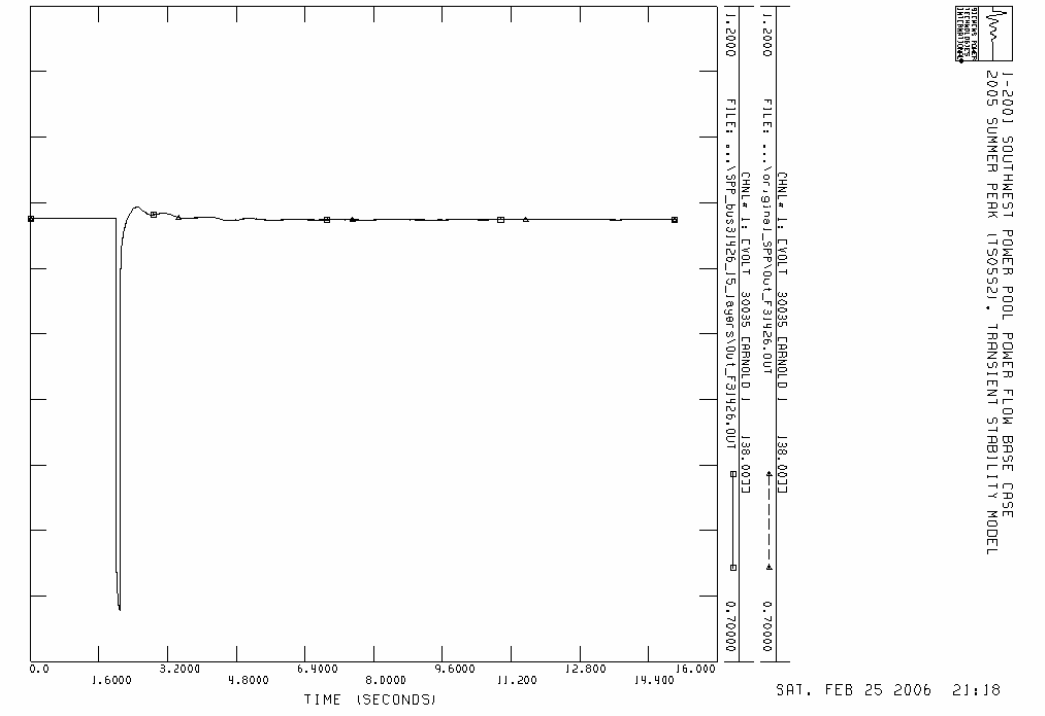

Channel 4: Power bus 30026 (161 kV)



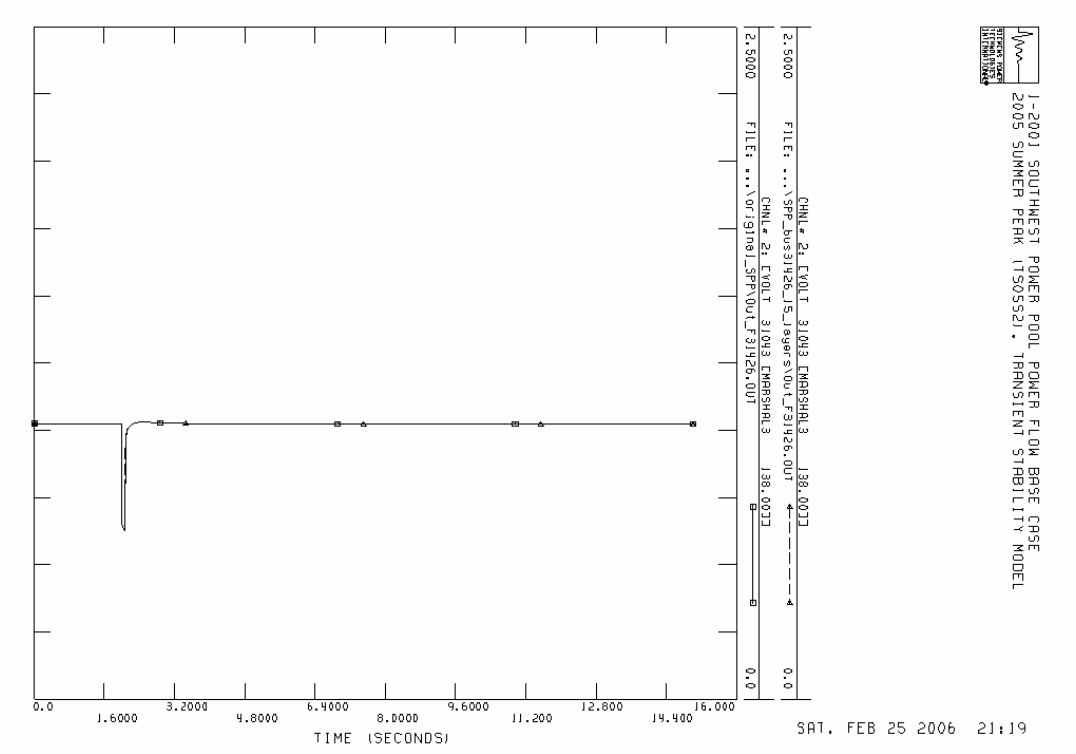
SPP_bus31426_15_layers
Initial condition for channel 1,2,3,4



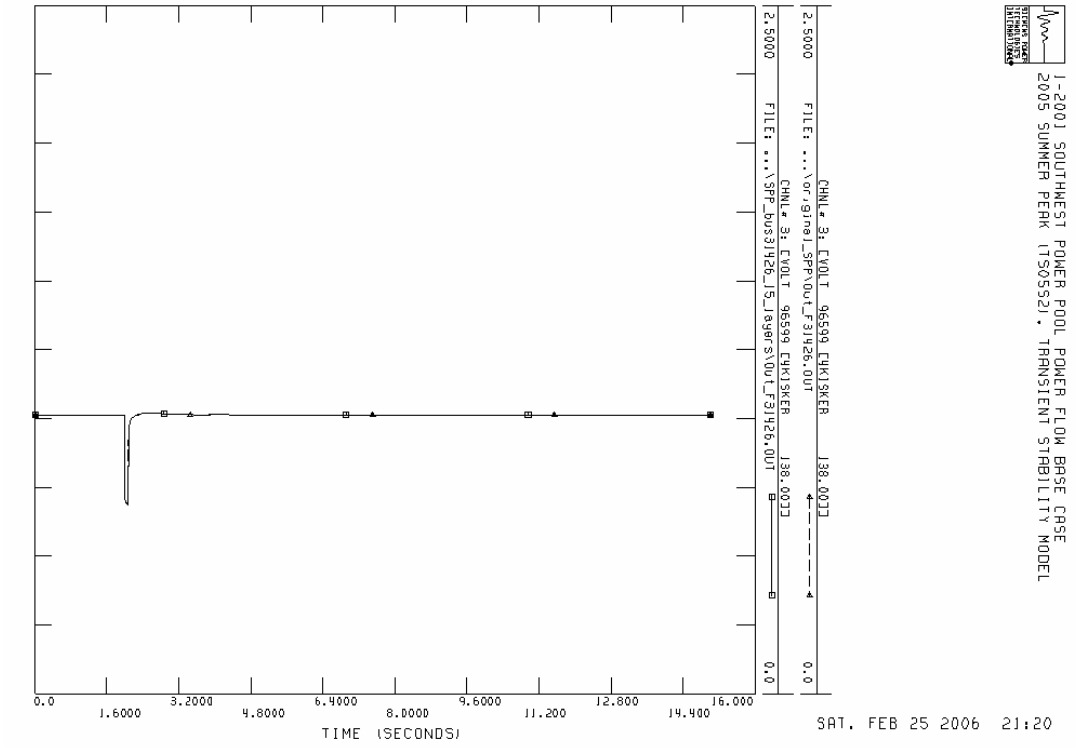

Channel 1: Power bus 30035 (138 kV)



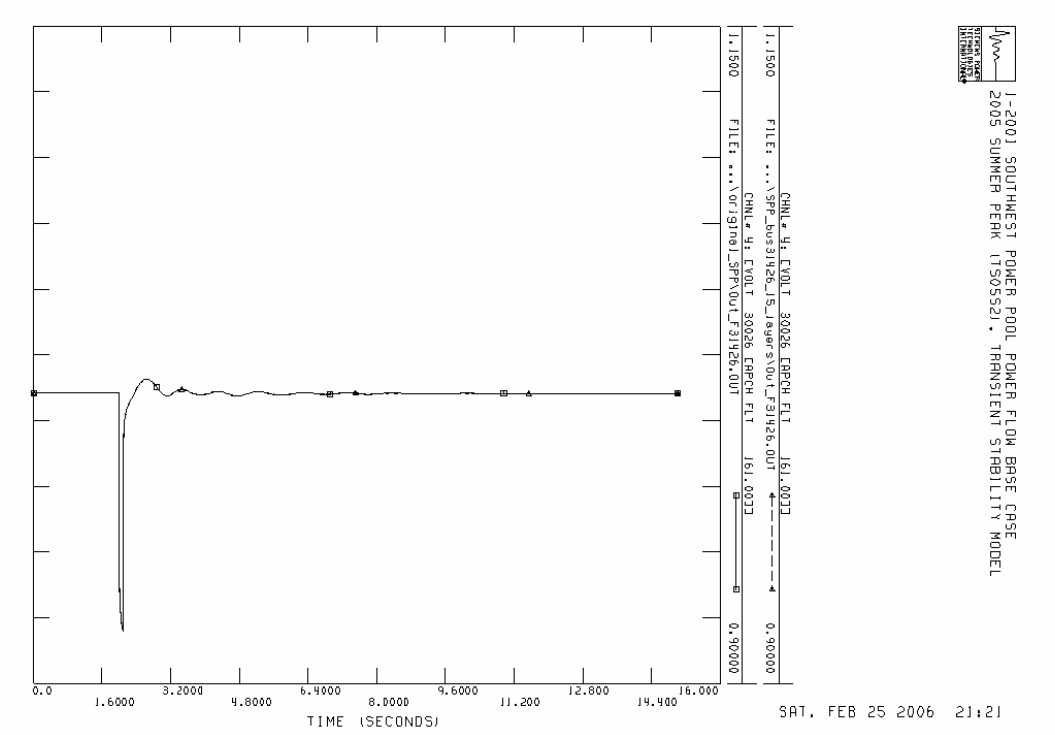
Channel 2: Power bus 31043 (138 kV)




Channel 3: Power bus 96599 (138 kV)

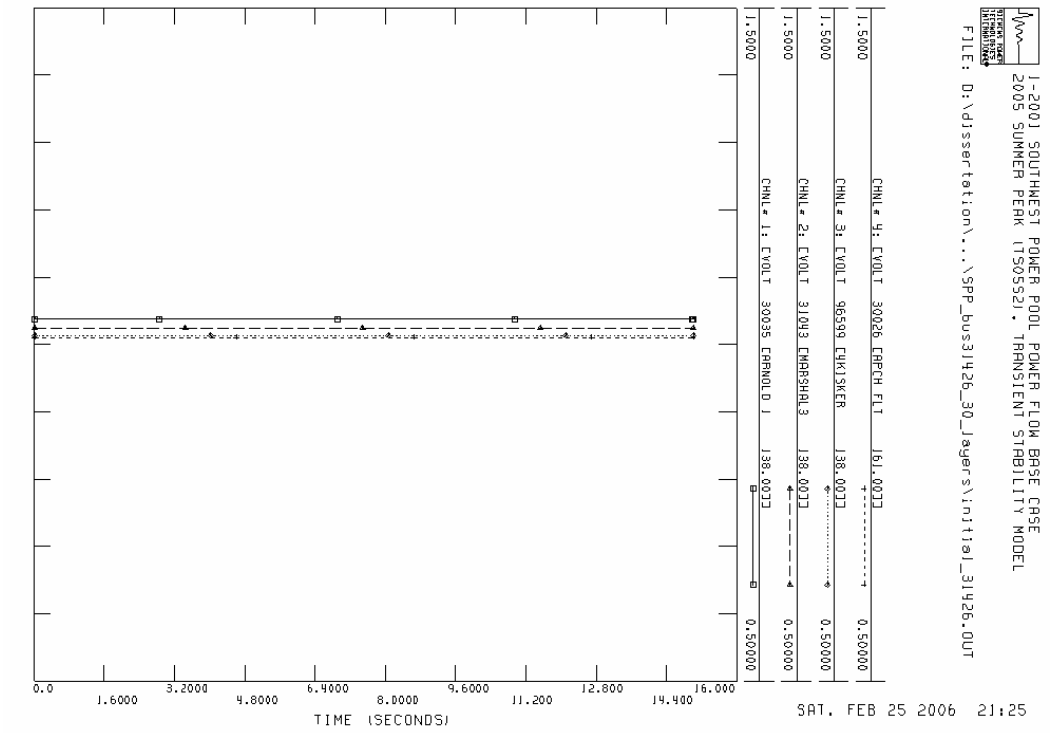


Channel 4: Power bus 30026 (161 kV)

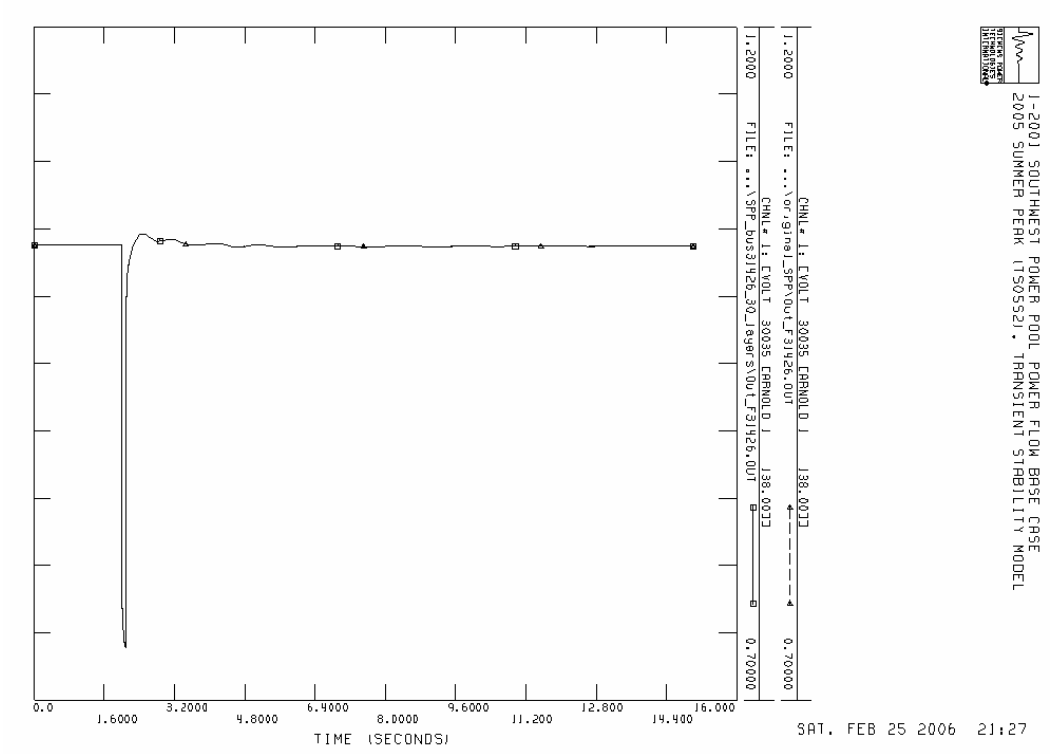



SPP_bus31426_30_layers

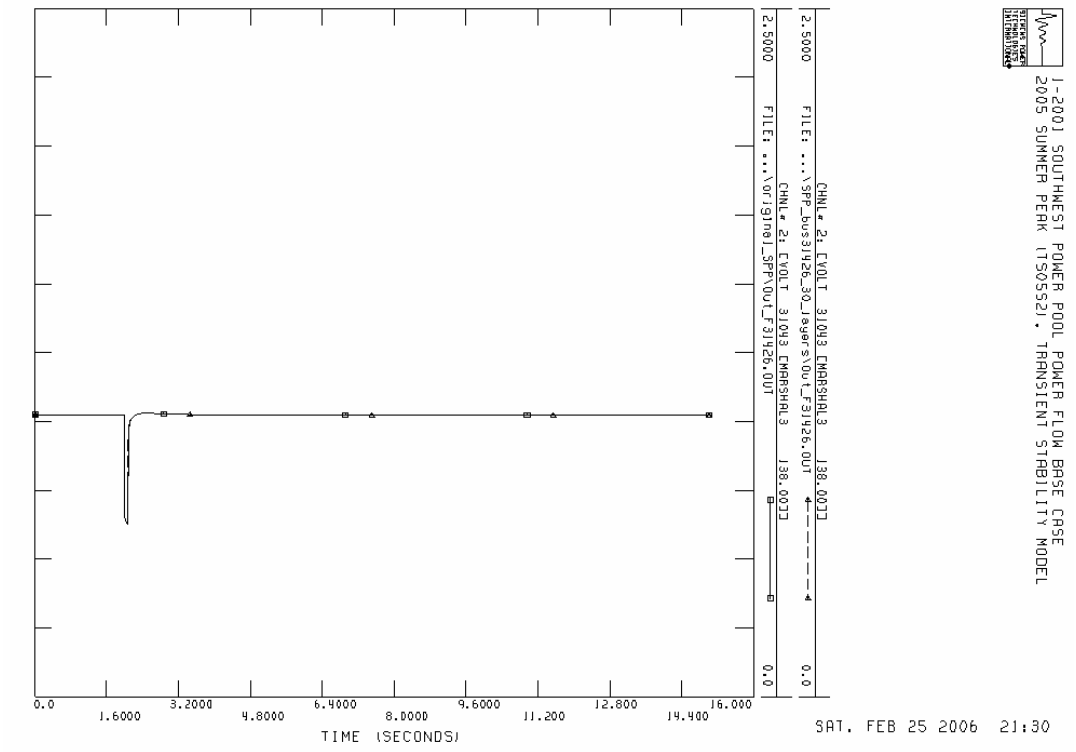
Initial condition for channel 1,2,3,4



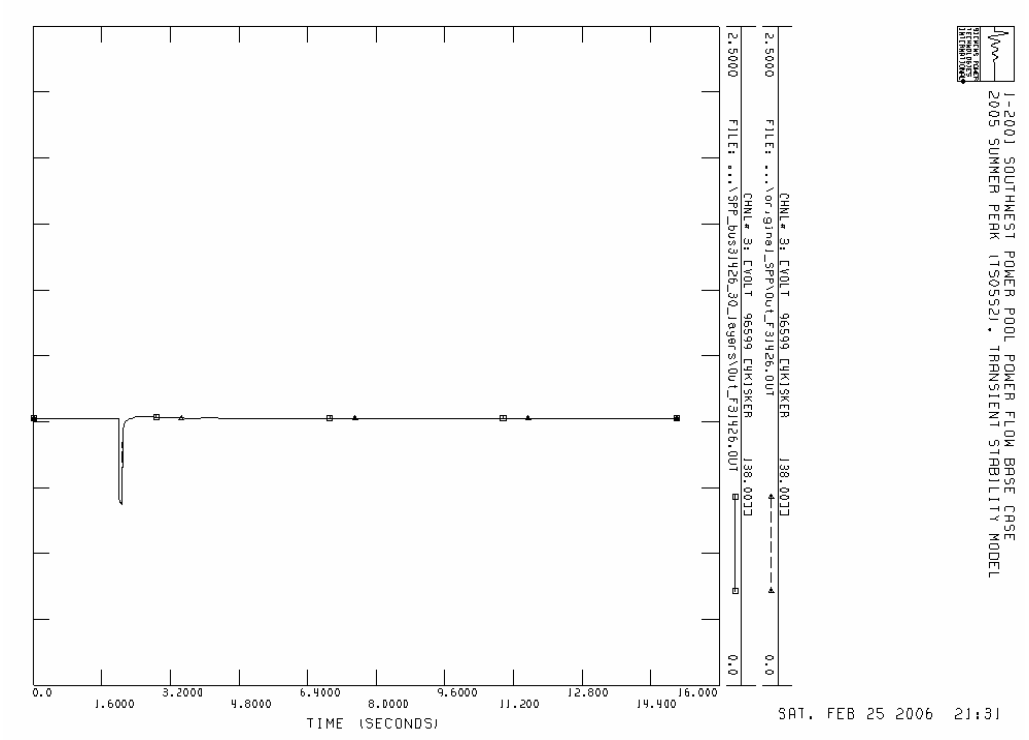
Channel 1: Power bus 30035 (138 kV)



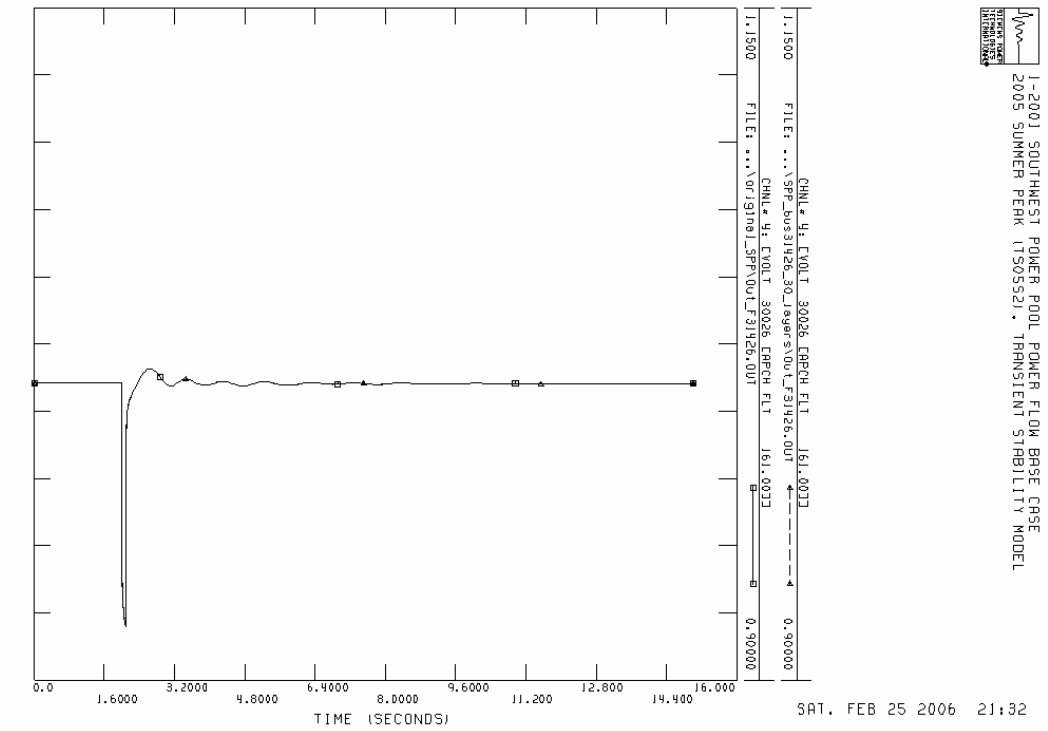

Channel 2: Power bus 31043 (138 kV)



Channel 3: Power bus 96599 (138 kV)

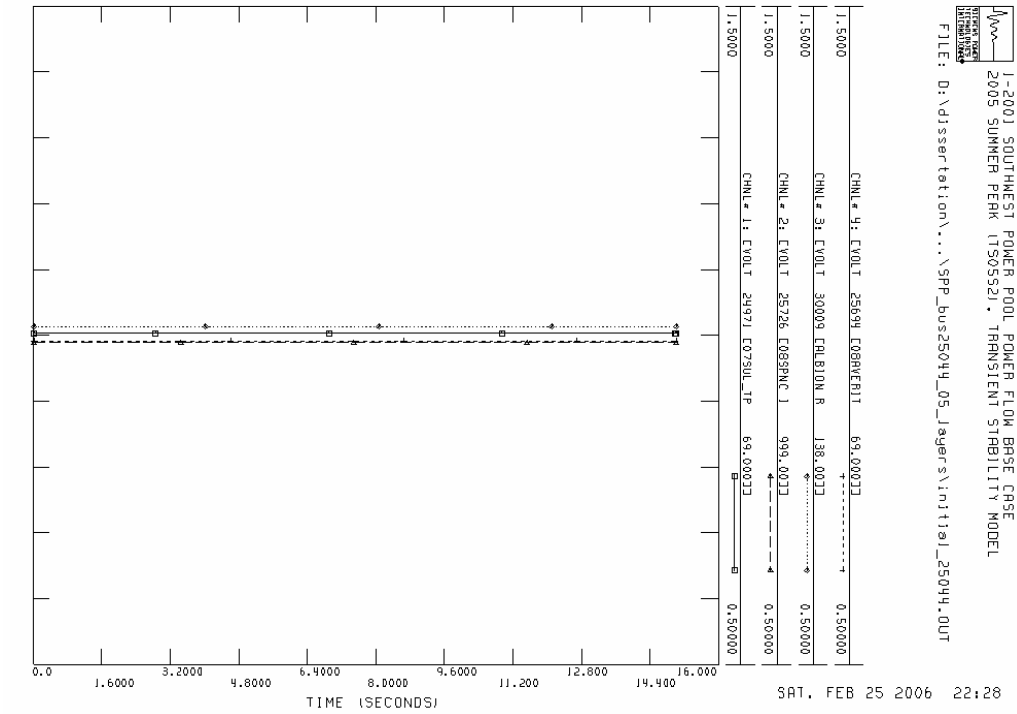



Channel 4: Power bus 30026 (161 kV)

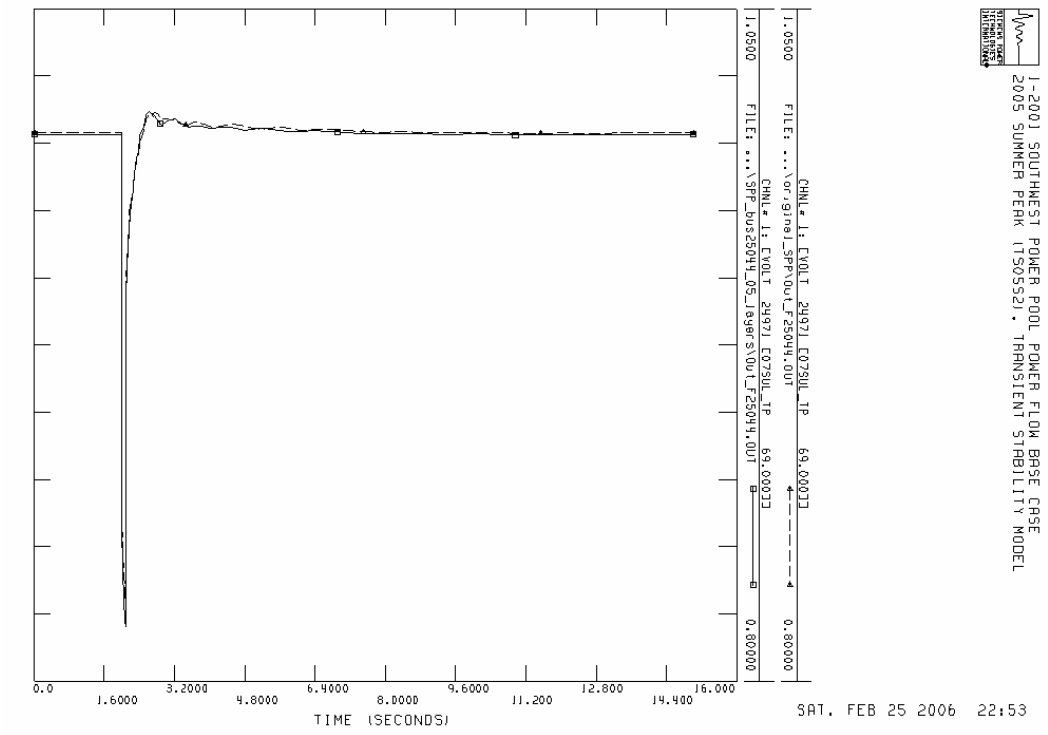


SPP_bus25044_5_layers

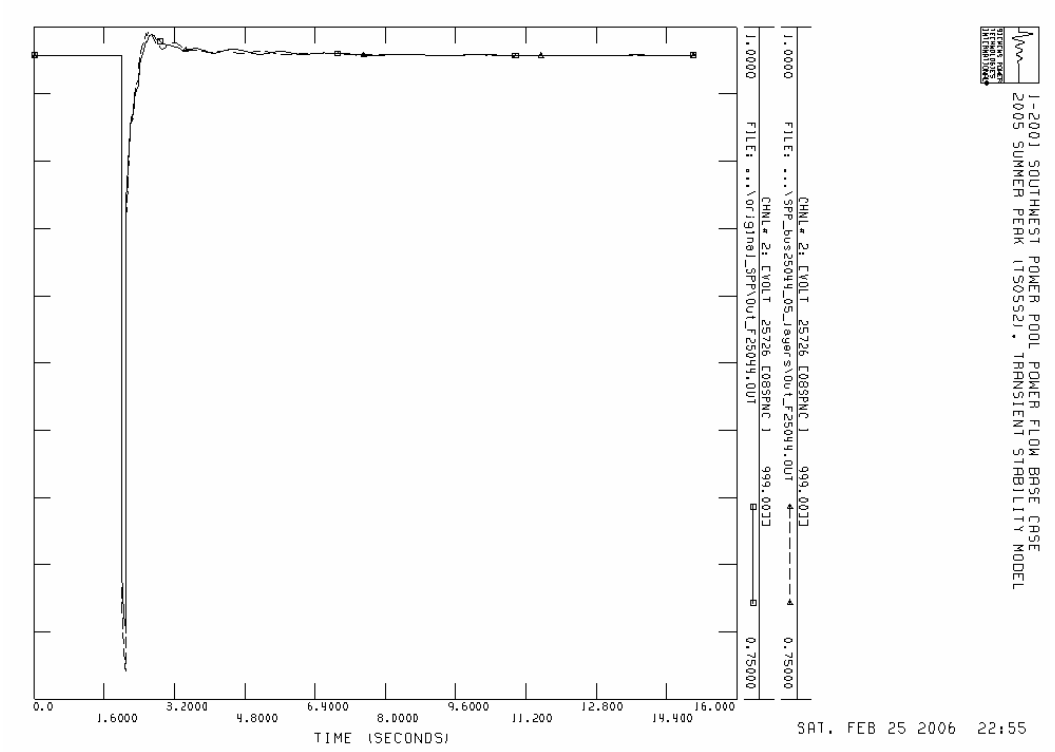
Initial condition for channel 1,2,3,4



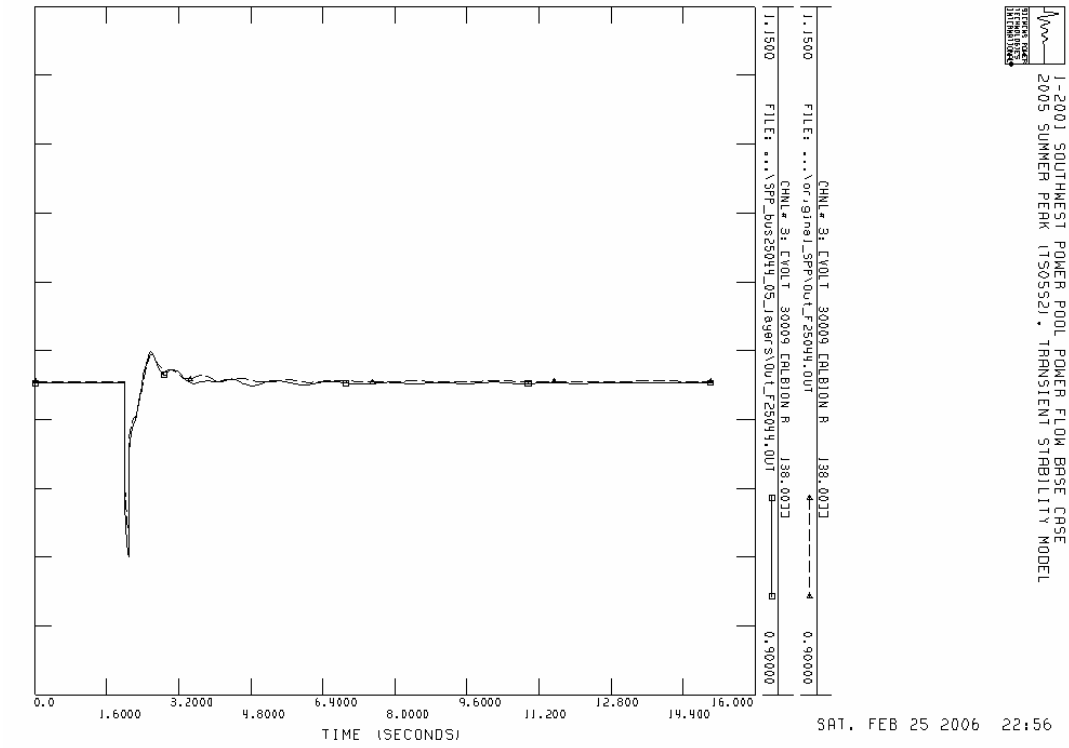

Channel 1: Power bus 24971 (69 kV)



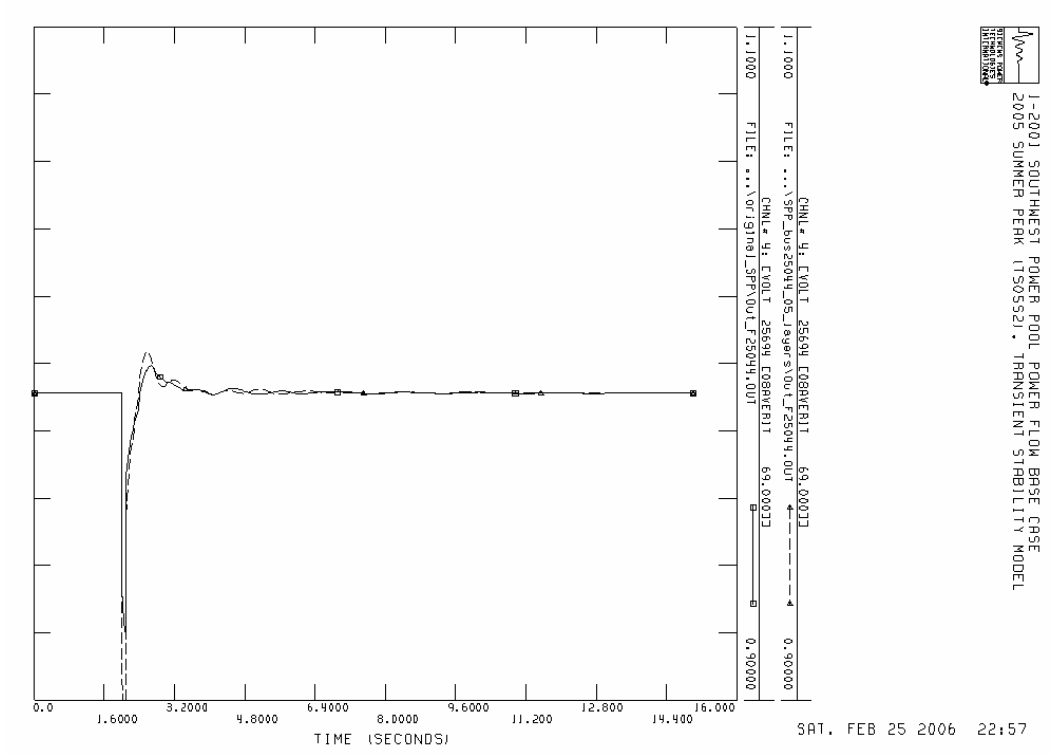
Channel 2: Power bus 25726 (999 kV)




Channel 3: Power bus 30009 (138 kV)

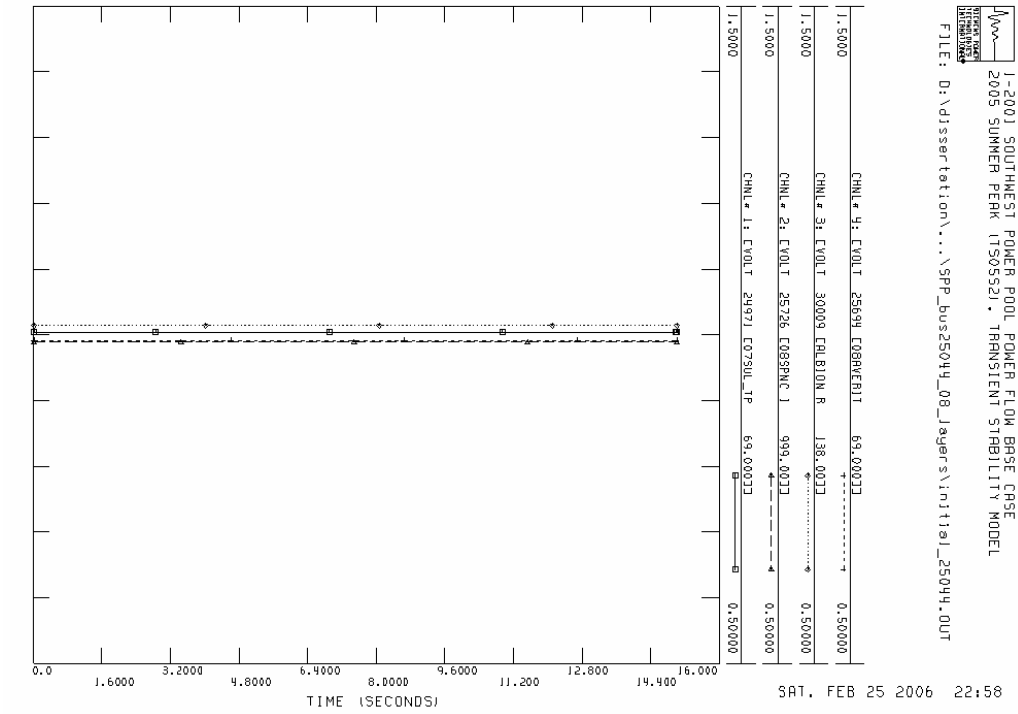


Channel 4: Power bus 25694 (69 kV)

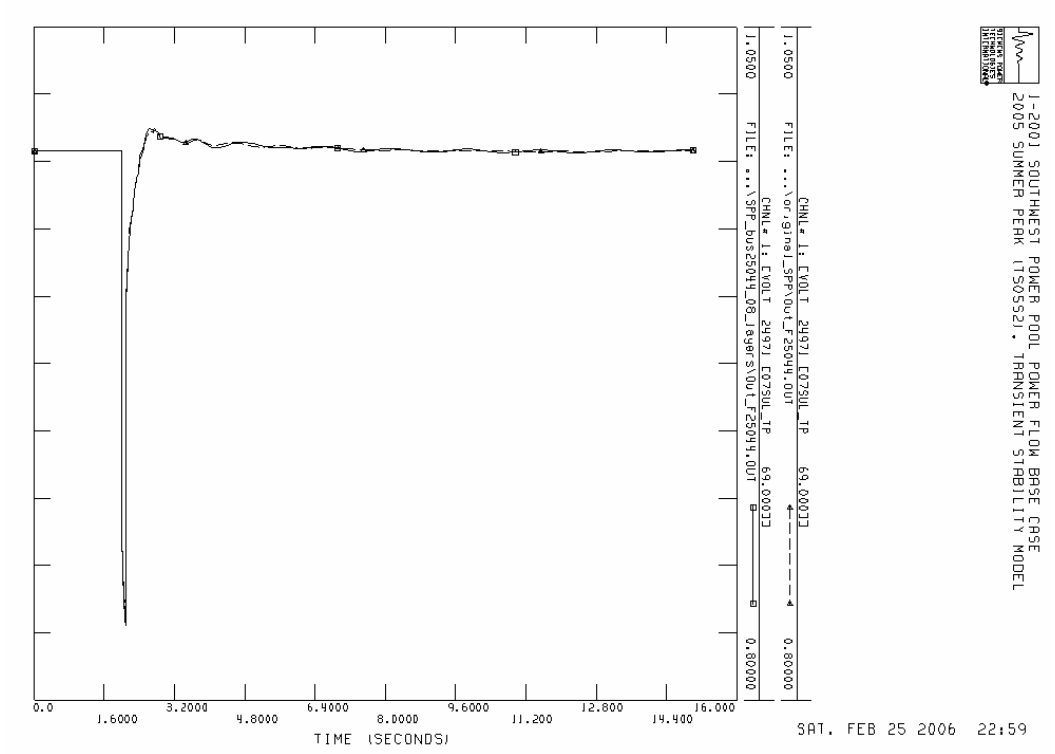



SPP_bus25044_8_layers

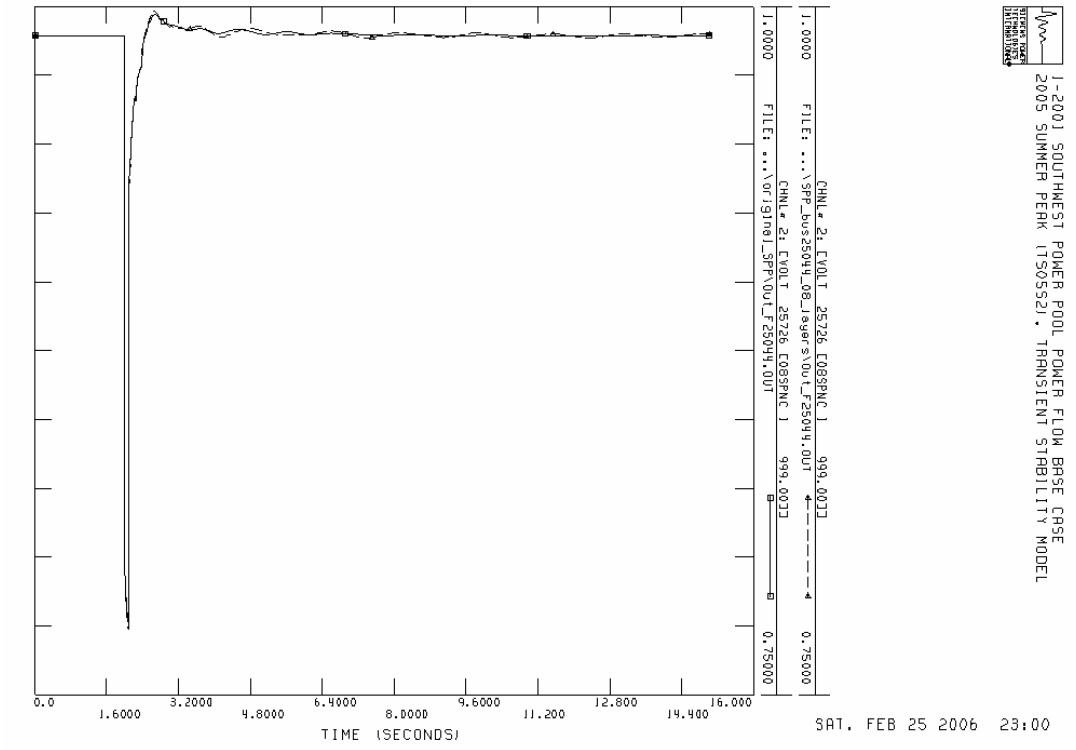
Initial condition for channel 1,2,3,4



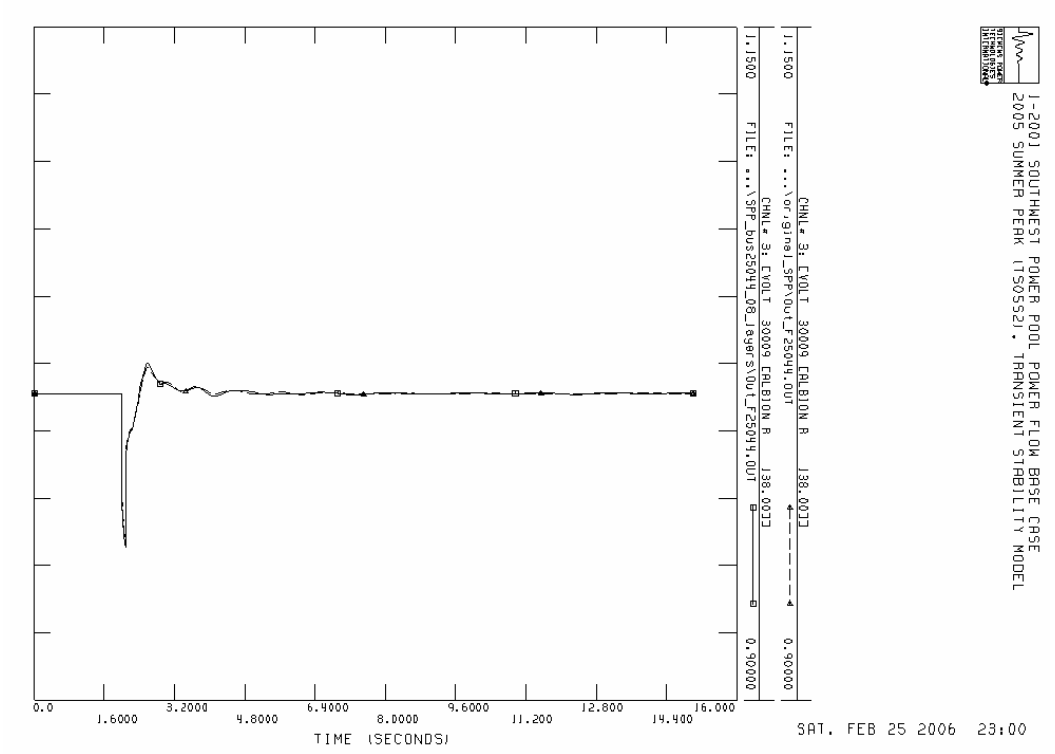
Channel 1: Power bus 24971 (69 kV)



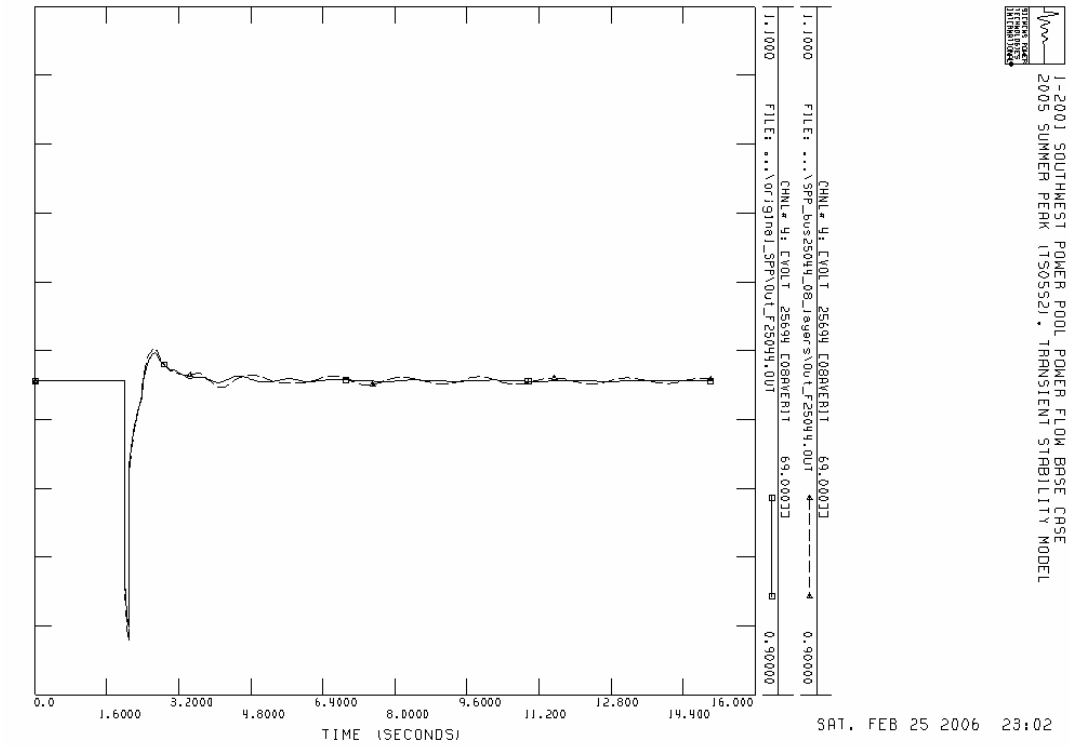

Channel 2: Power bus 25726 (999 kV)



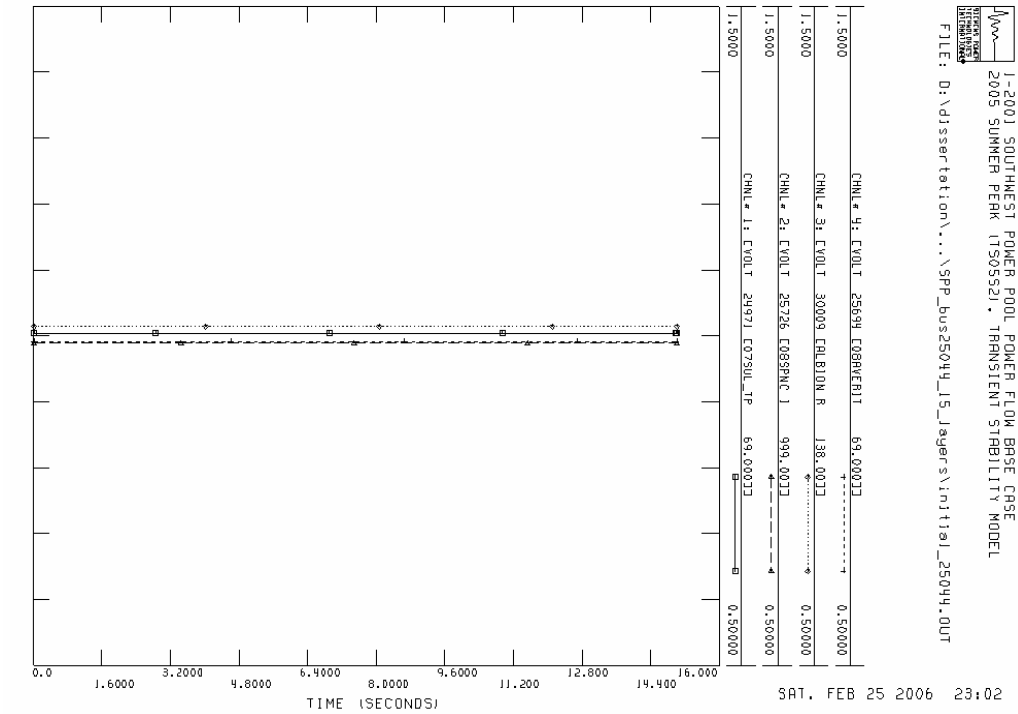
Channel 3: Power bus 30009 (138 kV)



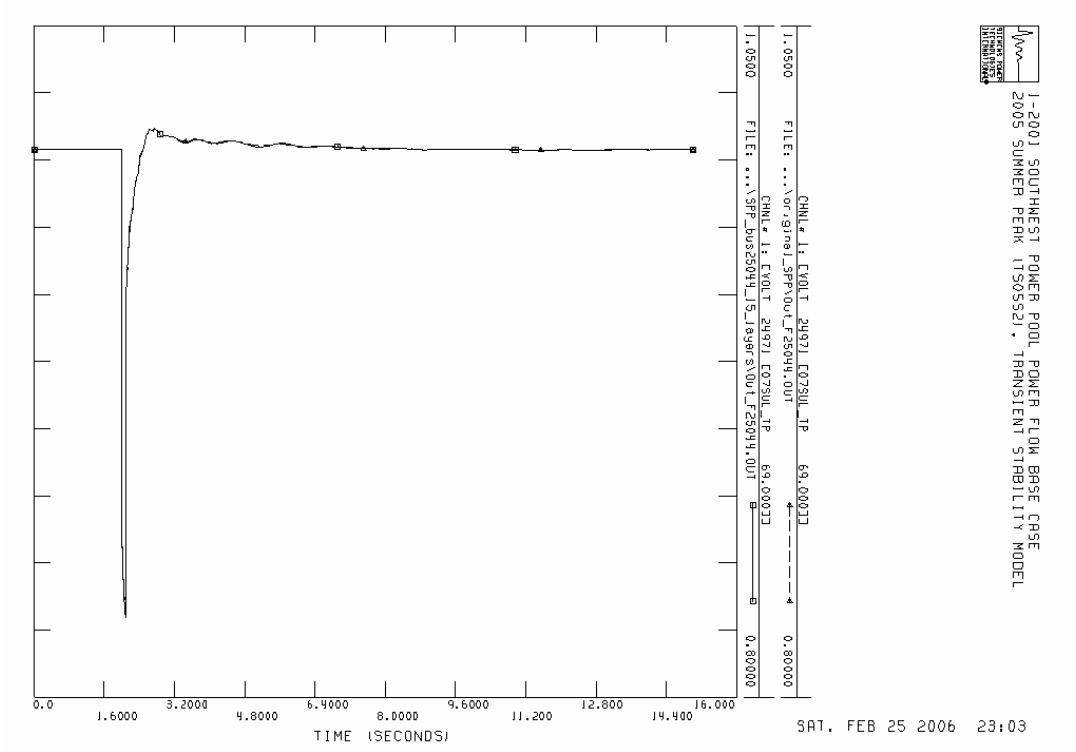

Channel 4: Power bus 25694 (69 kV)



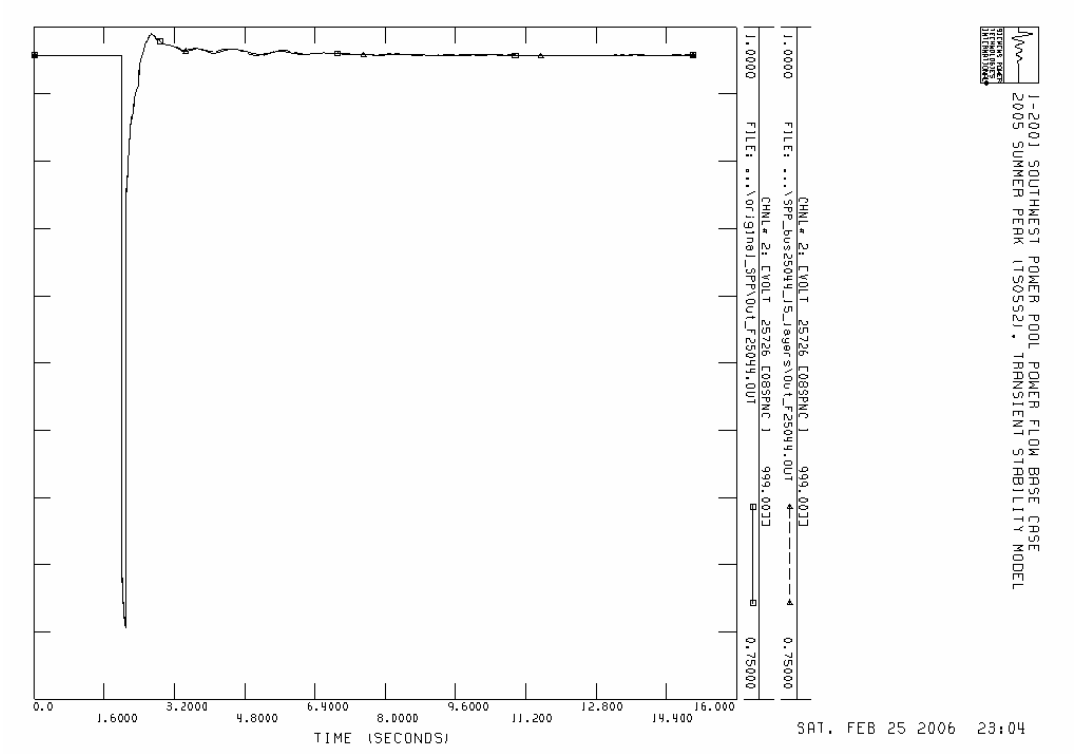
SPP_bus25044_15_layers
Initial condition for channel 1,2,3,4



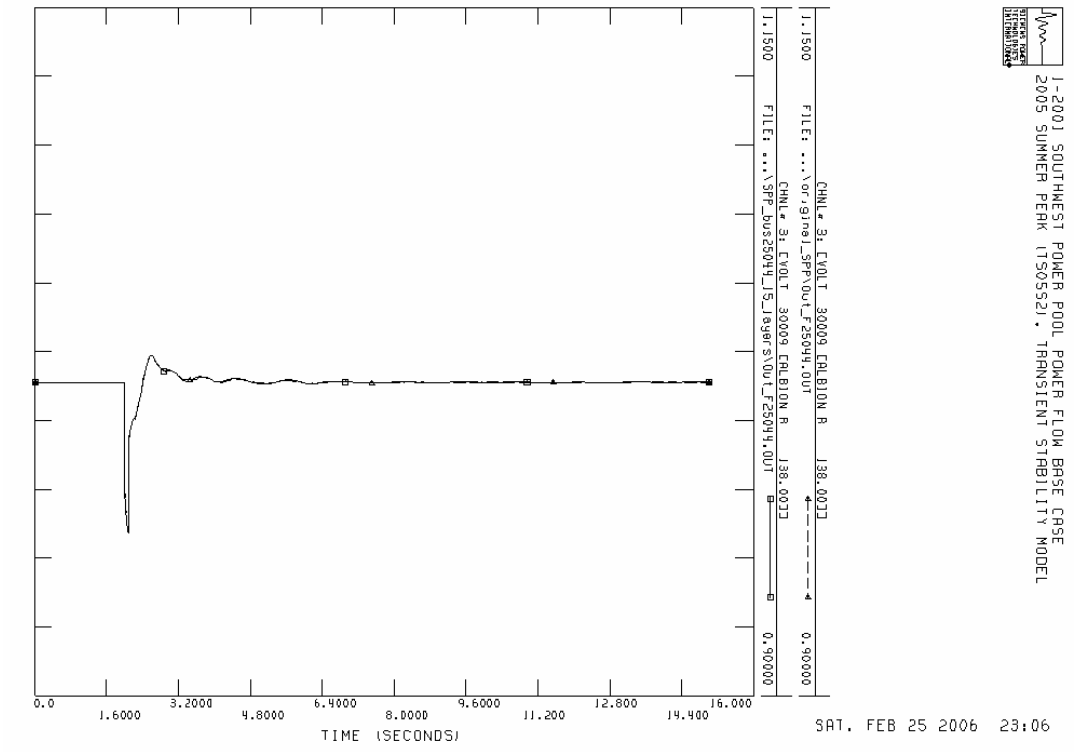

Channel 1: Power bus 24971 (69 kV)



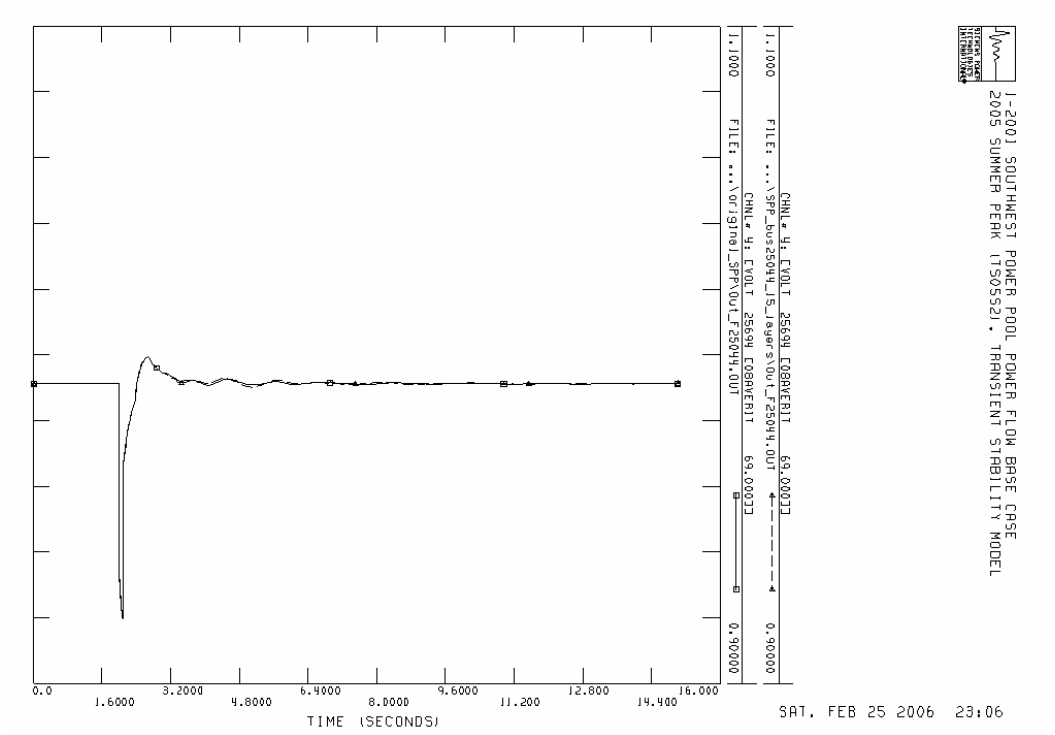
Channel 2: Power bus 25726 (999 kV)




Channel 3: Power bus 30009 (138 kV)

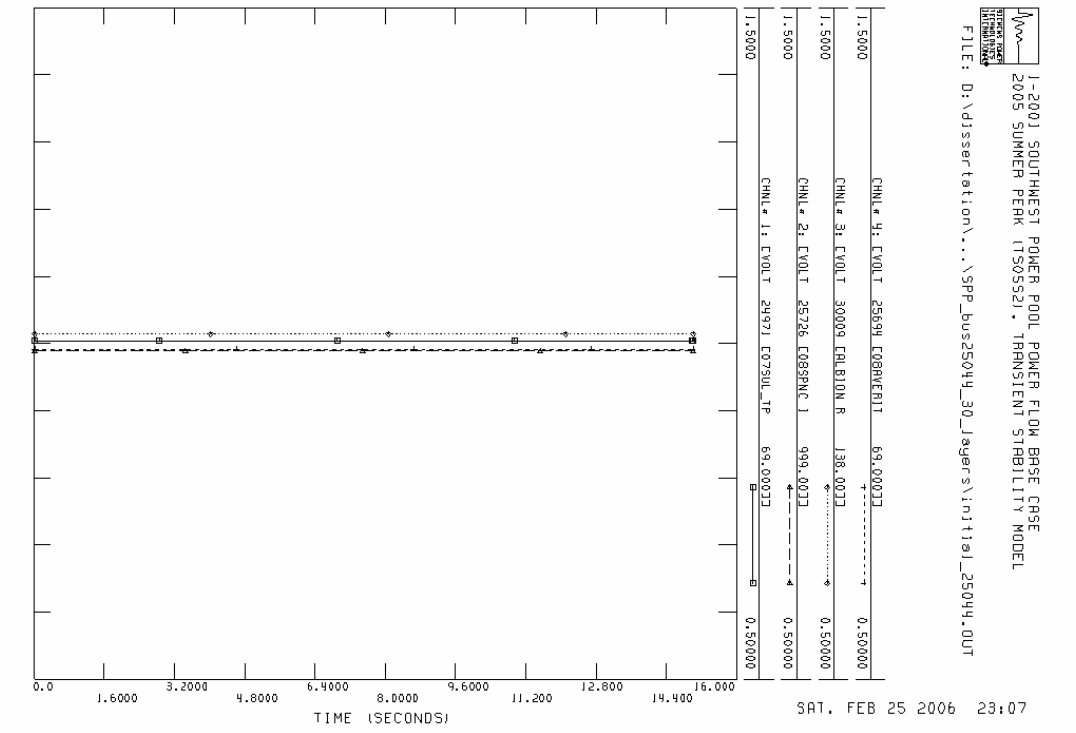


Channel 4: Power bus 25694 (69 kV)

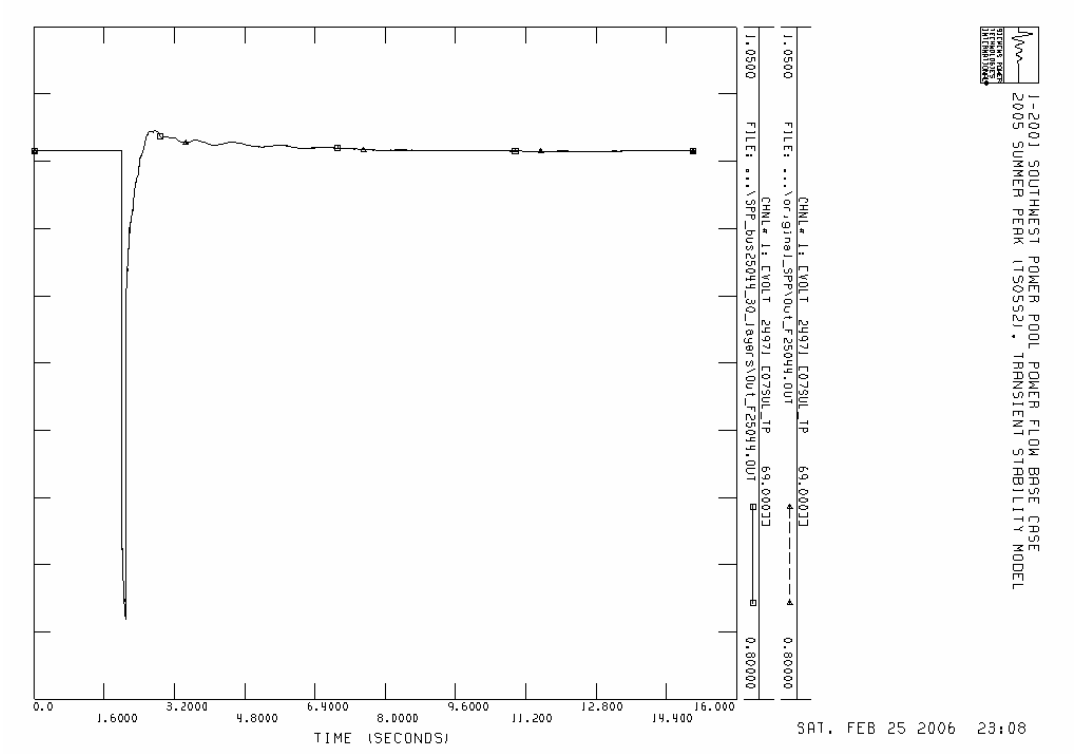



SPP_bus25044_30_layers

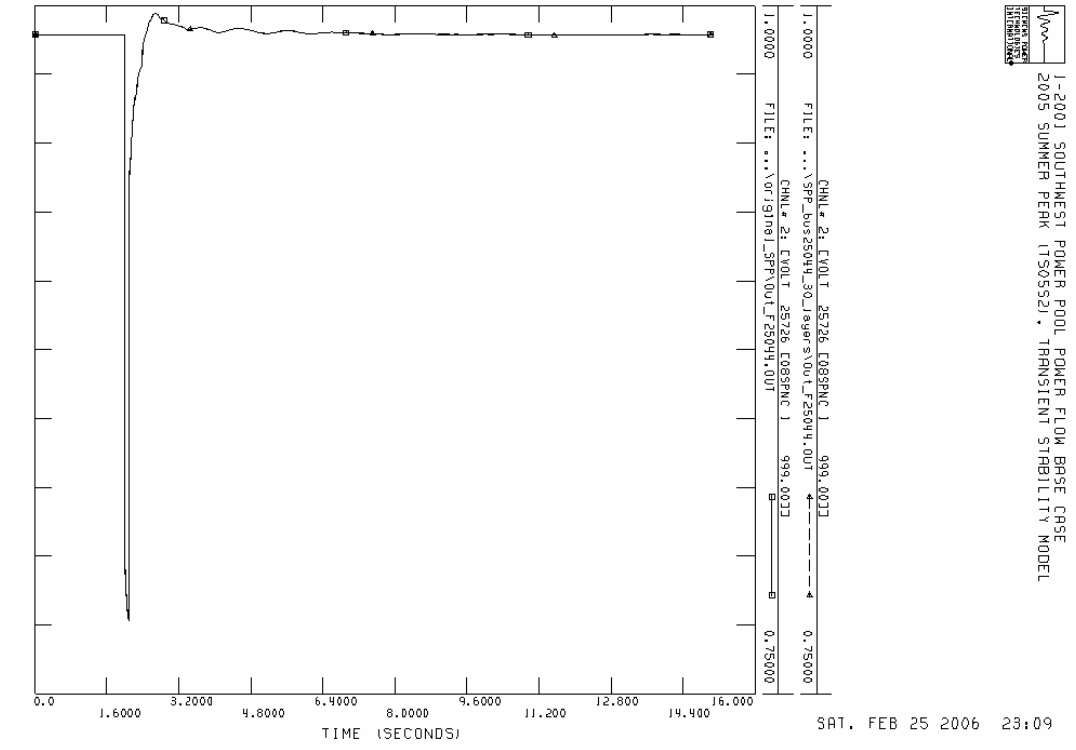
Initial condition for channel 1,2,3,4



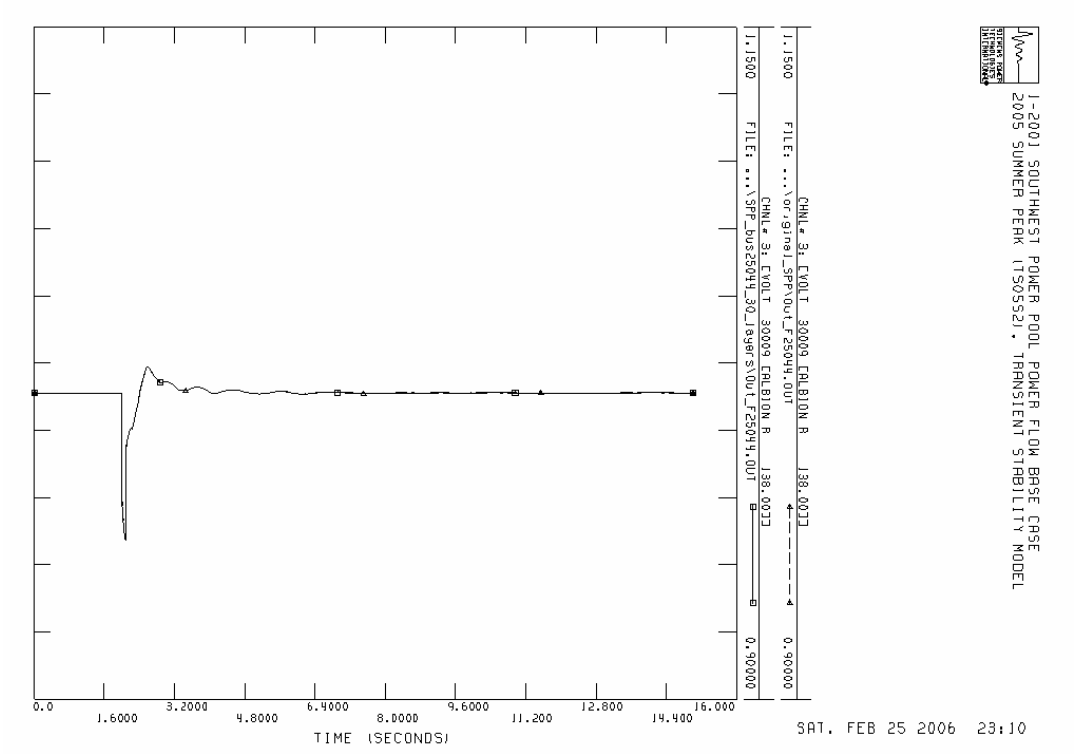
Channel 1: Power bus 24971 (69 kV)



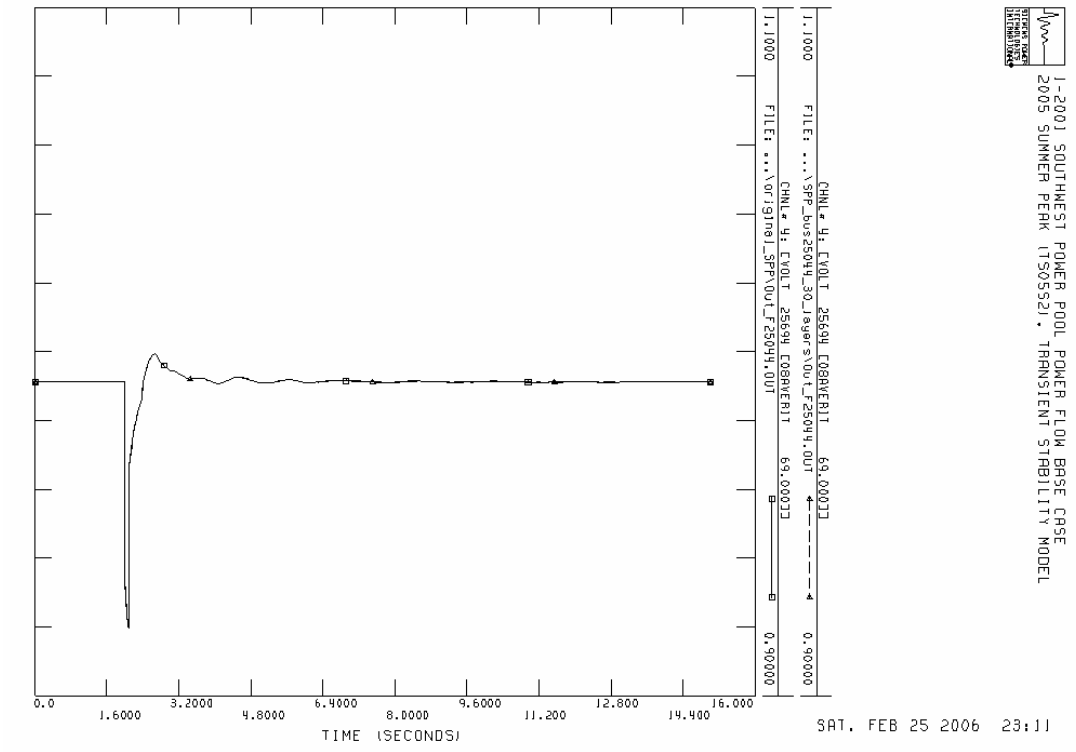

Channel 2: Power bus 25726 (999 kV)



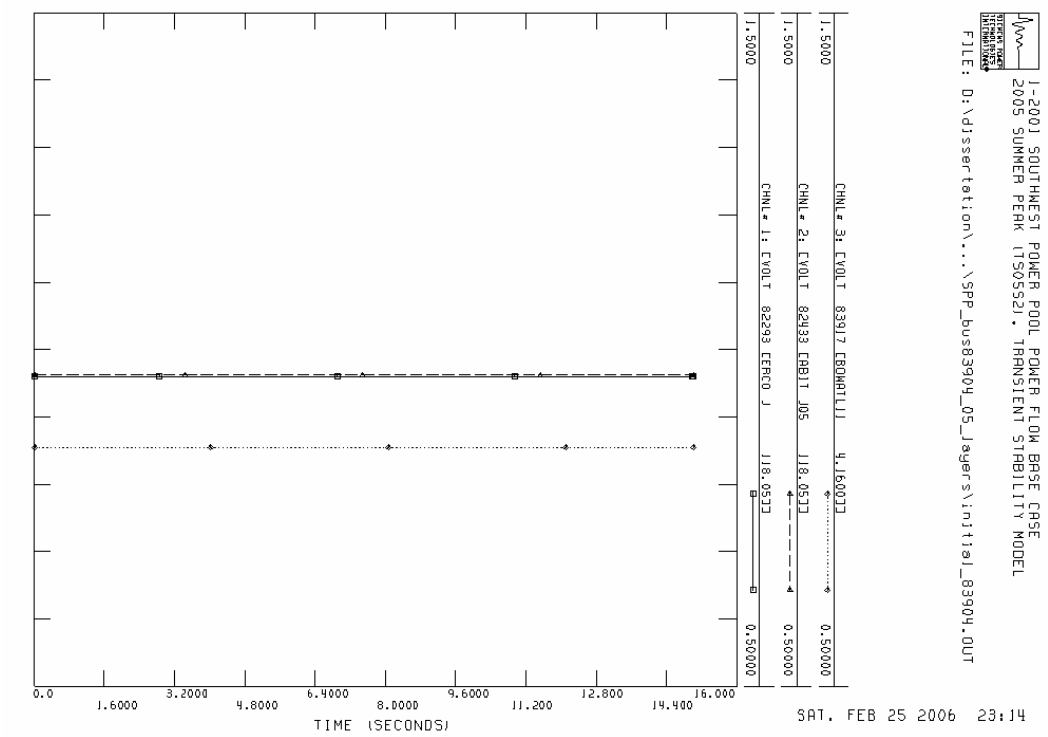
Channel 3: Power bus 30009 (138 kV)



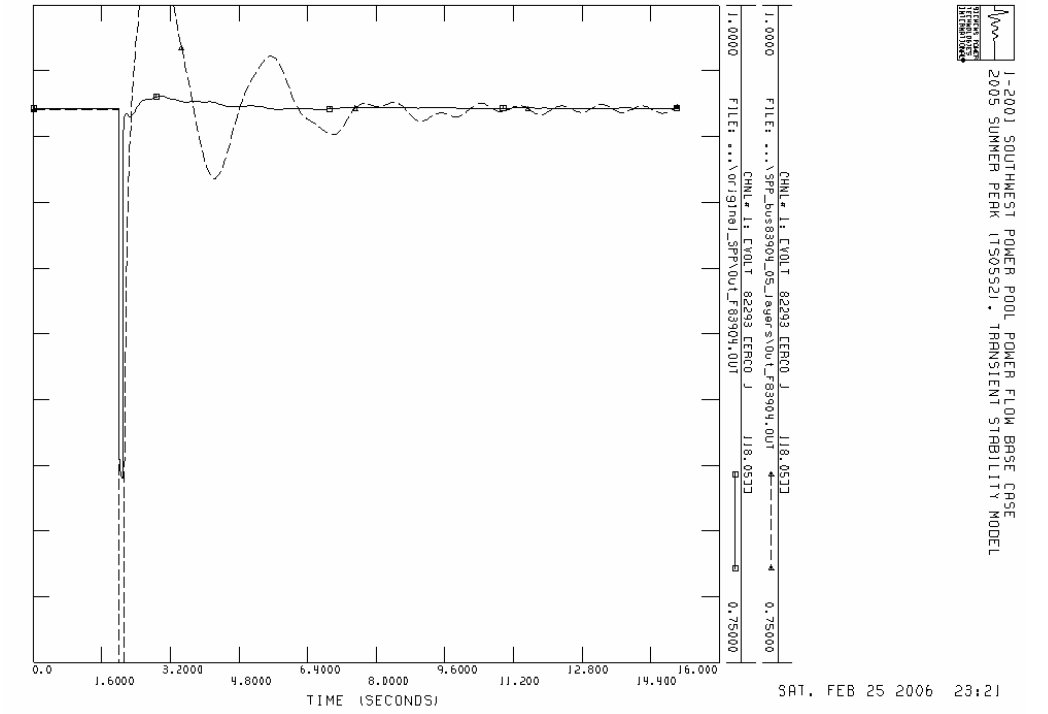

Channel 4: Power bus 25694 (69 kV)



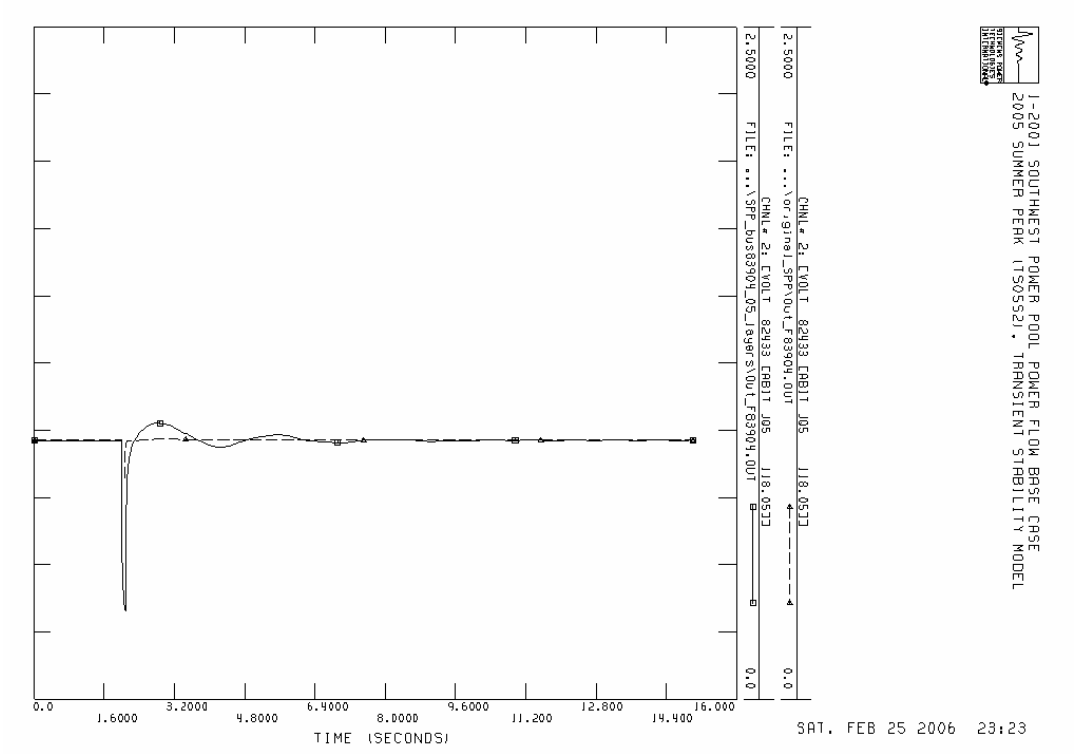
SPP_bus83904_5_layers
Initial condition for channel 1,2,3



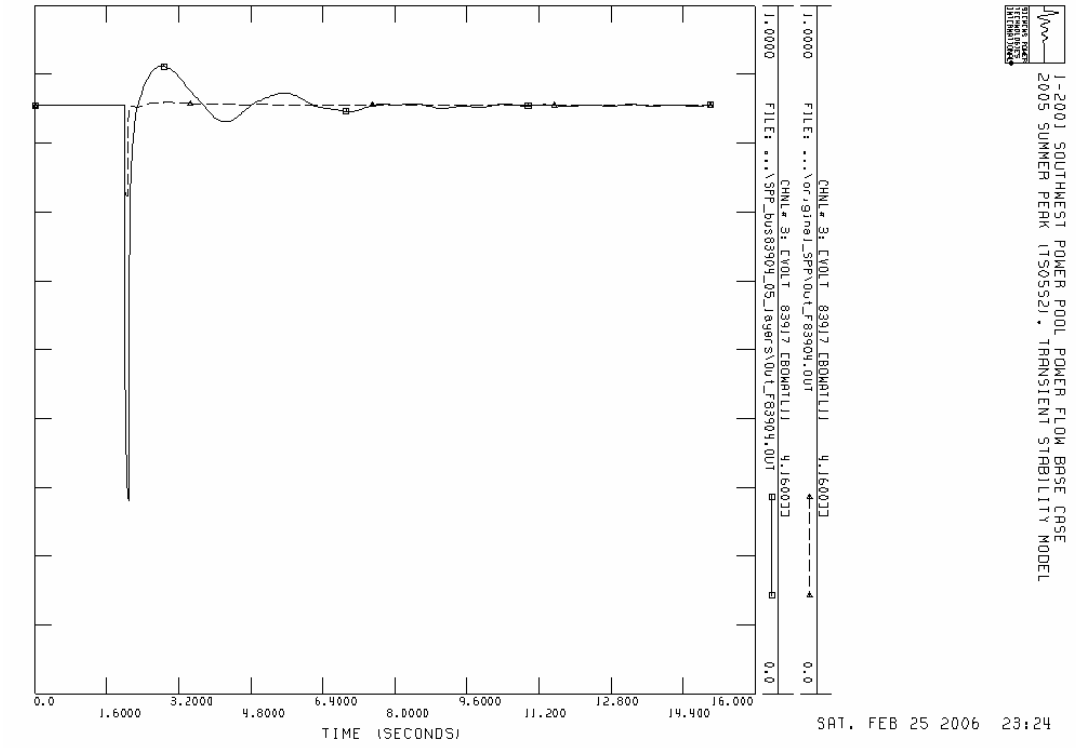

Channel 1: Power bus 82293 (118.05 kV)



Channel 2: Power bus 82433 (118.05 kV)

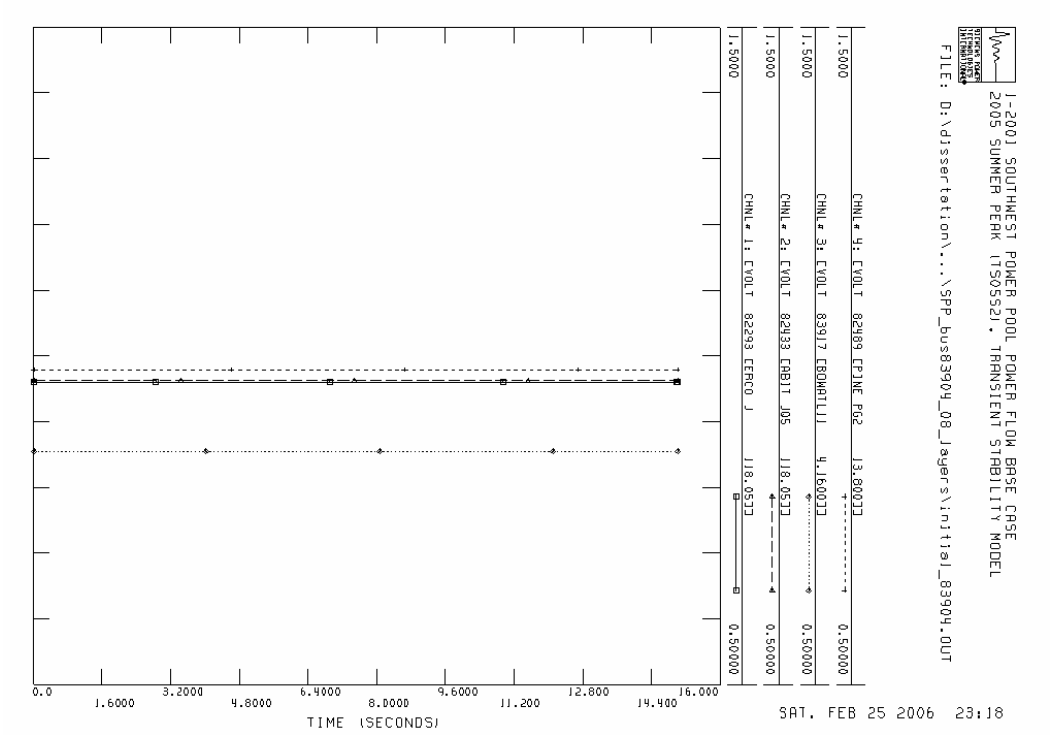



Channel 3: Power bus 83917 (4.16 kV)

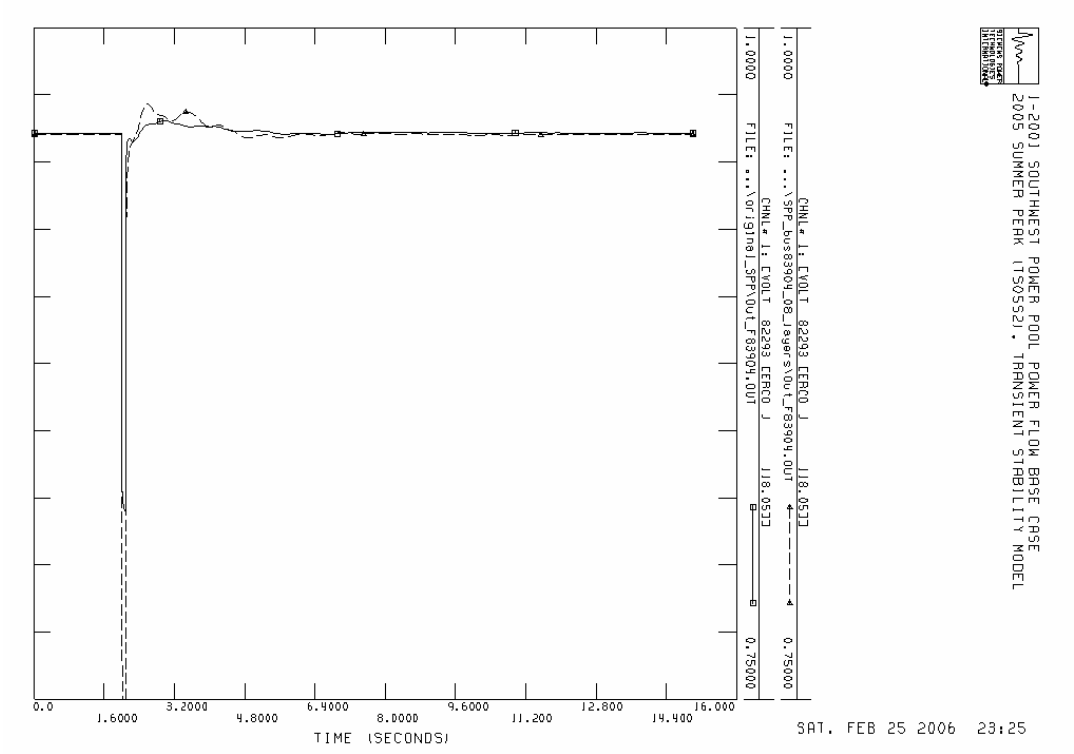


SPP_bus83904_8_layers

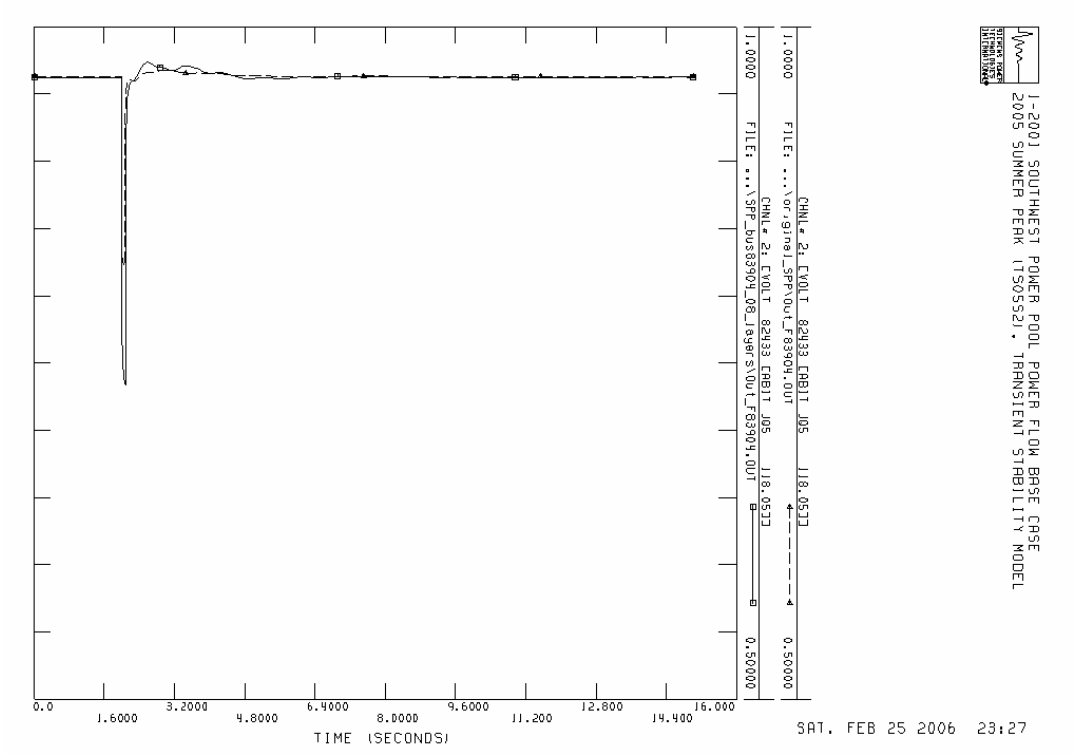
Initial condition for channel 1,2,3,4



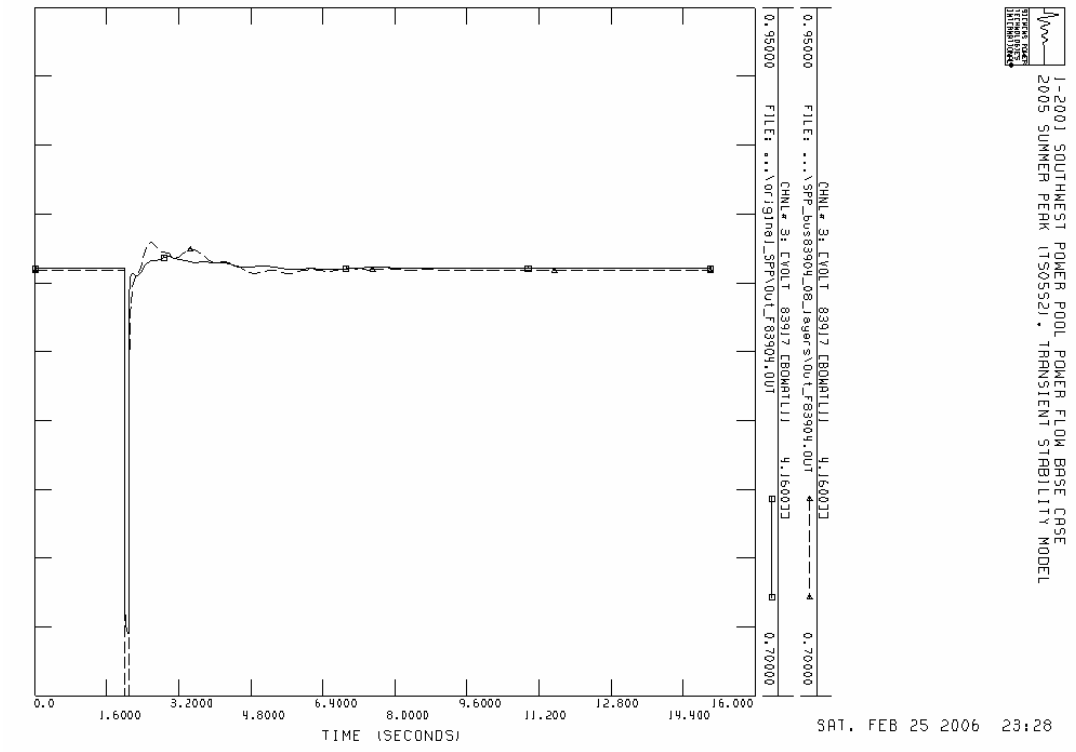

Channel 1: Power bus 82293 (118.05 kV)



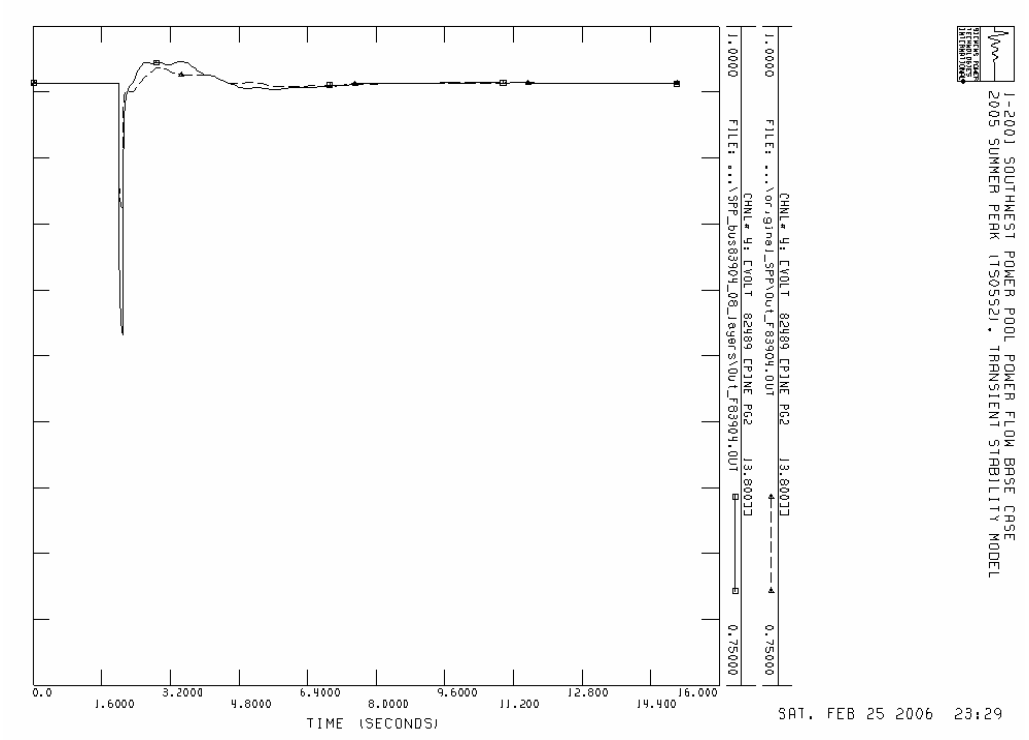
Channel 2: Power bus 82433 (118.05 kV)




Channel 3: Power bus 83917 (4.16 kV)

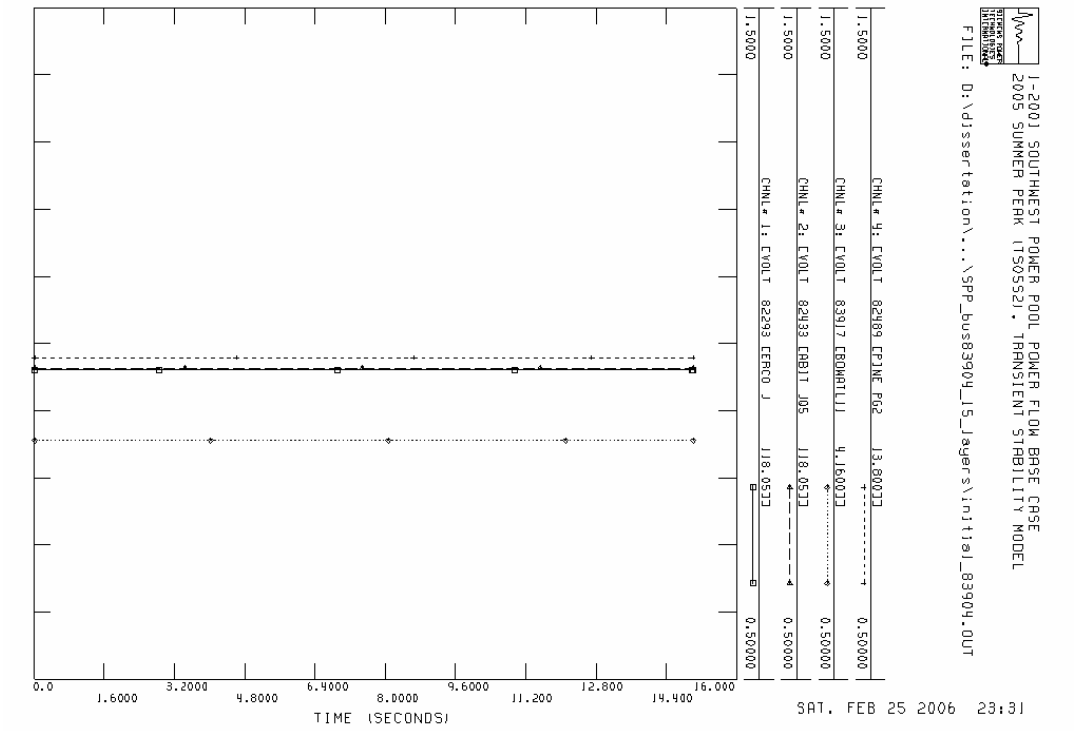


Channel 4: Power bus 82489 (13.8 kV)

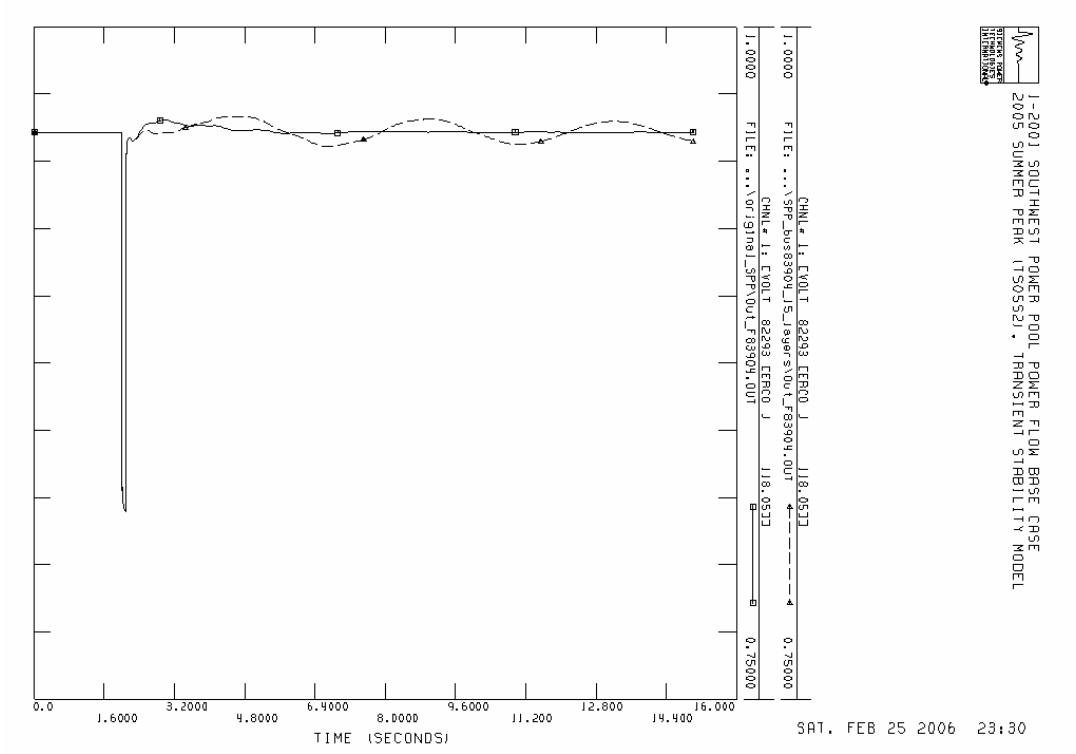



SPP_bus83904_15_layers

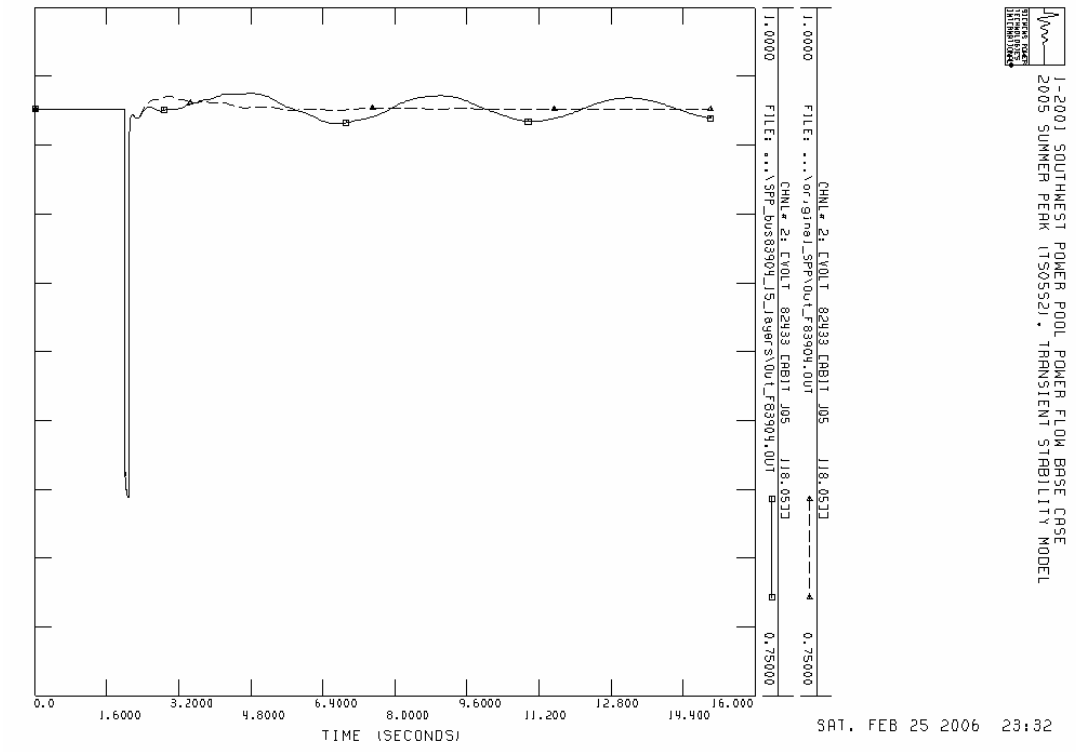
Initial condition for channel 1,2,3,4



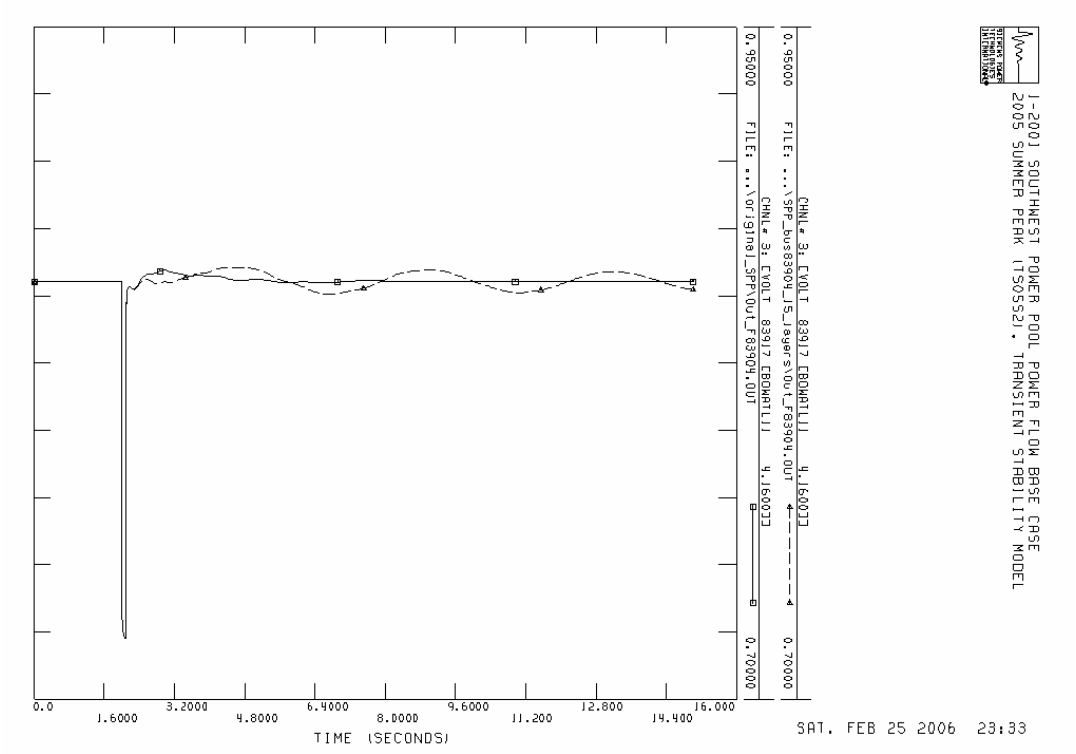
Channel 1: Power bus 82293 (118.05 kV)



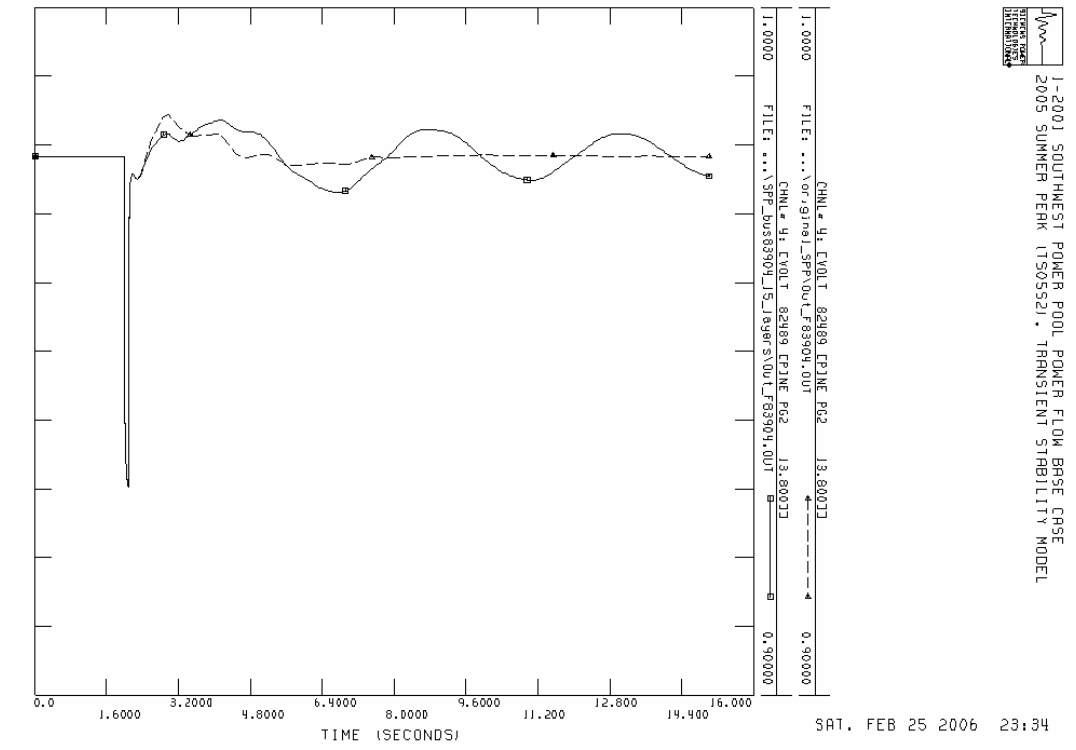

Channel 2: Power bus 82433 (118.05 kV)



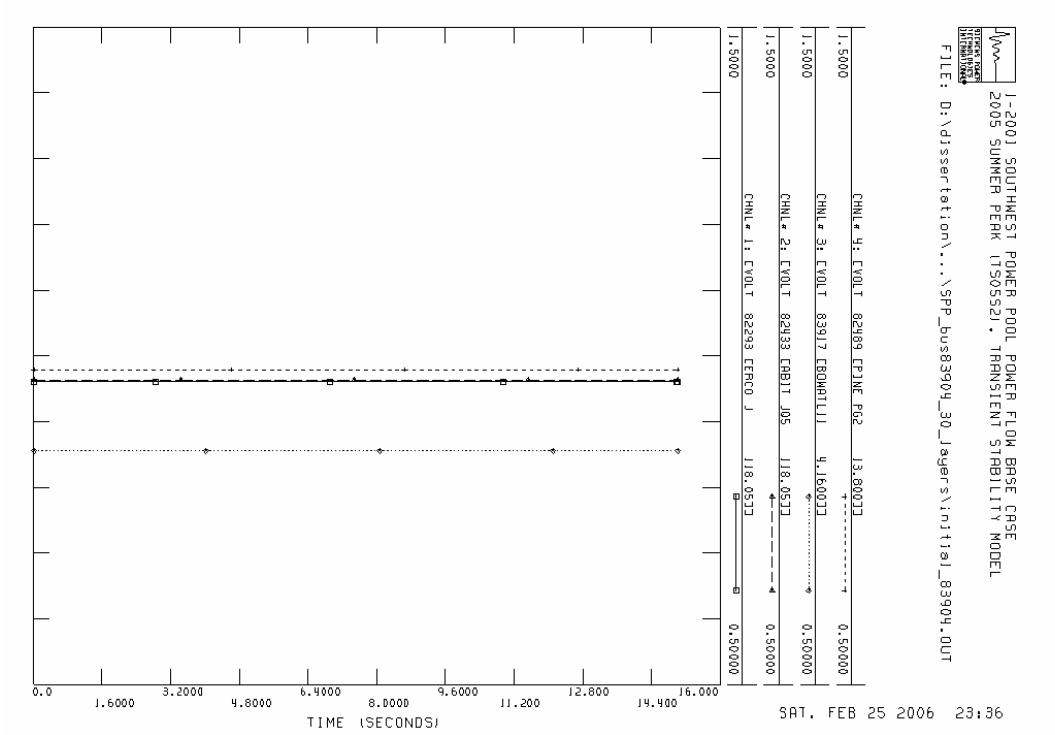
Channel 3: Power bus 83917 (4.16 kV)



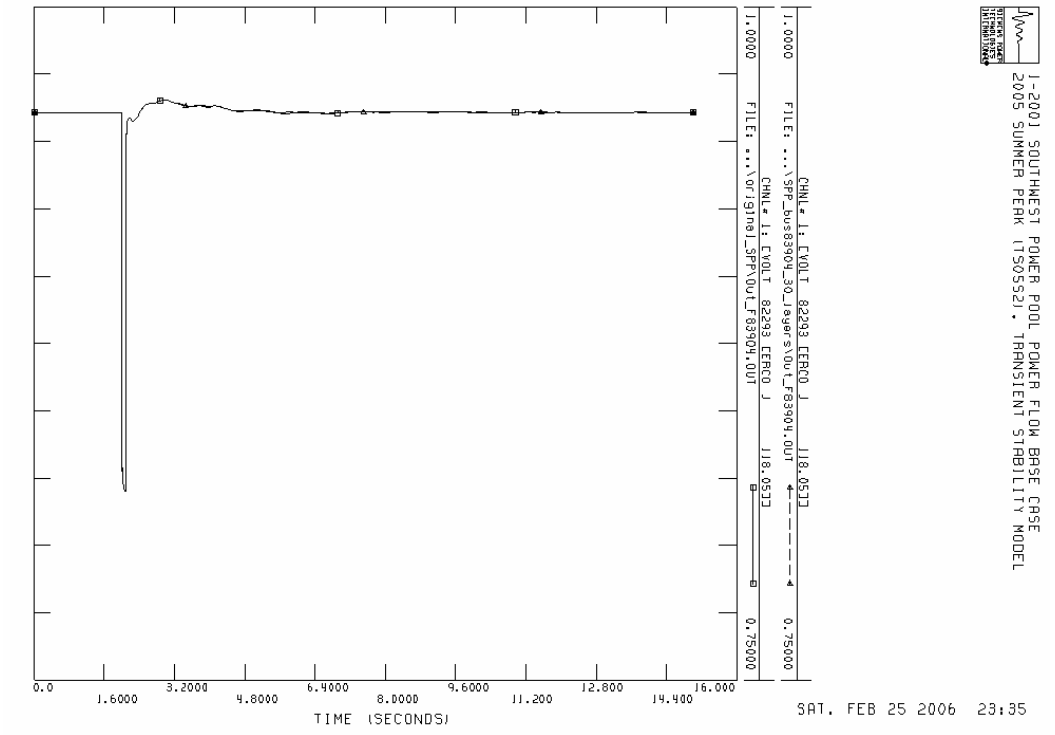

Channel 4: Power bus 82489 (13.8 kV)



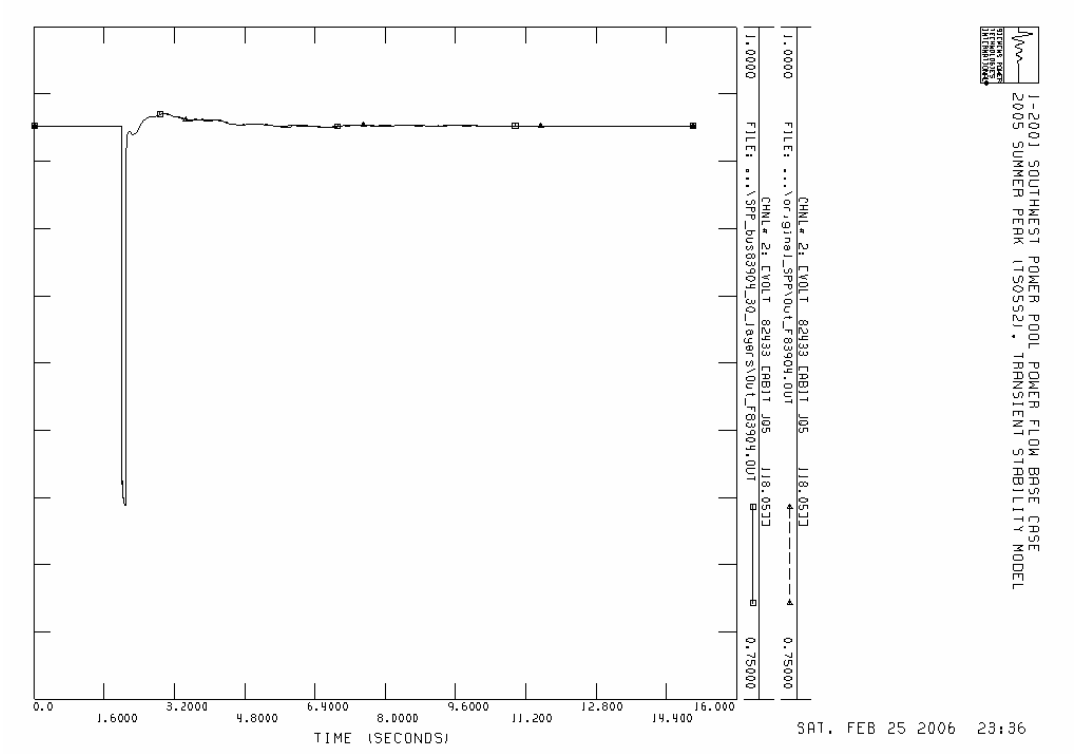
SPP_bus83904_30_layers
Initial condition for channel 1,2,3,4



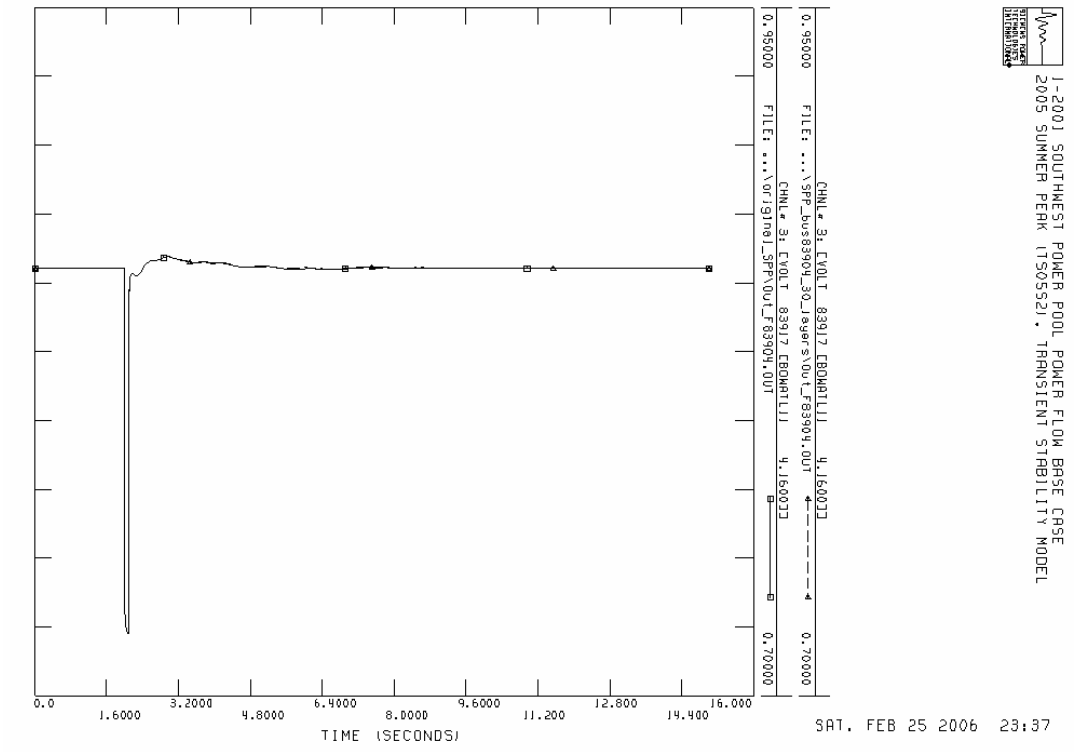

Channel 1: Power bus 82293 (118.05 kV)



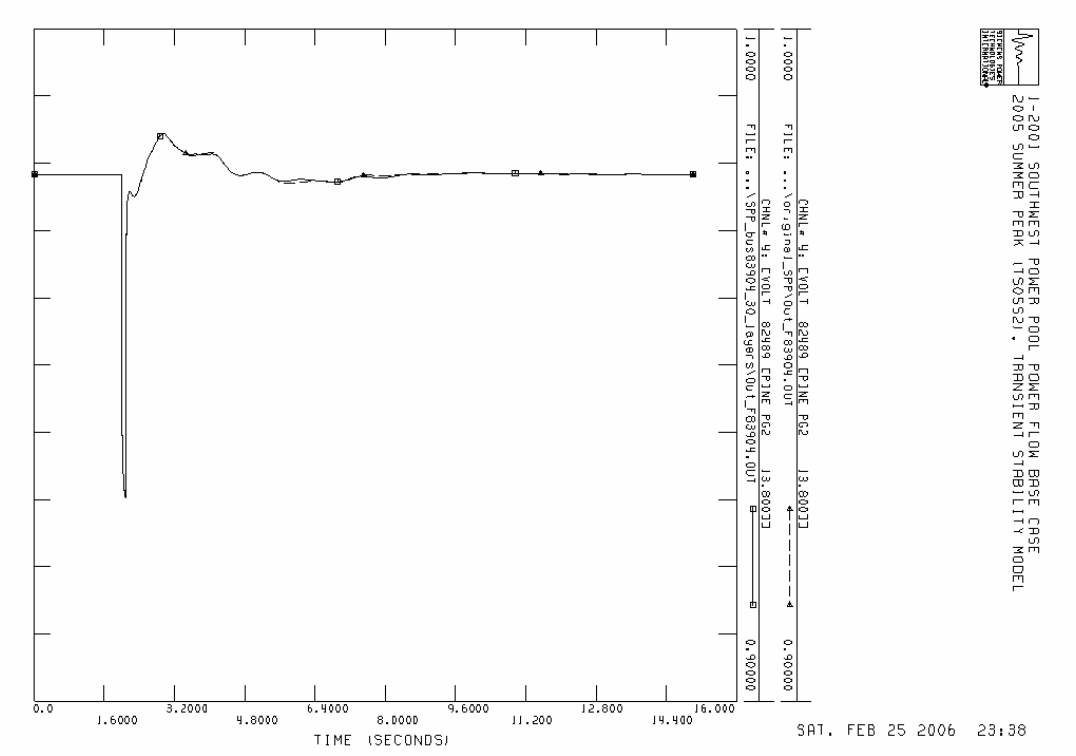
Channel 2: Power bus 82433 (118.05 kV)




Channel 3: Power bus 83917 (4.16 kV)



Channel 4: Power bus 82489 (13.8 kV)



The detail stability simulation step for SPP system shows below.

1_ts05s2x_ori_solve.sav: The system raw data was solved to an accepted level.

2_nerc-all-NO-SPP.dyr, 2_Ts-05SP.dyr: The system dynamic model.

3_PowerFlow_CONLG.py , 3_CONVLOADS-GENS.idv : Use python program to cut the system. Then use idv file to run power flow in dynamic environment.

4_SPPsystem_initial.idv: No fault test to check data integrity.

5_SPPsystem_F54121_87456.idv: Two faults simultaneously happen at load bus 54121 and generator bus 87456.

6_SPPsystem_F64895_83904.idv: Two faults simultaneously happen at load bus 64895 and load bus 83904.

7_SPPsystem_F31426_25044.idv: Two faults simultaneously happen at load bus 31426 and generator bus 25044.

Step 5:

| How many layer of buses near the fault? | | Compare with the original Whole system |
|---|---|---|
| Bus 54121 | Bus 87456 | |
| 5 | 15 | V |
| 8 | 15 | V |
| 15 | 15 | V |
| 30 | 30 | O |

Step 6:

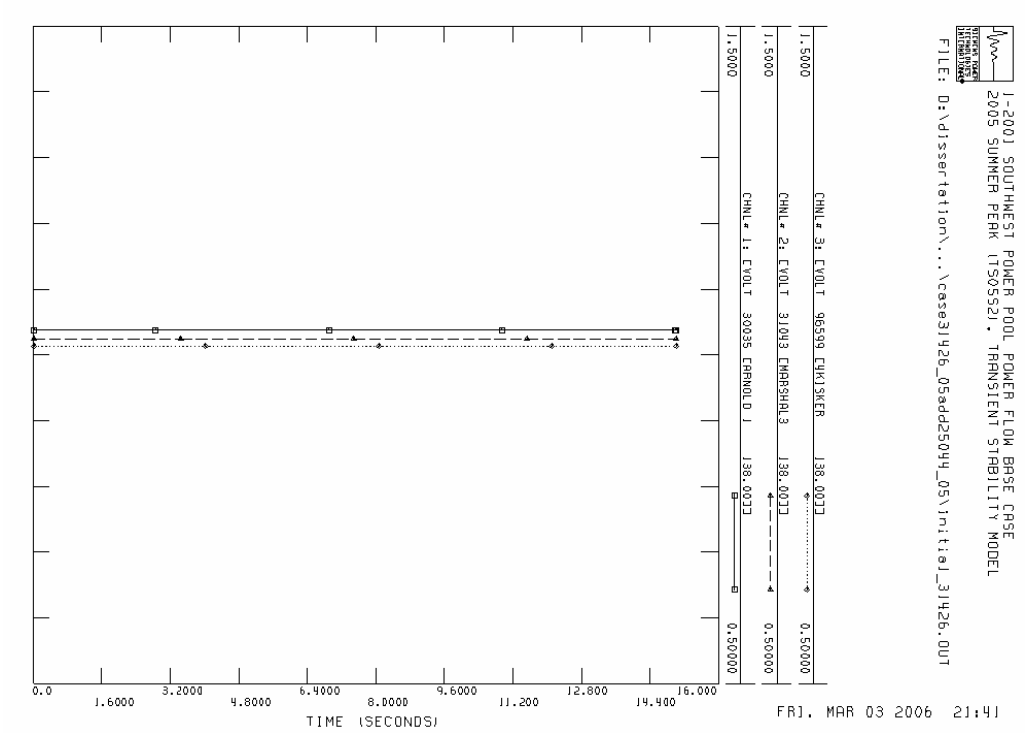
| How many layer of buses near the fault? | | Compare with the original Whole system |
|---|---|---|
| Bus 64895 | Bus 83904 | |
| 5 | 8 | V |
| 8 | 8 | V |
| 15 | 15 | V |
| 30 | 30 | O |

Step 7:

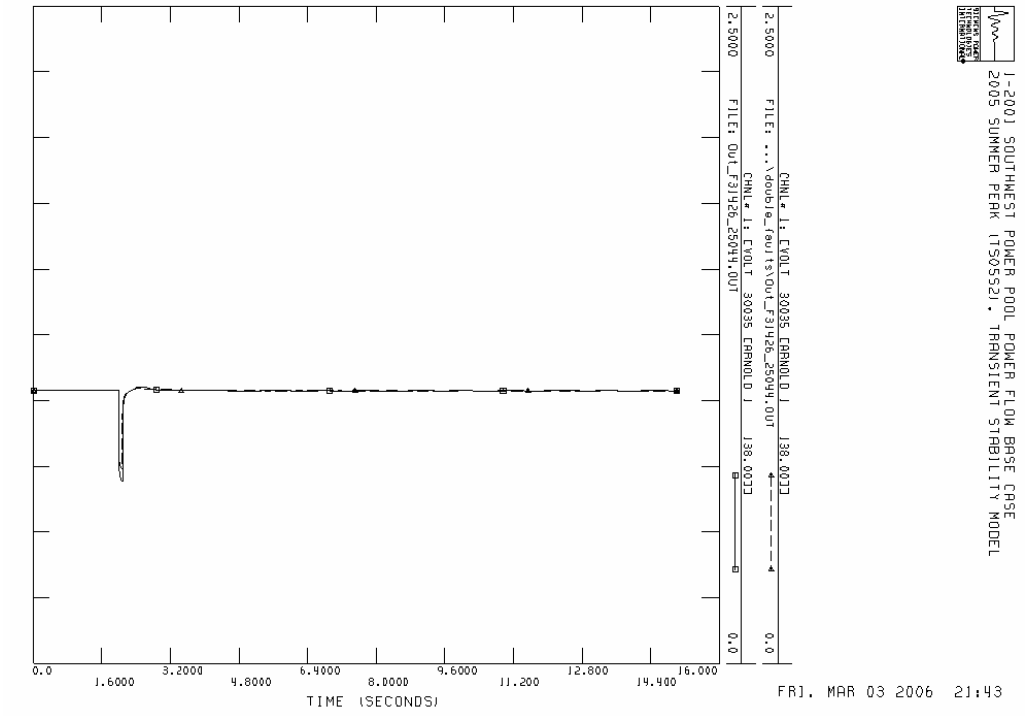
| How many layer of buses near the fault? | | Compare with the original Whole system |
|---|---|---|
| Bus 31426 | Bus 25044 | |
| 5 | 5 | O |
| 8 | 8 | O |
| 15 | 15 | O |
| 30 | 30 | O |

case31426_05add25044_05

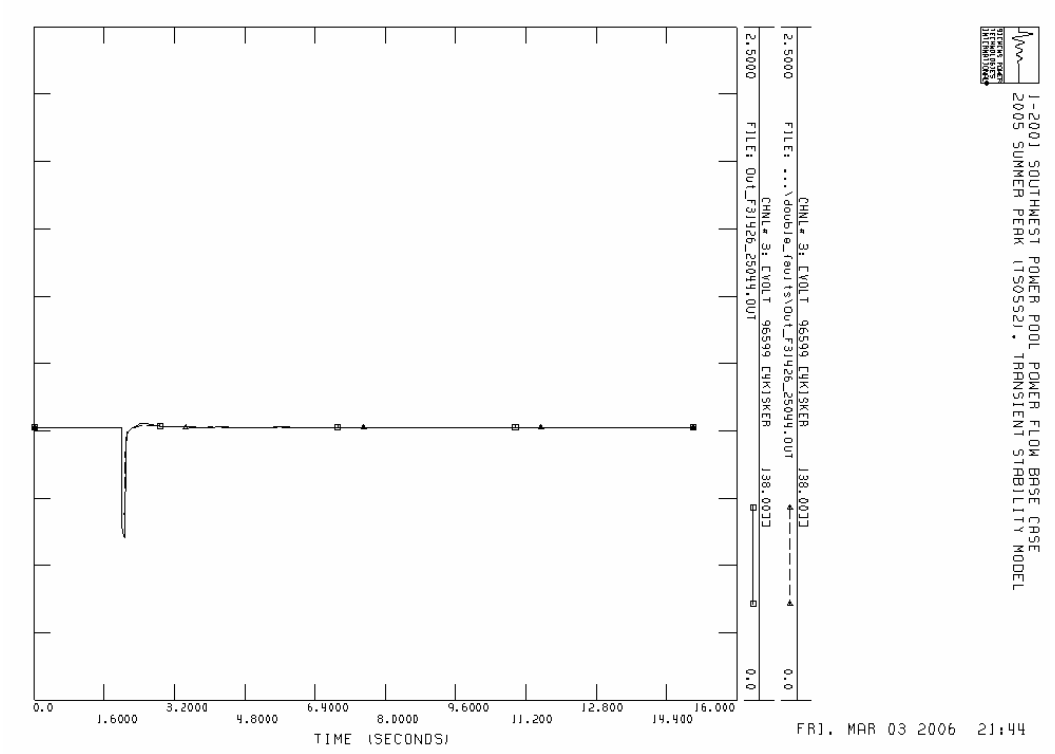
Initial condition for channel 1,2,3



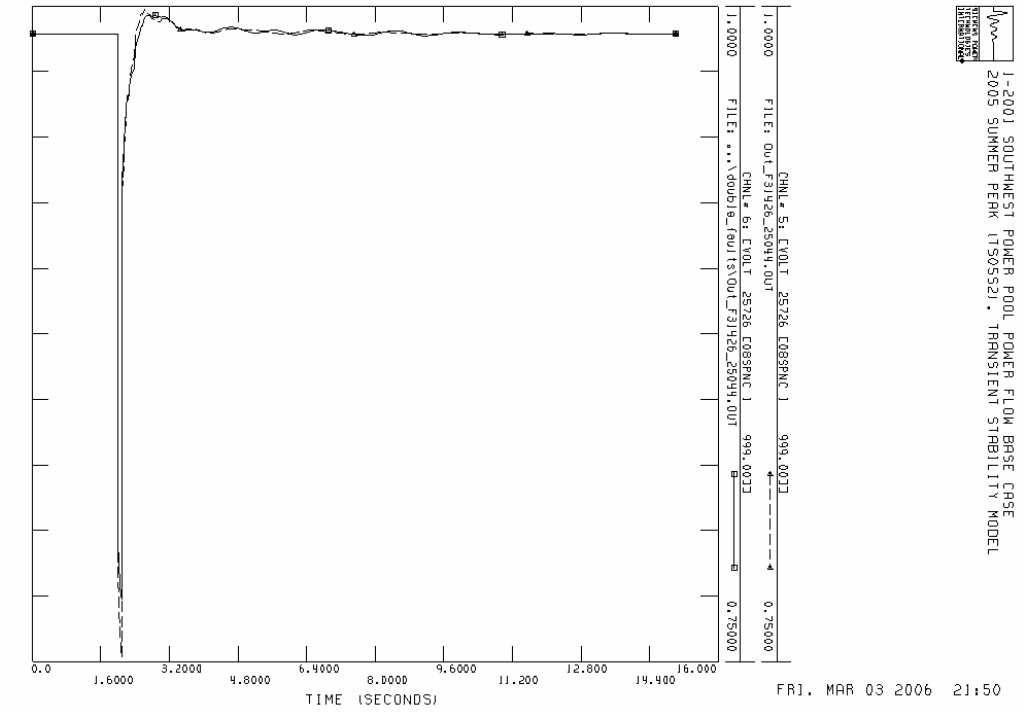

Channel 1: Power bus 30035 (138 kV)



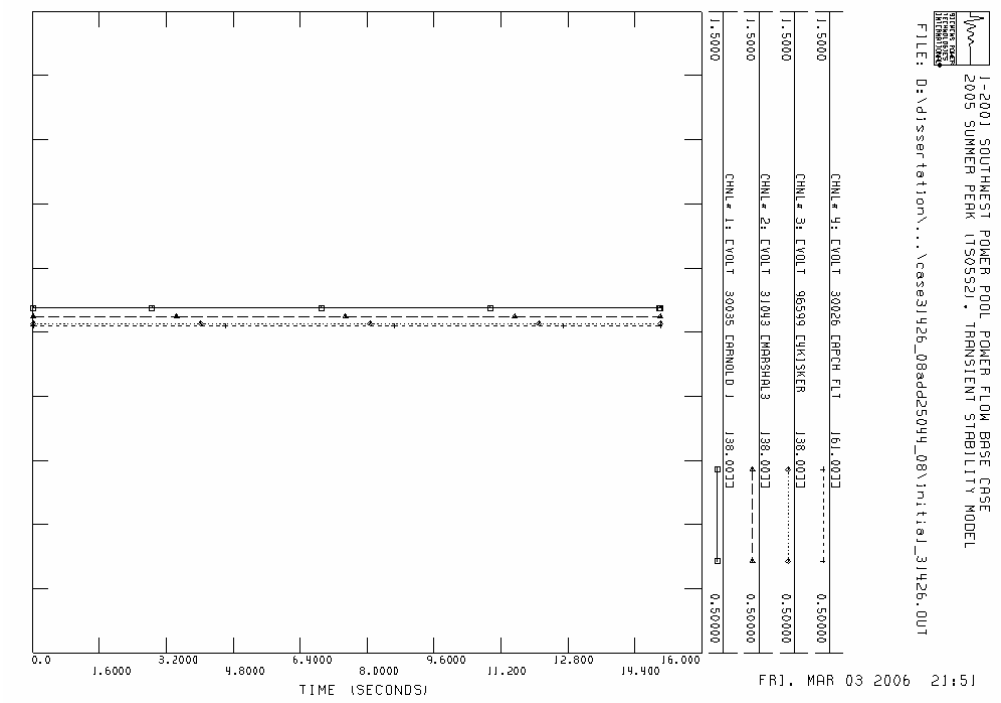
Channel 3: Power bus 96599 (138 kV)



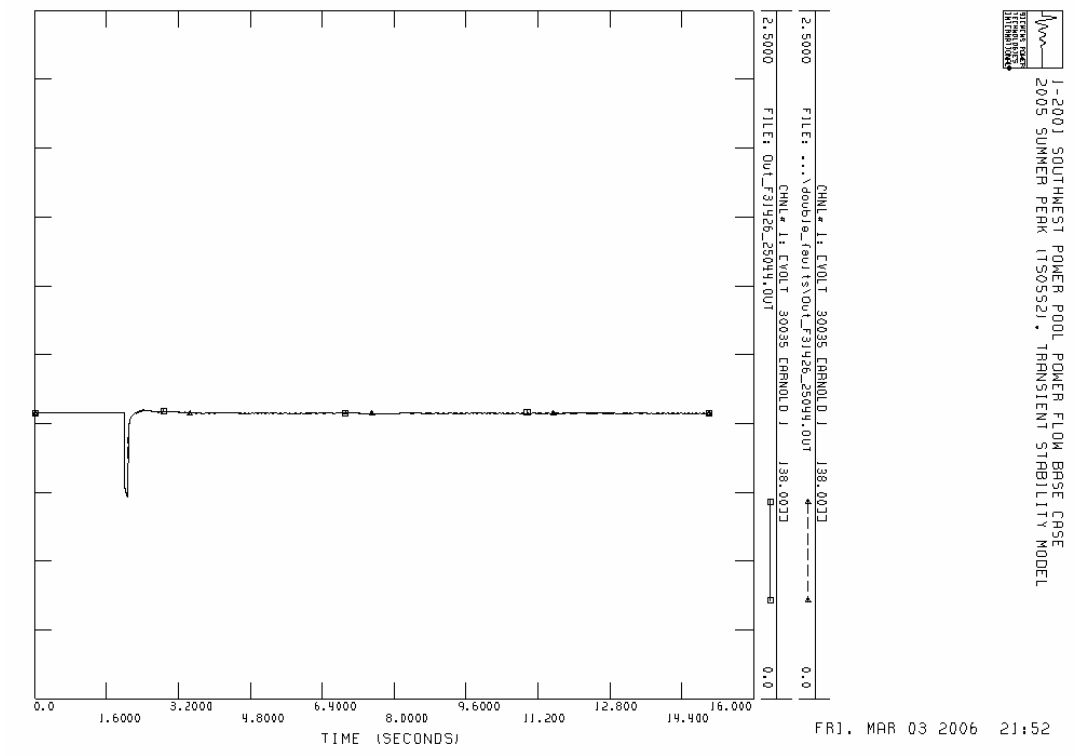

Channel 5: Power bus 25726 (999 kV)



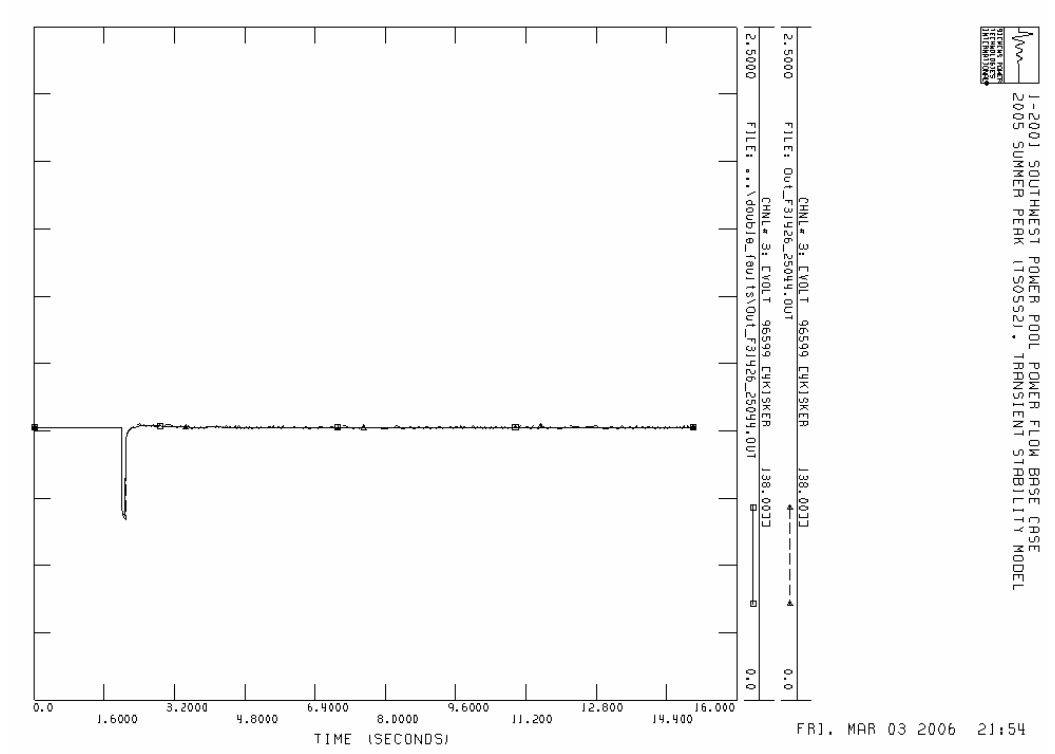
case31426_08add25044_08
Initial condition for channel 1,2,3,4



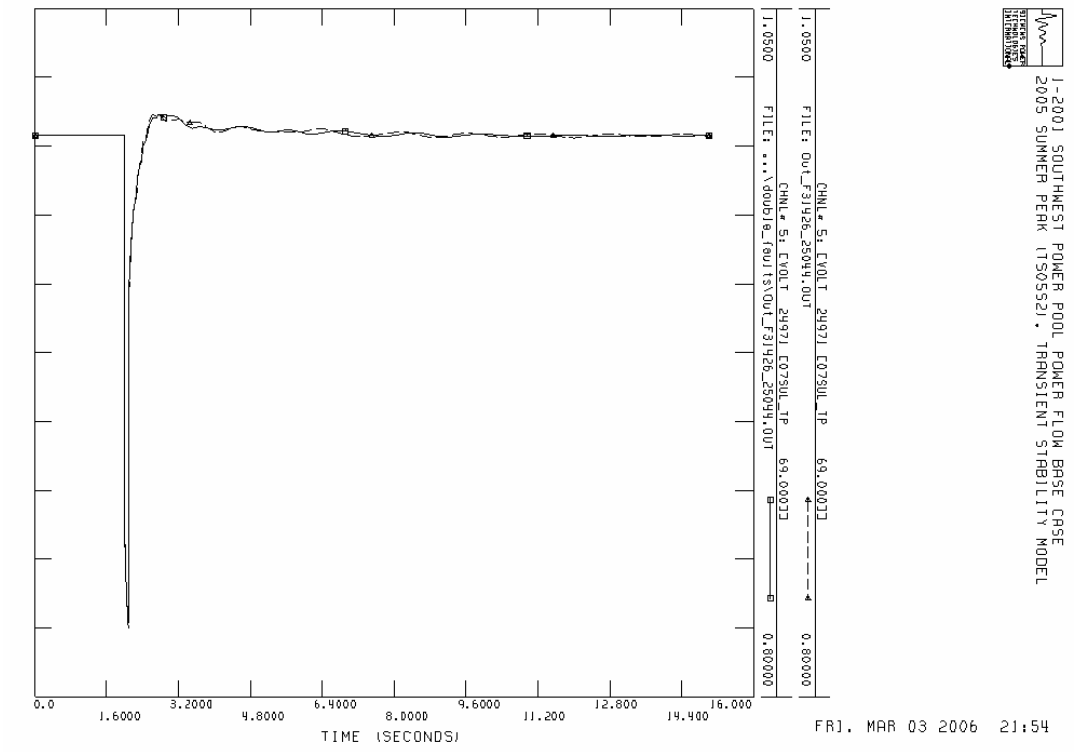

Channel 1: Power bus 30035 (138 kV)



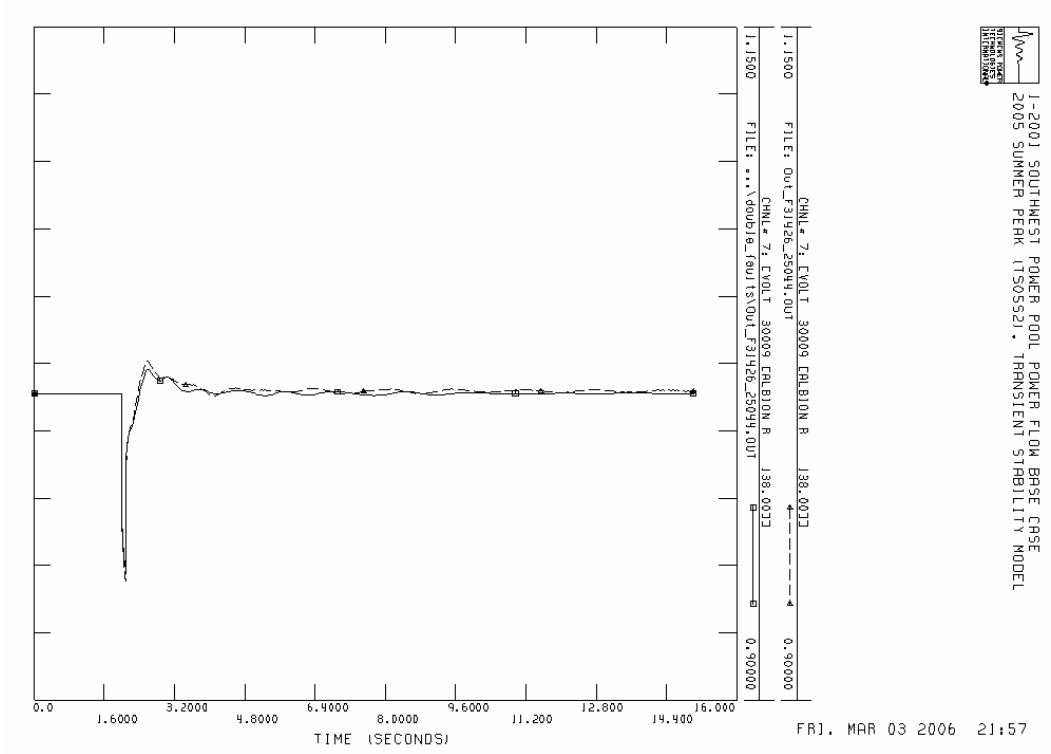
Channel 3: Power bus 96599 (138 kV)



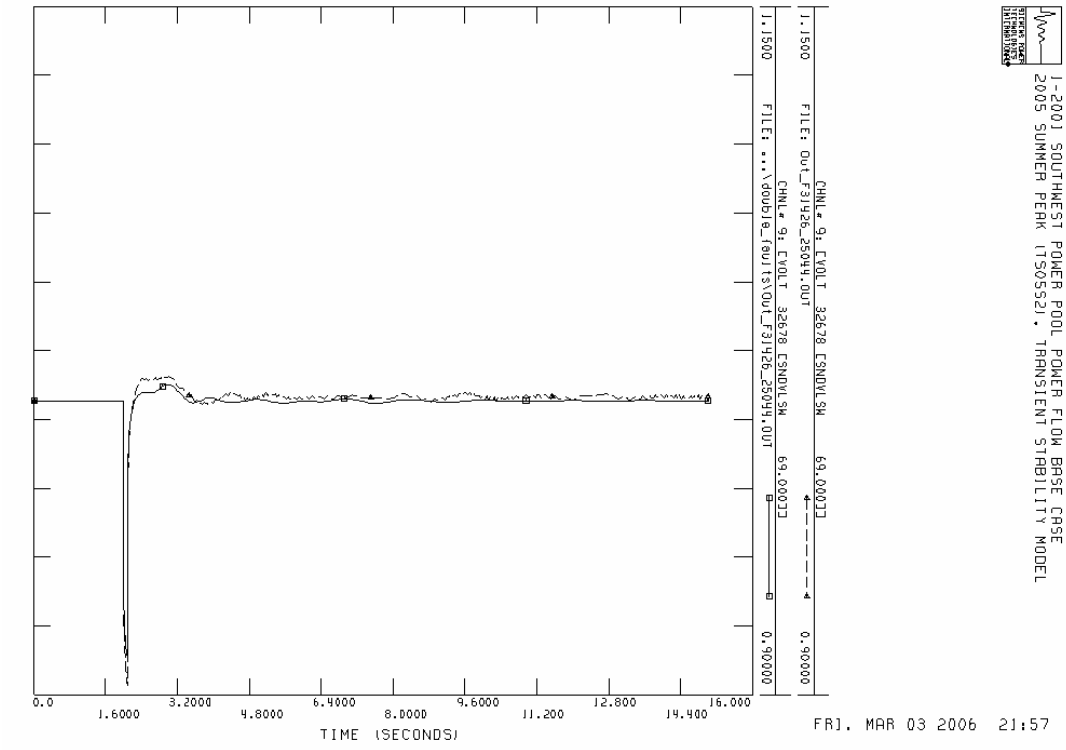

Channel 5: Power bus 24971 (69 kV)



Channel 7: Power bus 30009 (138 kV)

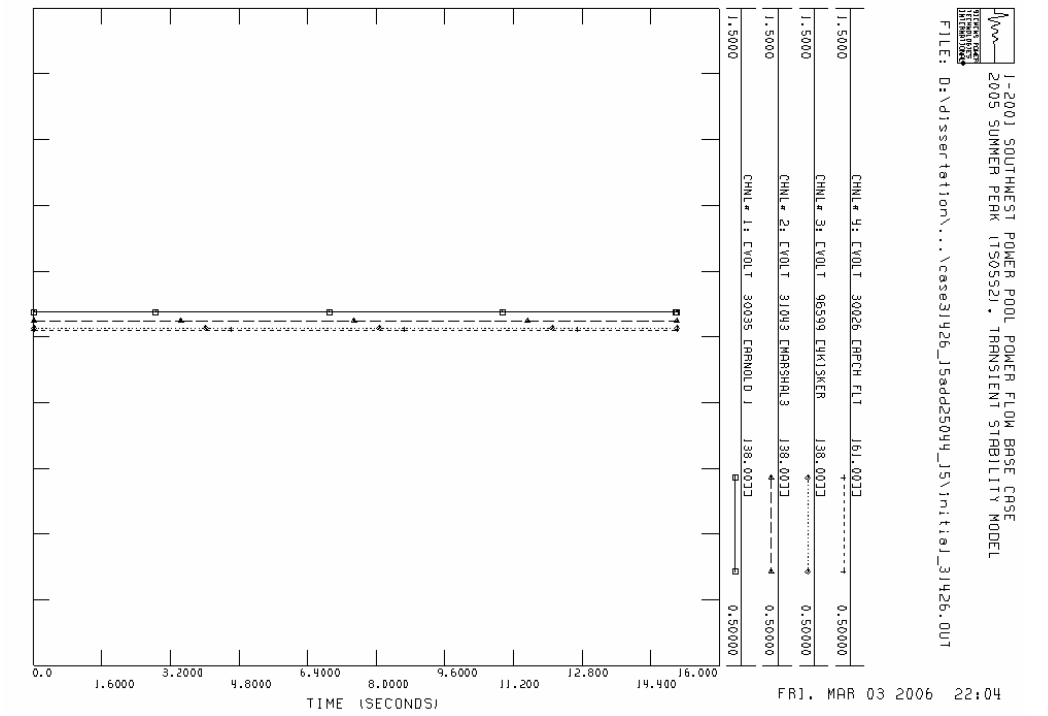



Channel 9: Power bus 32678 (69 kV)

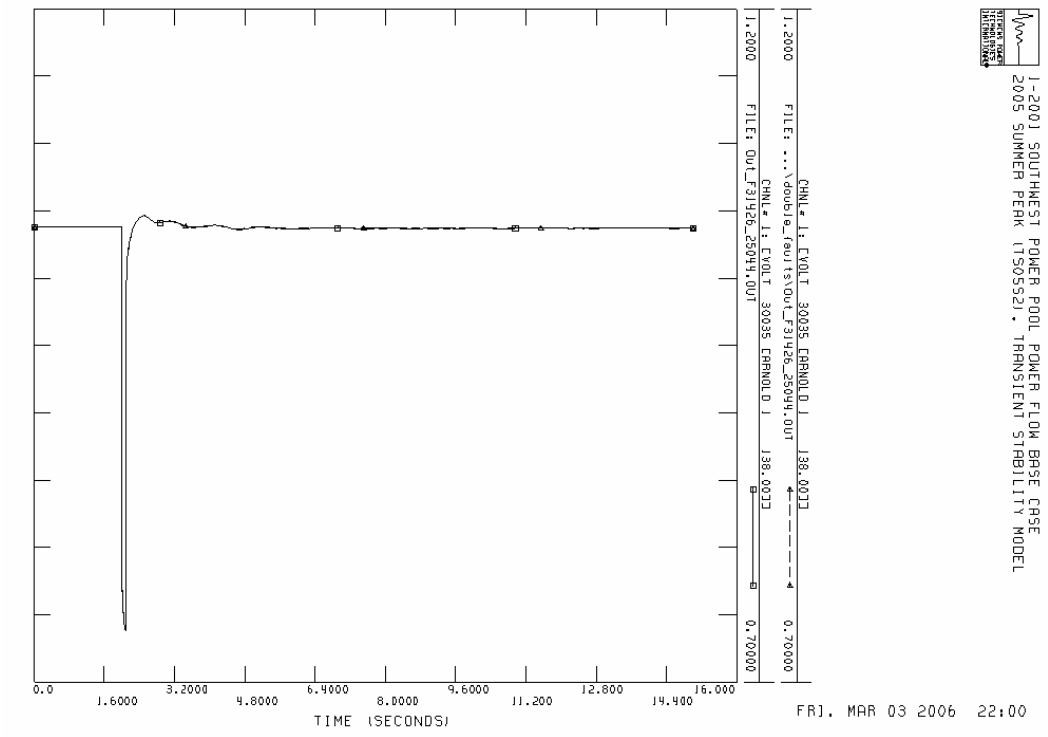


case31426_08add25044_15

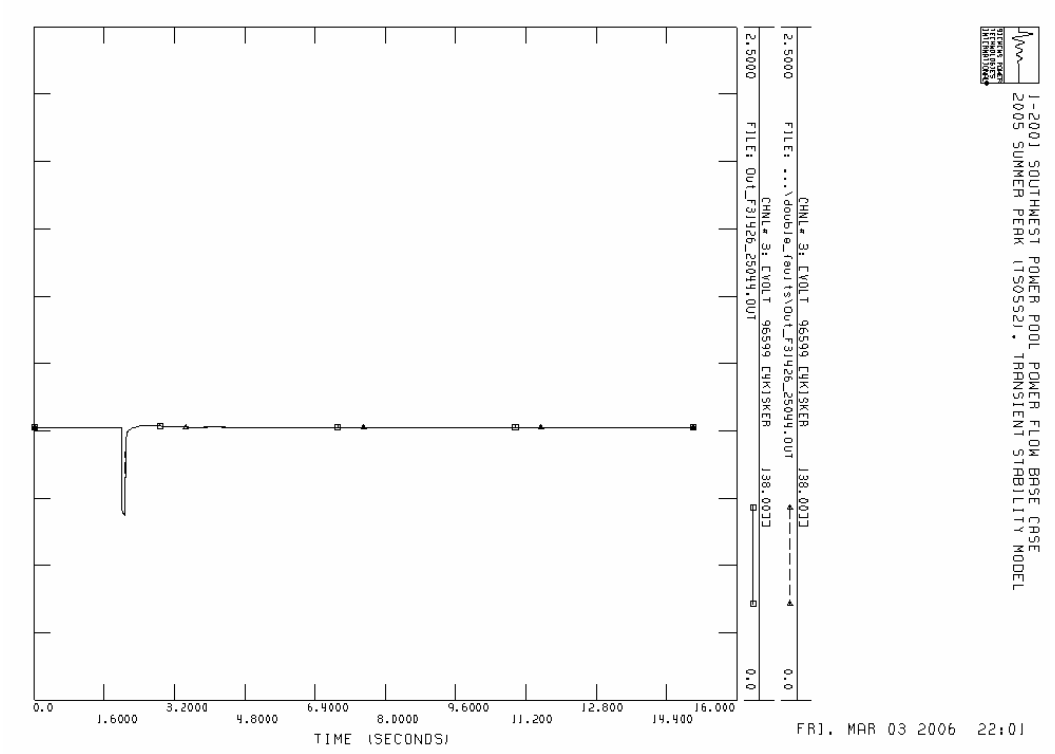
Initial condition for channel 1,2,3,4



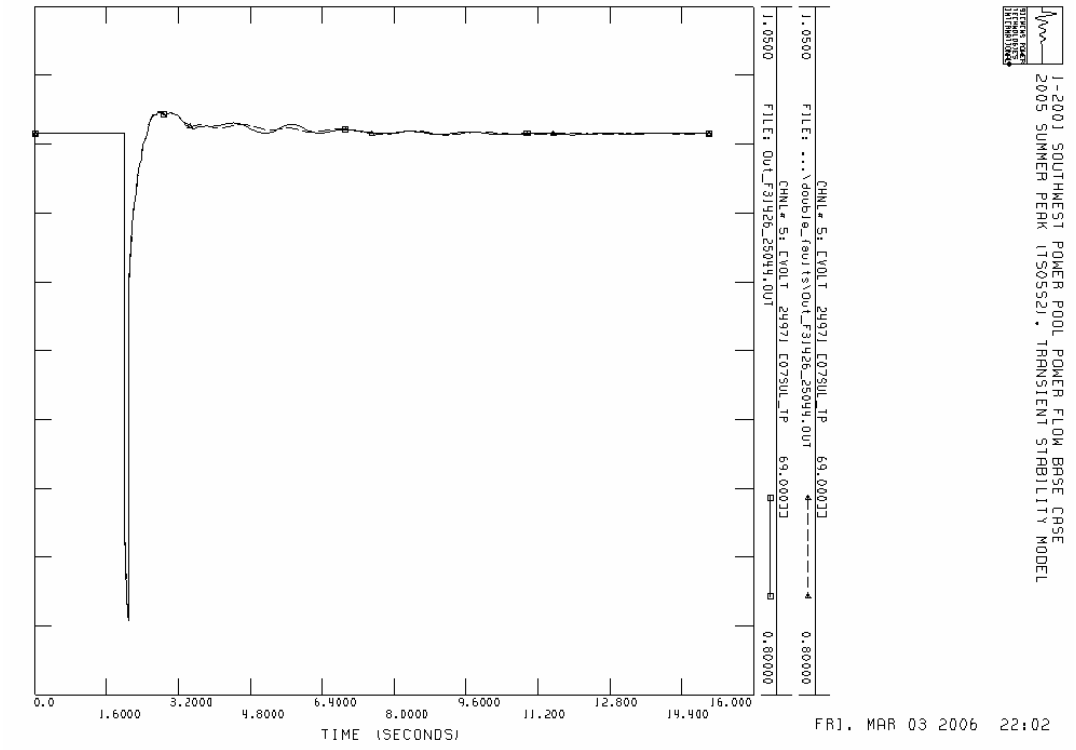

Channel 1: Power bus 30035 (138 kV)



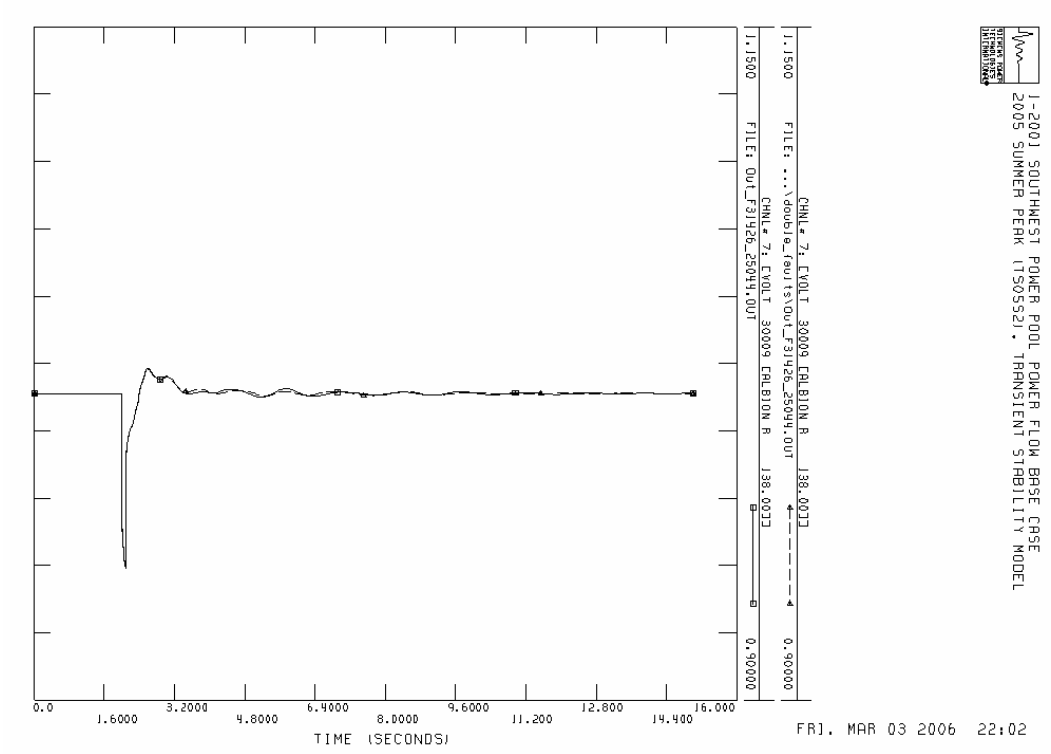
Channel 3: Power bus 96599 (138 kV)



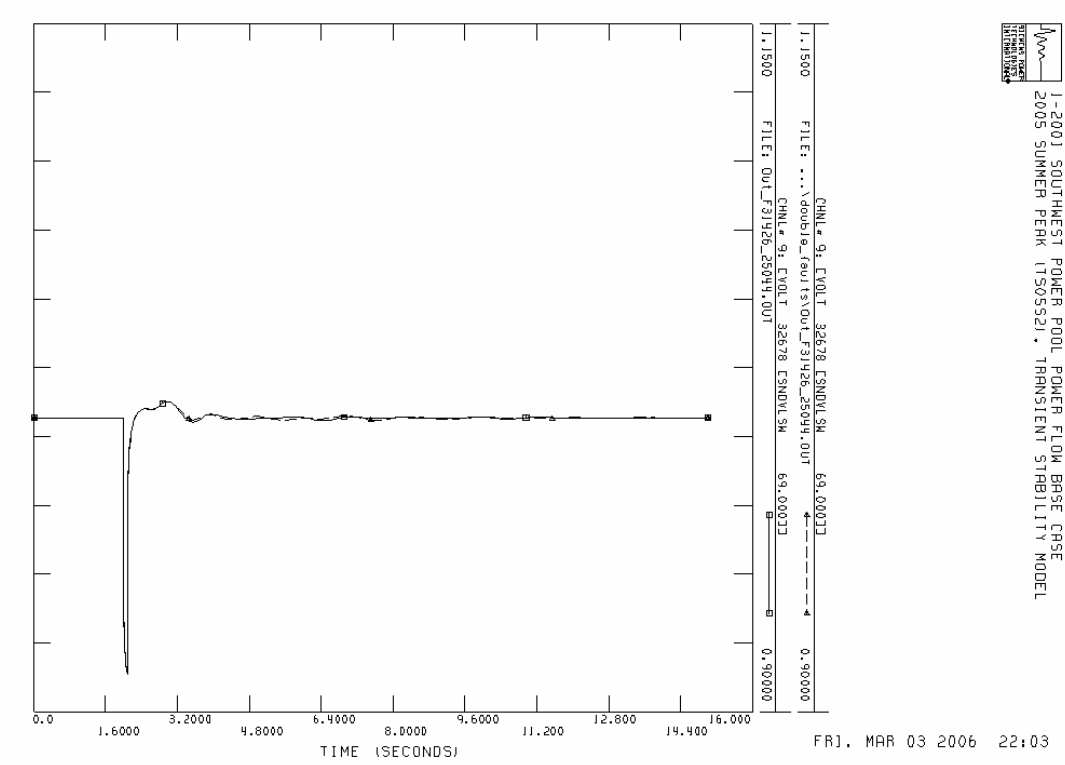

Channel 5: Power bus 24971 (69 kV)



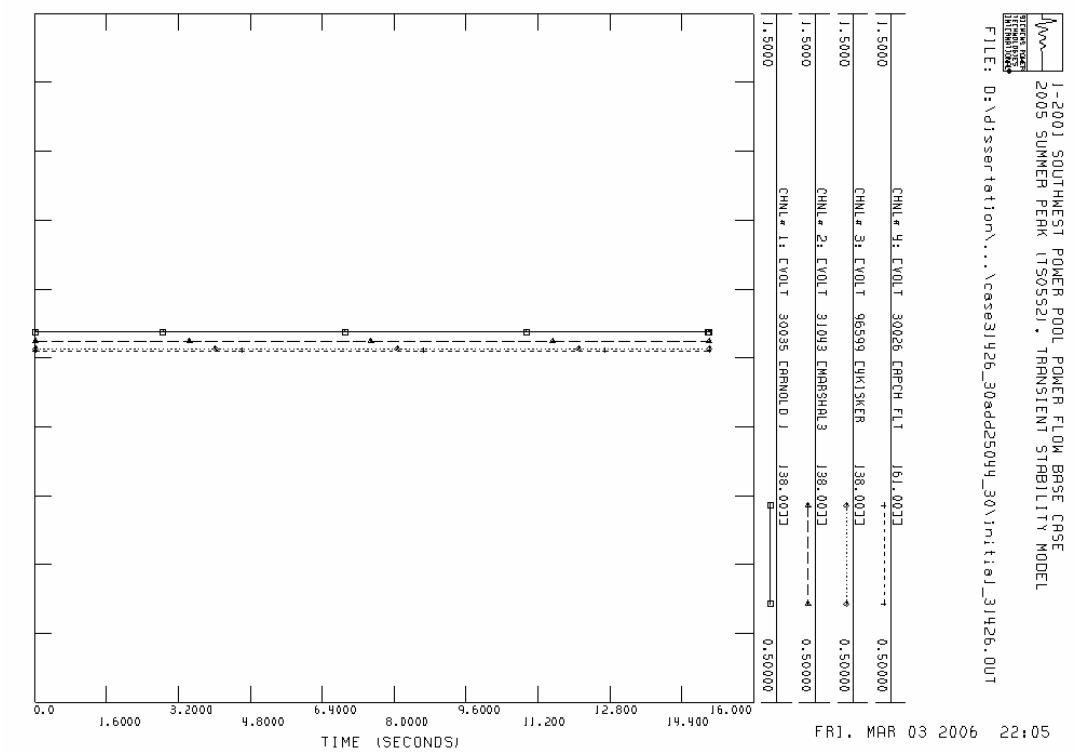
Channel 7: Power bus 30009 (138 kV)



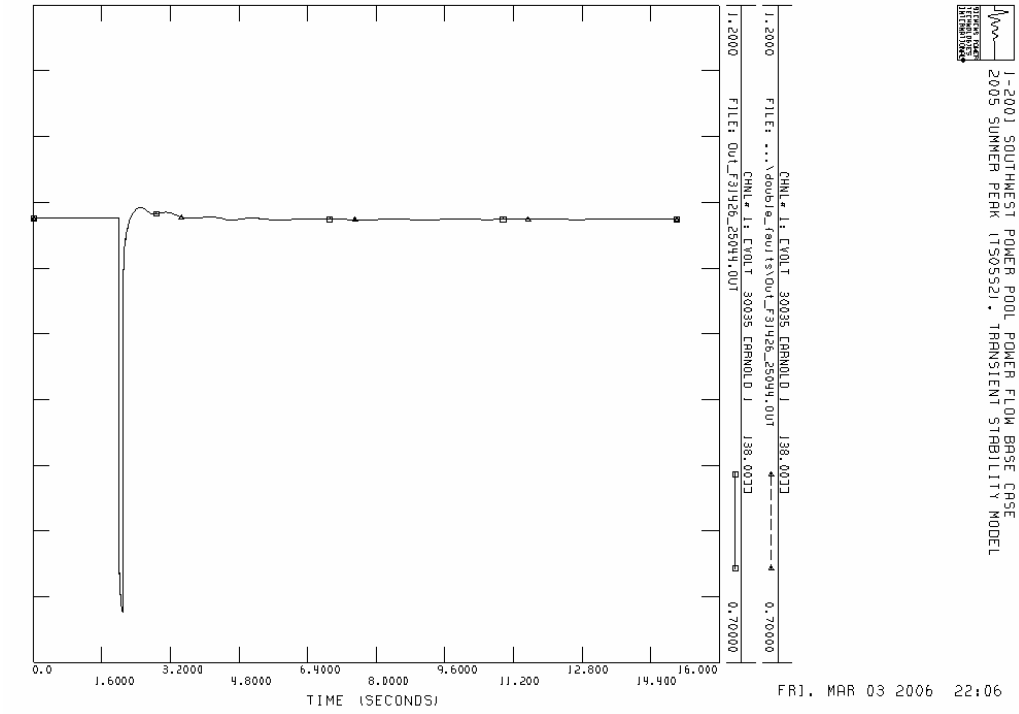

Channel 9: Power bus 32678 (69 kV)



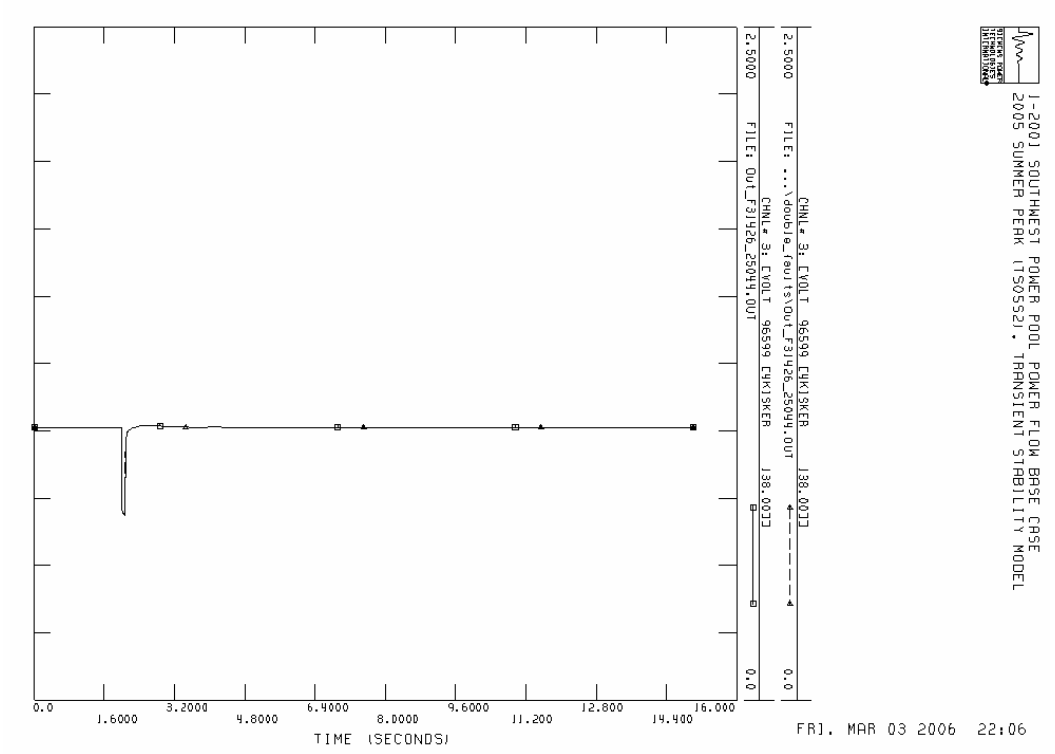
case31426_08add25044_30
Initial condition for channel 1,2,3,4



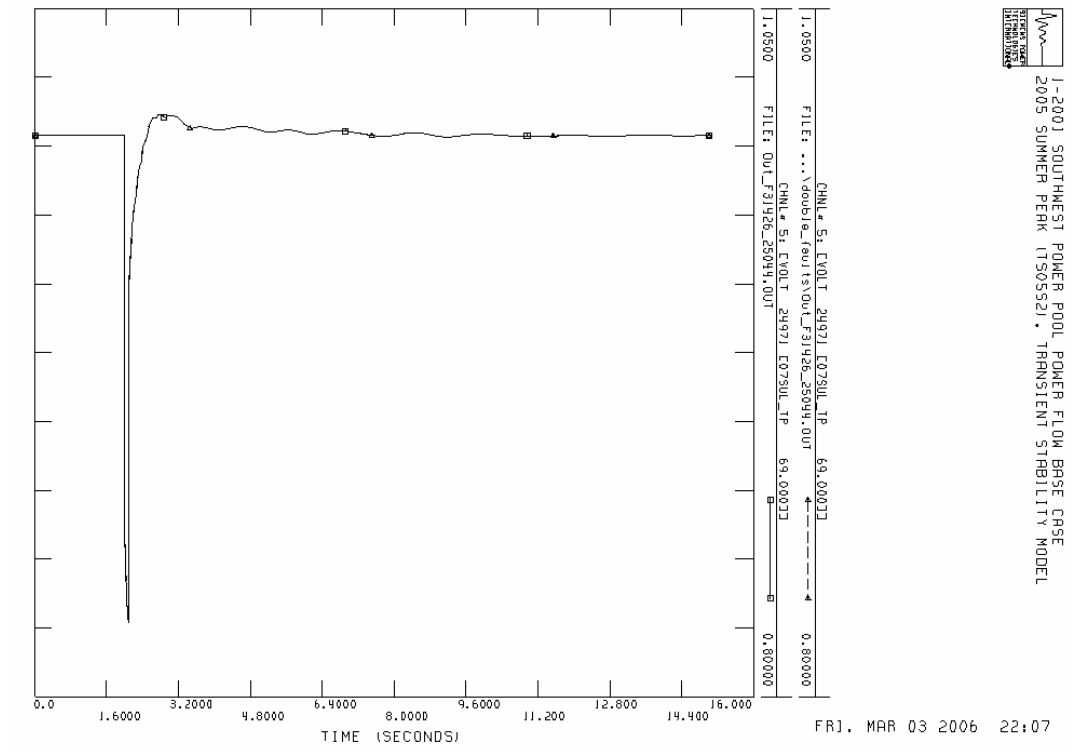

Channel 1: Power bus 30035 (138 kV)



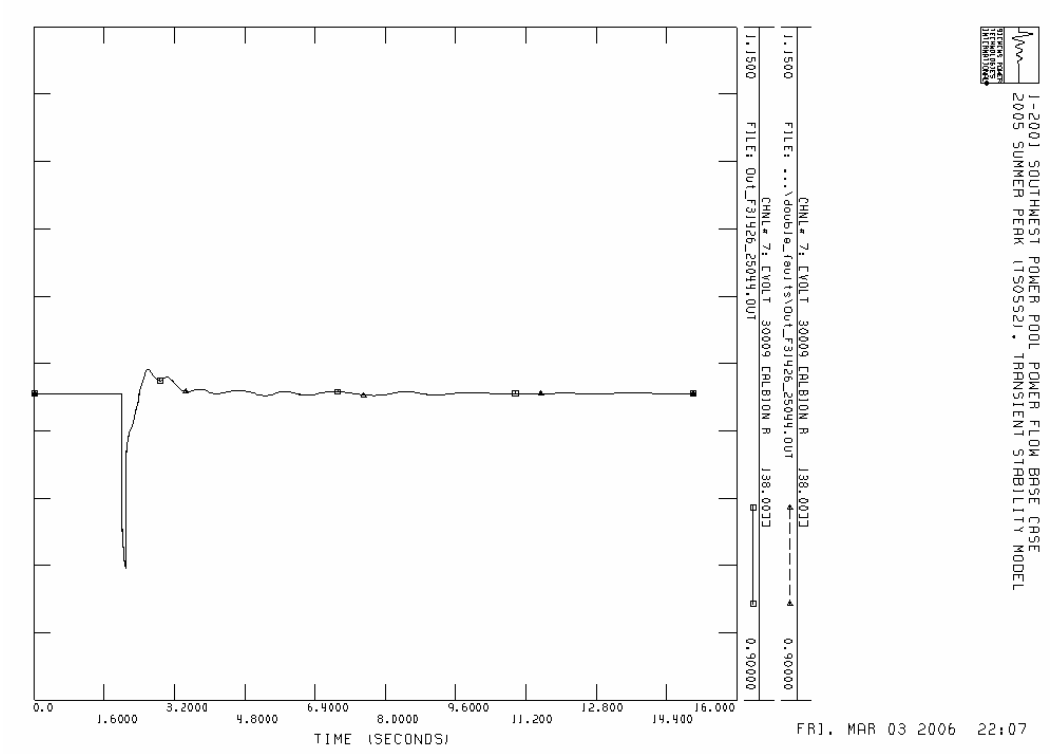
Channel 3: Power bus 96599 (138 kV)



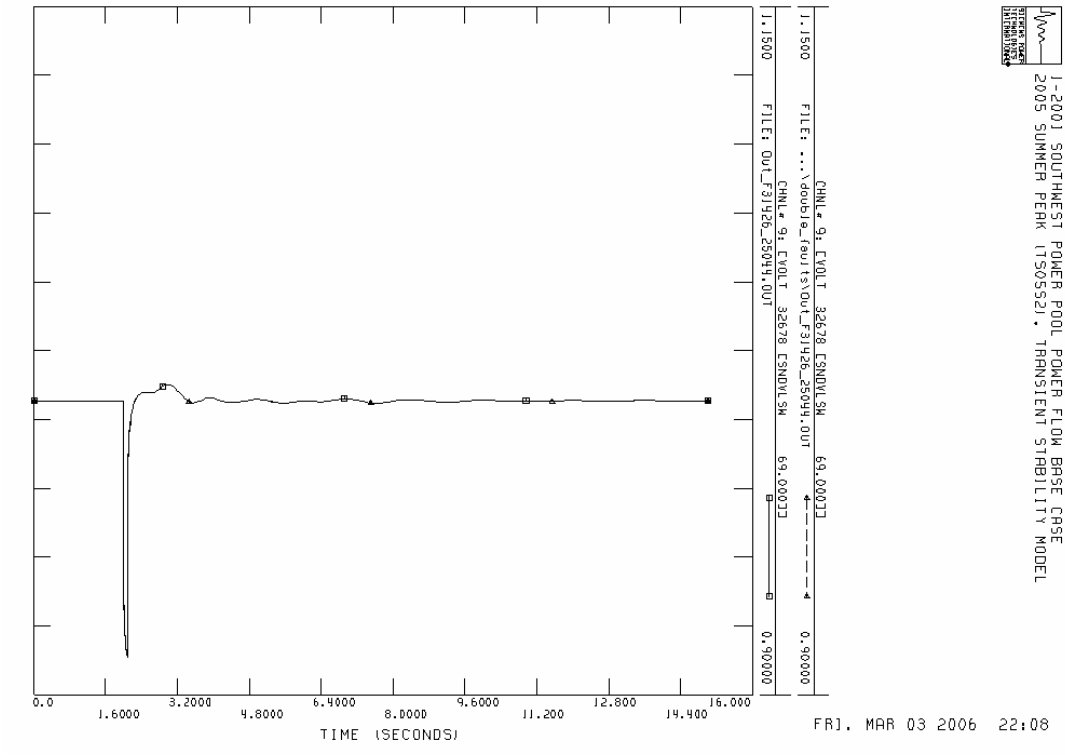

Channel 5: Power bus 24971 (69 kV)



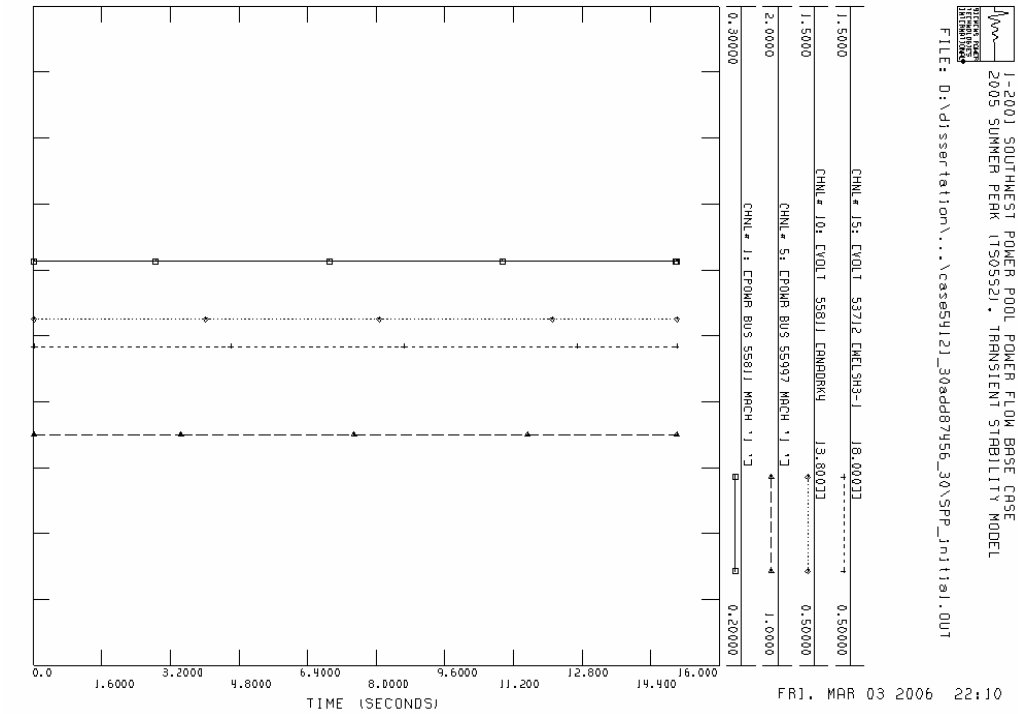
Channel 7: Power bus 30009 (138 kV)



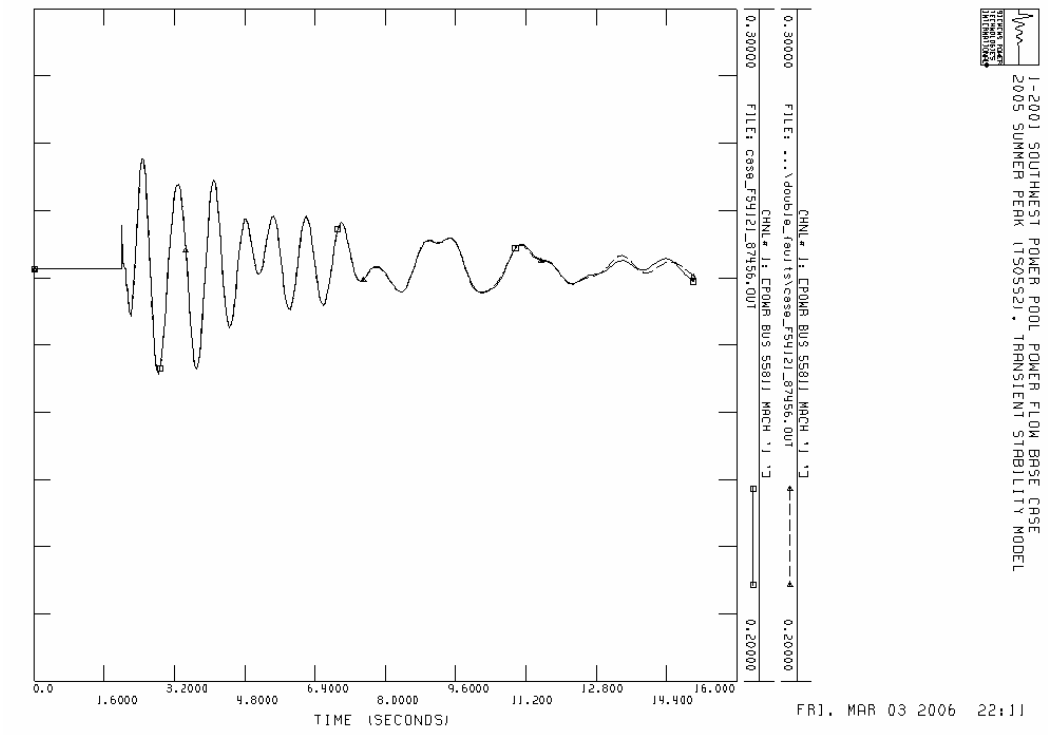

Channel 9: Power bus 32678 (69 kV)



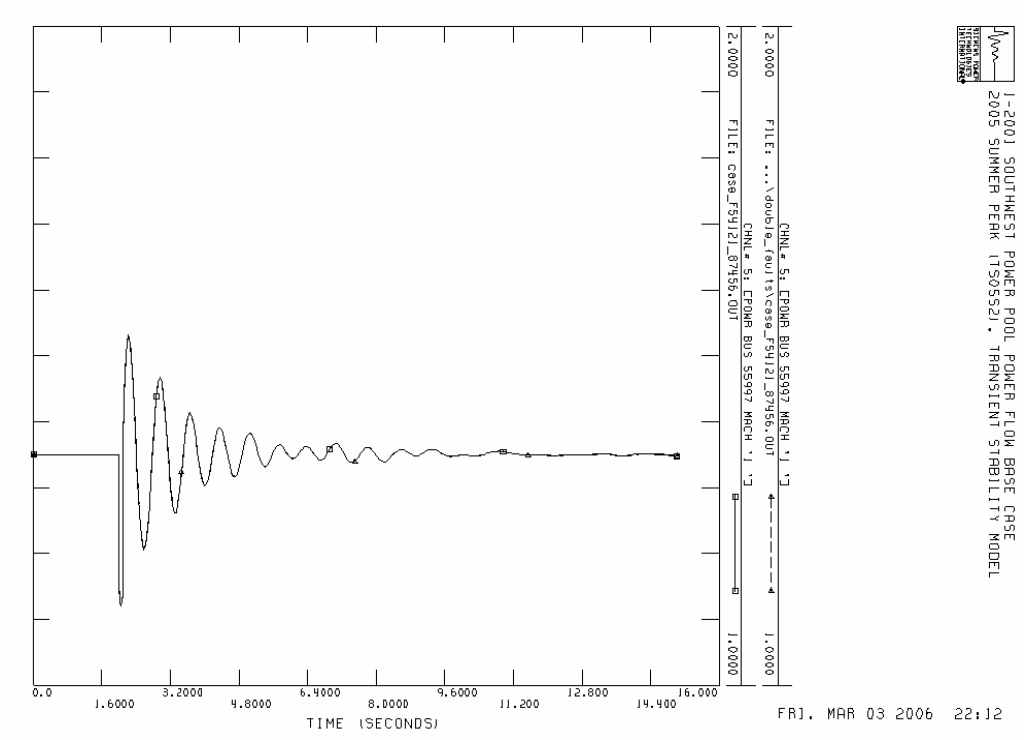
case54121_30add87456_30
Initial condition for channel 1,5,10,15



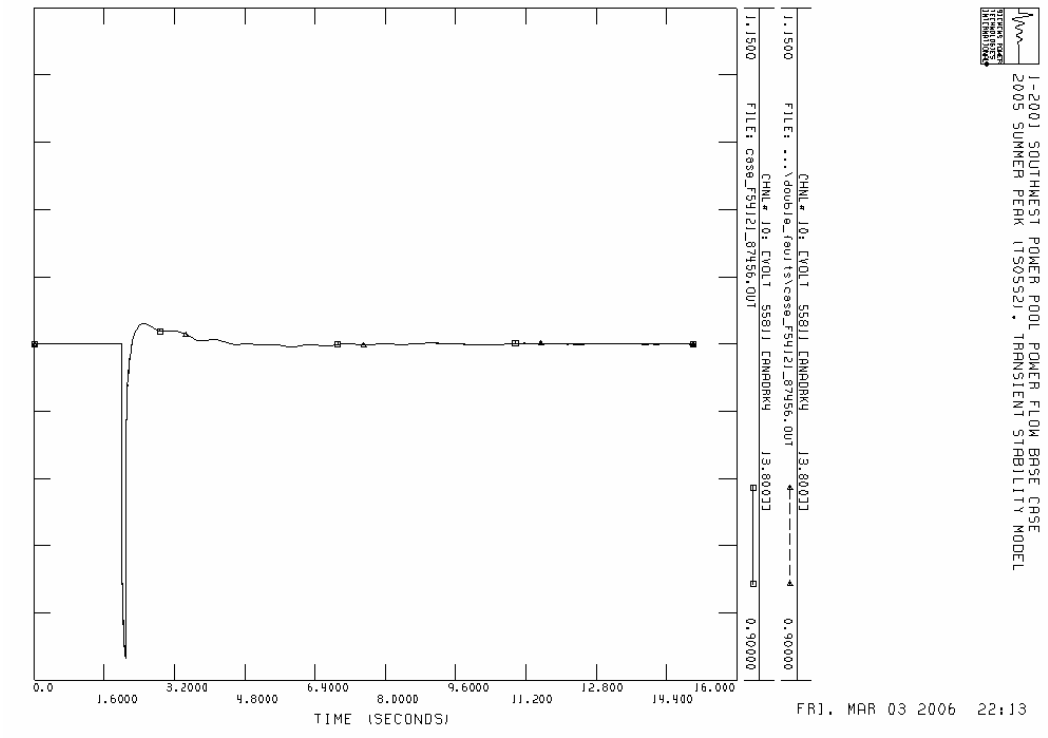

Channel 1: Power bus 55811 MAC ‘1’



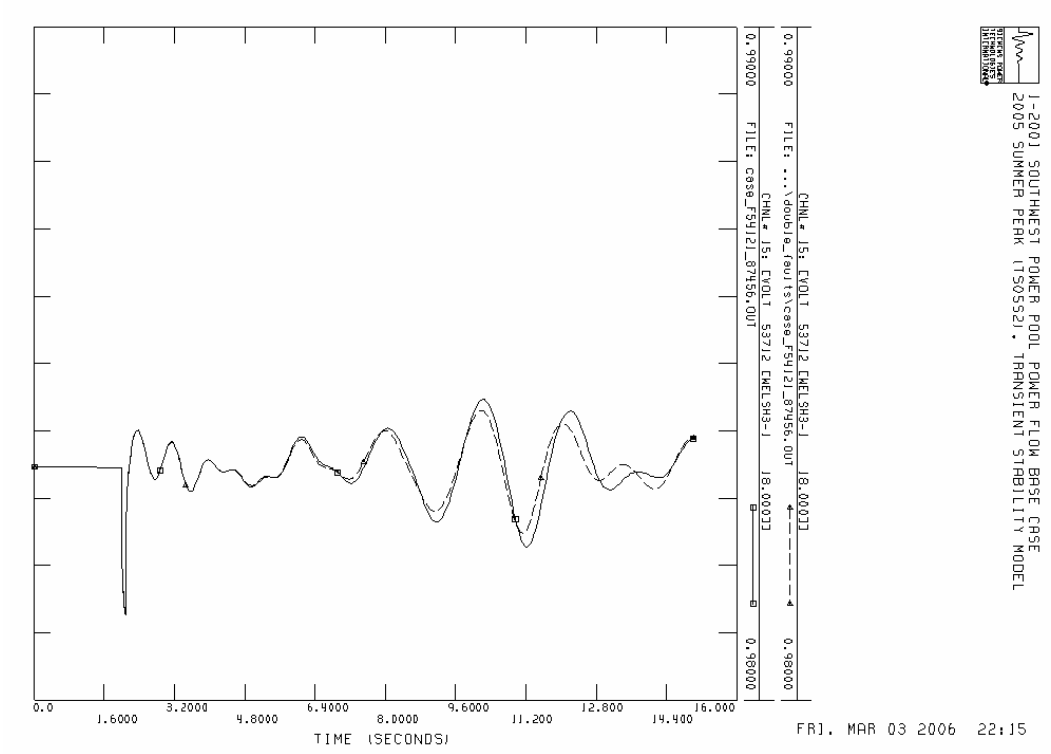
Channel 5: Power bus 55997 MAC ‘1’



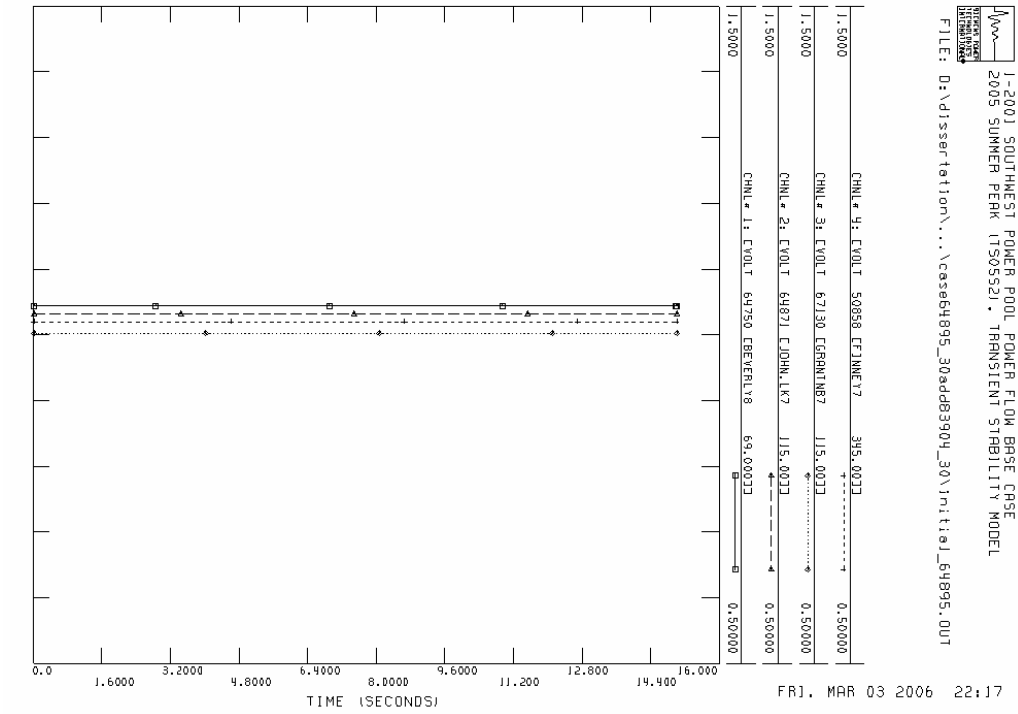

Channel 10: Power bus 55811 (13.8 kV)



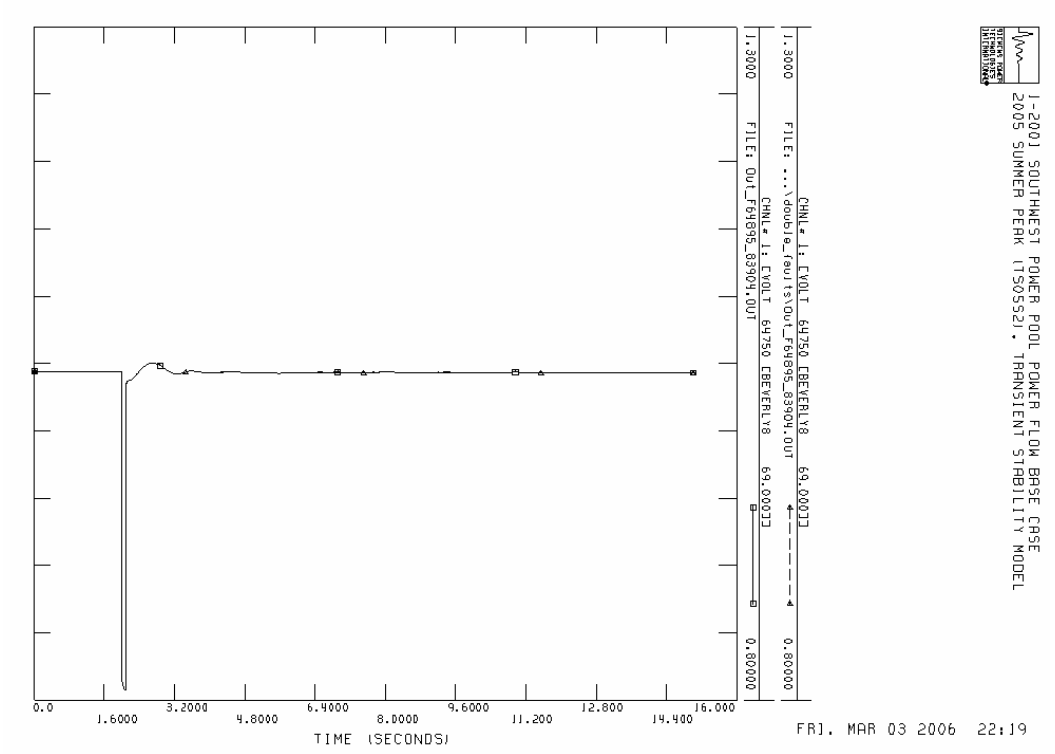
Channel 15: Power bus 53712 (18 kV)



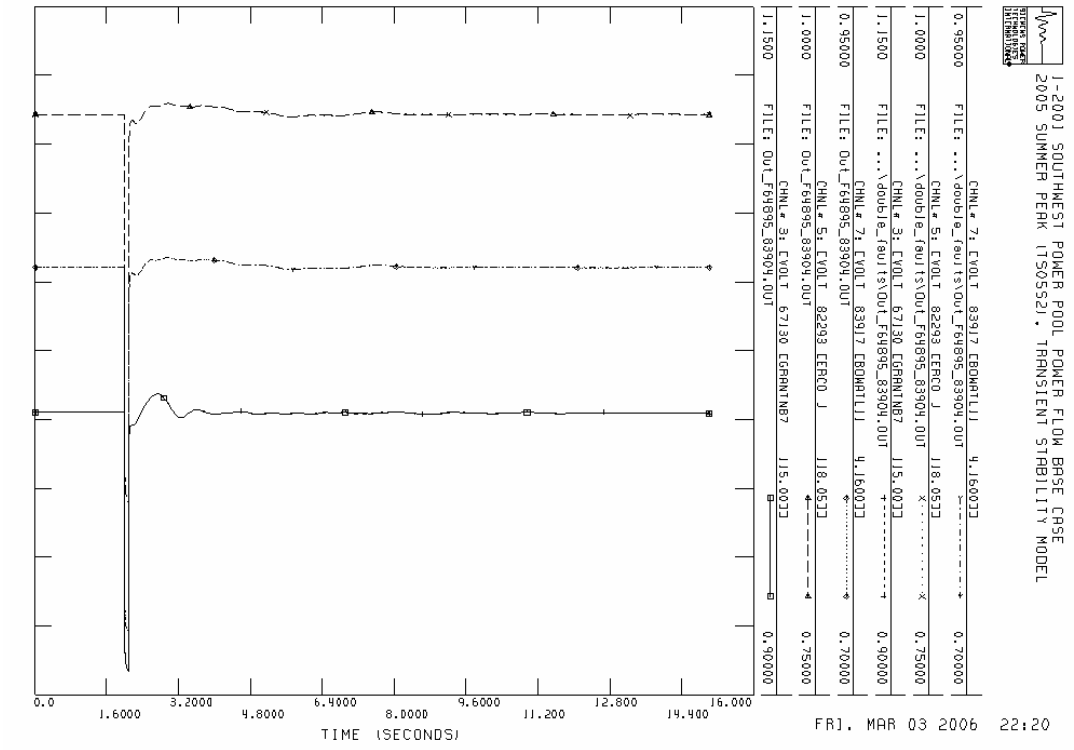

case64895_30add83904_30
Initial condition for channel 1,2,3,4



Channel 1: Power bus 64750 (69 kV)




Channel 3,5,7



REFERENCES


[1] Chitra Yingvivatanapong, "Multi-area Unit Commitment and Economic Dispatch with Market Operation Components," Doctor Dissertation, The University of Texas – Arlington, Dept. EE., May 2006.

[2] North American Electric Reliability Council , Maps, [available online] http://www.nerc.com/regional/NERC_Interconnections_color.jpg

[3] Prabha Kundur, Power System Stability and Control, McGraw Hill (EPRI Power System Engineering Series), Palo Alto, California, 1993.

[4] Northeast Blackout of 1965, [available online] http://en.wikipedia.org/wiki/Northeast_Blackout_of_1965

[5] Northeast Blackout of 2003, [available online] http://en.wikipedia.org/wiki/2003_North_America_blackout

[6] Eduardo Alberto Perez Mancia, "Parametric Study to Evaluate the Transient Stability of an Industrial Power System," Master Thesis, The University of Texas – Arlington, Dept. EE., May 2005.

[7] Shu, J.; Wei Xue; Weimin Zheng, "A parallel transient stability simulation for power systems," IEEE Trans. Power Systems, Volume 20, Issue 4, pp. 1709 – 1717, Nov. 2005.

[8] Fang Hualiang; Mao Chengxiong; Zhang Buhan; Lu Jiming, "The Grid Computing Model—A New Computing Model For The Analysis and Computation Of Large-Scale Power System," Transmission and Distribution Conference, pp. 1 – 6, Aug. 2005

[9] Galarza, R.J.; Chow, J.H.; Price, W.W.; Hargrave, A.W.; Hirsch, P.M., "Aggregation of exciter models for constructing power system dynamic equivalents, " IEEE Trans. Power Systems, on Volume 13, Issue 3, pp. 782 – 788, Aug. 1998

[10] NERC Pre-1/1/2006 , Maps, [available online] http://www.epis.com/EnergyLinks/Reliability%20Regions/reliability_regions.htm

[11] Diestel, Reinhard., Graph Theory, 2nd ed., New York Springer-Verlag, New York, 2000. p 24.

[12] Demmel, J. "CS 267: Notes for Lecture 23, April 9, 1999. Graph Partitioning, Part 2." [available online] http://www.cs.berkeley.edu/~demmel/cs267/lecture20/lecture20.html

[13] Chung, F. R. K., Spectral Graph Theory, Providence, RI: Amer. Math. Soc., 1997.

[14] Chris Godsil, Gordon Royle, Algebraic Graph Theory, New York Springer-Verlag, New York, 2001.

[15] M. Fiedler, "Algebraic Connectivity of Graphs", Czechoslovak Mathematical Journal, 23, pp. 298-305, 1973.

[16] M. Fiedler, "A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory", Czechoslovak Mathematical Journal, 25, pp. 619--633, 1975.

[17] Derek Greene, Graph Partitioning and Spectral Clustering. [available online] https://www.cs.tcd.ie/research_groups/mlg/kdp/presentations/Greene_MLG04.ppt

[18] PSS/E-30.1 Users Manual, Siemens, April 2005

[19] J. D. Glover, and M. S. Sarma, Power System Analysis and Design, 3nd ed., Brooks/Cole, California, 2002, pp. 285-286.

[20] C.L. DeMarco, T. Yong, J. Meng, and F.L. Alvarado, "Efficient Computation of Higher Order Derivatives of Power Flow and Line Flow Quantities, " 13th Power System Computation Conference, 1998.

[21] Supun Tiptipakorn, "A Spectral Bisection Partitioning Method for Electric Power Network Applications," Master thesis, The University of Wisconsin – Madison, Dept. ECE., December 2001

[22] M. Fiedler, "Algebraic Connectivity of Graphs", Czechoslovak Mathematical Journal, 23, pp. 298-305, 1973

[23] Demmel, J. "CS 267: Notes for Lecture 23, April 9, 1999. Graph Partitioning, Part 2." Available: http://www.cs.berkeley.edu/~demmel/cs267/lecture20/lecture20.html

[24] M. Fiedler, "A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory", Czechoslovak Mathematical Journal, 25, pp. 619--633, 1975

[25] Kadoya, T.; Sasaki, T.; Ihara, S.; Larose, E.; Sanford, M.; Graham, A.K.; Stephens, C.A.; Eubanks, C.K., " Utilizing system dynamics modeling to examine impact of deregulation on generation capacity growth", Proceedings of the IEEE, Volume 93, Issue 11, pp. 2060 - 2069, Nov. 2005

[26] Chai Chompoo-inwai, "The Operation Platform Recommendation for the Deregulated Utility Industry In Thailand," Doctor dissertation, The University of Texas at Arlington, Dept. EE., July 2005

[27] Eduardo Alberto Perez Mancia, "Parametric Study to Evaluate the Transient Stability of an Industrial Power System," Master thesis, The University of Texas at Arlington, Dept. EE., May 2005

[28] Chai Chompoo-inwai, "The Operation Platform Recommendation for the Deregulated Utility Industry in Thailand," Doctor Dissertation, The University of Texas – Arlington, Dept. EE., July 2005.

[29] W.W.; Hargrave, A.W.; Hurysz, B.J.; Chow, J.H.; Hirsch, P.M, "Large-scale system testing of a power system dynamic equivalencing program, " IEEE Trans. Power Systems, on Volume 13, Issue 3, pp. 768 – 774, Aug. 1998

[30] Energy Systems Research Center, The University of Texas at Arlington, "STABILITY STUDIES FOR THE EXXONMOBIL BEAUMONT PLANT", Final Report, September 2004

[31] Sterian, A., PyMat software version 6-1.0.4, [available online] http://claymore.engineer.gvsu.edu/~steriana/Python/

[32] Albert Strasheim, Paul Barrett and the other 19 persons, Numerical Python software, [available online] http://sourceforge.net/project/showfiles.php?group_id=1369&package_id=1351

[33] Andrew Sterian, PyMat - An interface between Python and MATLAB, July 9, 1999, [available online] http://claymore.engineer.gvsu.edu/~steriana/Python/pymat.html

[34] Andrew Sterian, Pymat version 1.1.90, [available online] http://rpm.pbone.net/index.php3/stat/4/idpl/1129673/com/pymat-1.1.90-1.i386.rpm.html

[35] PSS/E-30.1 Users Manual, Siemens, April 2005

[36] PSSECHOP, PSS/E Channel Output Processor, Siemens, [available online] http://www.pti-us.com/pti/software/psse/user_support.cfm

[37] Bruce Hendrickson, Robert Lelandy, The Chaco Version 2.0, Sandia National Laboratories [available online] http://www.cs.sandia.gov/~bahendr/chaco.html

[38] George Karypis, Vipin Kumar, The METIS Version 4.0, University of Minnesota, Department of Computer Science / Army HPC Research Center [available online] http://glaros.dtc.umn.edu/gkhome/views/metis

[39] Chris Walshaw, JOSTLE Version 3.1, The University of Greenwich, [available online] http://claymore.engineer.gvsu.edu/~steriana/Python/pymat.html

[40] Robert Preis, PARTY Version 1.1, 16.9.1996, University of Paderborn [available online] http://wwwcs.uni-paderborn.de/fachbereich/AG/monien/RESEARCH/PART/party.html

[41] François Pellegrini, SCOTCH Version 4.0, Laboratoire Bordelais de Recherche en Informatique, [available online] http://www.labri.fr/perso/pelegrin/scotch/

[42] North American Electric Reliability Council , About NERC, [available online] http://www.nerc.com/about/

## BIOGRAPHICAL INFORMATION

Ming-Tse Kuo received Bachelor's and Master's degrees in electrical engineering from National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C, in 1996 and 1998. His employment experience includes being a product engineer in Silicon Integrated Systems Corp. in 2000 and MMIC engineer in Transcom Inc. in 2001. His major interests include electric power system analysis, deregulated power systems, power system transients and dynamic stability.