POWER SYSTEM DYNAMIC SIMULATION

PROGRAM - USERS MANUAL

Prepared by

R. PODMORE

Department of Electrical Engineering

University of Saskatchewan

Saskatoon January 1974

ACKNOWLEDGEMENTS

This work was performed by the author as a Post-doctoral Fellow under the direction of Professor R.J. Fleming. Financial support for the project was received from the National Research Council of Canada under Grant No. 2765.

TABLE OF CONTENTS

ABSTRACT

This report describes the relevant information for the usage of a special purpose program for simulating power system dynamic behaviour. The report includes details of the program theory, method and capability as well as instructions for data preparation with an example study.

1. INTRODUCTION

1.1 Program Objectives

The program has been developed as a special purpose program for allowing detailed study of the effects of generators, excitation systems and turbine-governors on power system transient and dynamic stability. A particularly high degree of flexibility is allowed in the representations of the machines and their controllers. A wide range of possible machine and controller models is available within the program. Furthermore, the program user is also able to include subroutines which represent his own specialized models. Consequently, in a particular study, any desired degree of machine and controller representation can be included and non-standard representations can be handled without difficulty.

In the program development, detailed information on the network behaviour has been considered to be of minor importance. This consideration has allowed the production of a program which is much more compact and easier to use than any general purpose transient stability program.

1.2 Supporting Programs

The power system simulation program requires the support of two other programs:

i) A power system loadflow program

ii) A network reduction program.

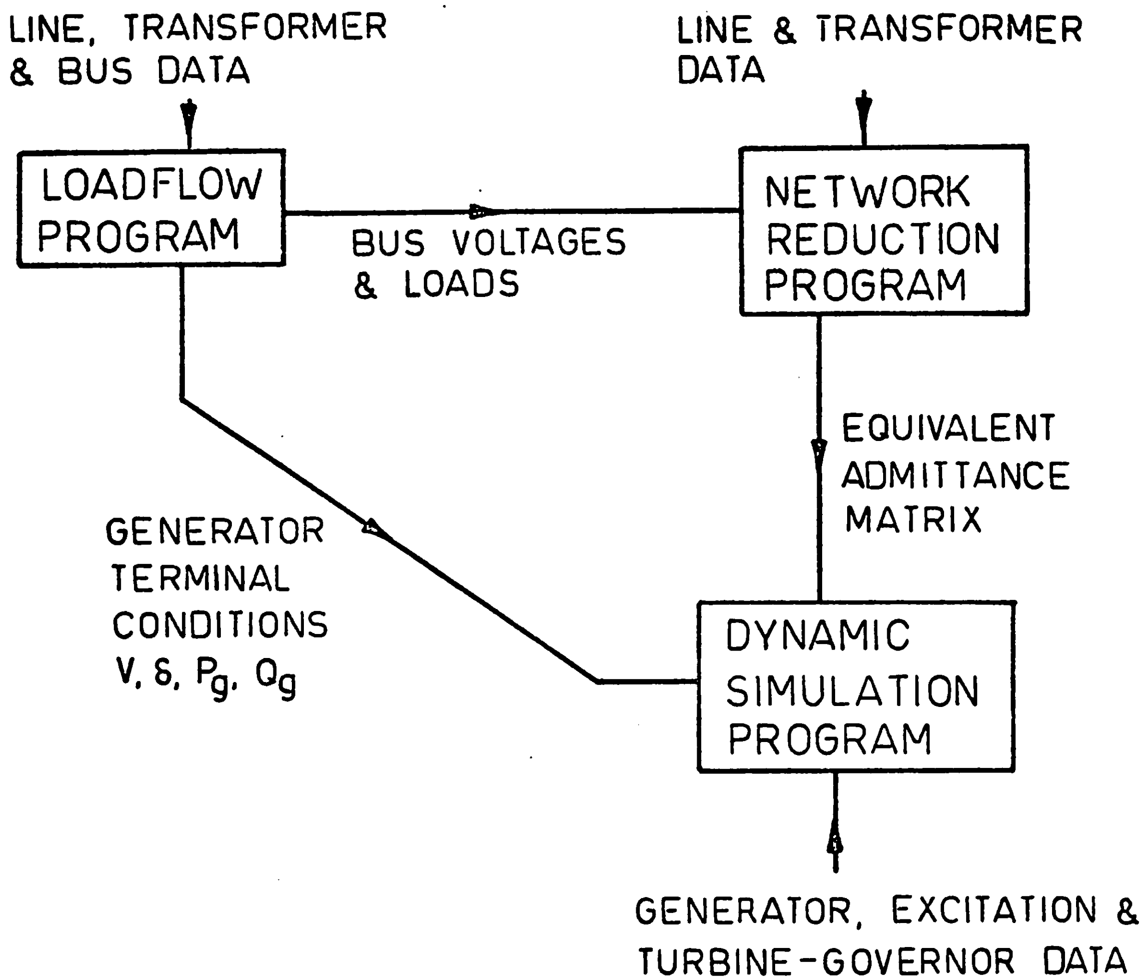


Figure 1: Data Requirements of Power System Simulation Program and its Support Programs.

The inter-related data requirements of the three programs are shown in Figure 1. The network reduction program has been separated from the simulation program for the sake of simplicity in the programming and usage. For those specialized studies in which the machine and controller behaviour is of prime importance, the loadflow and network reduction programs will be executed far less frequently than the simulation program.

The source listing for the network reduction program is supplied along with the simulation program, and the user's manual for this is included in appendix 8.4. Most general purpose loadflow programs should be suitable for providing the required supporting data. However, if required, a loadflow program and user's manual[8] may also be provided.

2. GENERAL BACKGROUND

2.1 Structure of Power System Dynamic Equations

The equations which represent the dynamic behaviour of a power system can be conveniently divided into two groups:

i) Equations describing the dynamic behaviour of the machines and their controllers. These consist of a set of differential equations which may take the form:

$$\dot{y} = g(y,z) \tag{2.1}$$

ii) Equations describing the steady state behaviour of the network and generator armature circuits. These consist of algebraic equations which may take the form:

$$h(y,z) = 0 \tag{2.2}$$

Variables y are defined as the system state variables. Variables z are referred to as auxiliary variables and are associated with the network. Generator speed and angular position are examples of state variables, whereas generator terminal voltage and current are examples of auxiliary variables.

The structure of the network equations (2.2) may be altered in time due to network changes such as fault initialization, fault clearing, line switching etc. At such instants, discontinuities occur in the auxiliary variables but not in the state variables.

2.2 Overview of Solution Technique

This section gives an overall description of the procedure which is adopted for the step by step solution of the power system dynamic equations presented in the previous section. A complete cycle of the step by step procedure can be divided into three major tasks which are outlined below:

i) The network and armature equations are solved for the auxiliary variables. That is, equation (2.2) is solved for z with the state variables y known.

ii) The time derivatives of the state variables are calculated from the machine and controller equations in (2.1).

iii) The state variables are calculated for the next increment in time by applying an approximate integration formula. The state variables and the state variable derivatives at the present and preceding time interval are used to calculate the state variables at a new time interval.

The first step in the cycle relies on knowing the present values of the state variables. Thus, the initial values of the state variables must be calculated before commencing the first cycle. However, when step i) is repeated in the succeeding cycles it uses the values of the state variables previously calculated in step iii).

2.3 Solution of Network and Armature Equations

This section describes the network and generator armature equations (2.2) and outlines the method for solving the auxiliary variables z when the state variables y are known. This forms a major step in the overall solution procedure as outlined in section 2.2. The state variables, which are specifically used in the solution comprise of the internal voltages E_q' and E_d' and rotor position Θ for each generator. The auxiliary variables which are calculated are the complex voltages and currents at the generator terminals. The method for solving the network and armature equations includes an effective technique for handling generator transient saliency ($X_d' \neq X_q'$). [3]

We consider the transmission system to be represented by the matrix of driving point and transfer admittances seen from the terminals of the generators. This matrix includes the system loads which are represented as constant impedances. We can, therefore, describe the network by the equation:

$$[\bar{I}] = [Y_{TT}][\bar{V}] \tag{2.3}$$

where $\bar{I}$ and $\bar{V}$ are based upon the synchronous reference frame.

For each synchronous generator we have two scalar equations describing the armature circuits:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} E_d' \\ E_q' \end{bmatrix} - \begin{bmatrix} R_a & -X_q' \\ X_d' & R_a \end{bmatrix} \begin{bmatrix} I_d \\ I_q \end{bmatrix} \tag{2.4}$$

In equation (2.4) the voltages and currents are based upon the individual d,q reference frames of each generator. In order to obtain a combined solution of equations (2.3) and (2.4) all the equations must first be transformed to a common reference frame. Let,

$$\bar{I} = I_{real} + jI_{imag} \tag{2.6}$$

then,

$$I_d + jI_q = \bar{I}\, e^{-j\Theta} \tag{2.6}$$

where Θ is the angle between the generator d axis and the synchronous reference. Similar expressions also exist for $\bar{V}$ and $\bar{E}$. If transient saliency is neglected ($X_d' = X_q'$) then the simultaneous solution of equations (2.3) and (2.4) is quite simple. In this case, the two scalar equations in (2.4) can be combined into a single complex equation:

$$(V_d + jV_q) = (E_d' + jE_q') - (R_a + jX_d')(I_d + jI_q) \tag{2.7}$$

Equation (2.7) transforms to the synchronous reference to become:

$$\bar{V} = \bar{E}' - (R_a + jX_d')\bar{I} \tag{2.8}$$

Equations (2.3) and (2.8) can be solved by adding the generator transient impedances $R_a + jX_d'$ at the network terminals. The terminal voltages $\overline{V}$ and currents $\overline{I}$ are calculated from knowing the voltages $\overline{E}' = (E_d' + jE_q')\, e^{j\Theta}$ behind the transient impedances.

On the other hand, if transient saliency is not neglected the simultaneous solution of equations (2.3) and (2.4) is considerably more difficult. In this case, when (2.4) is transformed to the synchronous reference frame we obtain:

$$\begin{bmatrix} V_{real} \\ V_{imag} \end{bmatrix} = \begin{bmatrix} E'_{real} \\ E'_{imag} \end{bmatrix} - \begin{bmatrix} \cos\Theta & -\sin\Theta \\ \sin\Theta & \cos\Theta \end{bmatrix} \begin{bmatrix} R_a & -X_q' \\ X_d' & R_a \end{bmatrix} \begin{bmatrix} \cos\Theta & \sin\Theta \\ -\sin\Theta & \cos\Theta \end{bmatrix} \begin{bmatrix} I_{real} \\ I_{imag} \end{bmatrix} \tag{2.9}$$

Two difficulties arise in the solution of equation (2.9):

i) It cannot be combined into a single complex equation with the form of (2.7).

ii) The coefficients of I_{real} and I_{imag} are functions of Θ and are therefore time varying.

It is possible to combine equations (2.3) and (2.9) to obtain a set of 2N (N = no. of generators) real equations which can be solved directly. However, because the coefficients are time varying, the matrix must be refactored or reinverted for every integration time step and this is very time consuming.

One effective method for overcoming these difficulties has been presented by Dommel and Sato[3]. The method requires iterations at each integration time step but uses a matrix which is constant as long as a certain network configuration exists. The method is based upon representing the generator by a fictitious slack voltage behind a fictititous admittance as shown in Figure 2. The generator fictitious admittance is defined by:

$$\overline{Y}^{fict} = \frac{R_a - j\tfrac{1}{2}(X_d' + X_q')}{R_a^2 + X_d' X_q'} \tag{2.10}$$

and the generator ficititious slack voltage is calculated as:

complex conj.

$$\overline{E}^{fict} = \overline{E}' + \frac{j\tfrac{1}{2}(X_q' - X_d')}{R_a - j\tfrac{1}{2}(X_d' + X_q')} (\overline{E}'^{*} - \overline{V}^{*})\, e^{j2\theta} \tag{2.11}$$

Equation (2.11) is derived (see appendix 8.2) by considering that the current produced by $\overline{E}^{fict}$ behind $\overline{Y}^{fict}$ should be identical to that obtained by solving equation (2.4) for given E_d' and E_q'.

When the fictitious generator admittances are added to the network, the combined equations for the network and generator armature circuits are of the form:

$$\begin{bmatrix} 0 \\ \overline{I} \end{bmatrix} = \begin{bmatrix} Y_{TT} & Y_{TG} \\ Y_{GT} & Y_{GG} \end{bmatrix} \begin{bmatrix} \overline{V} \\ \overline{E}^{fict} \end{bmatrix} \tag{2.12}$$

Eliminating $\overline{V}$ with Kron's reduction formula we obtain:

$$[\overline{I}] = [\overline{Y}_{GG} - \overline{Y}_{GT}\, \overline{Y}_{TT}^{-1}\, Y_{TG}][\overline{E}^{fict}] \tag{2.13}$$

The simultaneous solution of equations (2.11) and (2.13) is obtained by iterating the voltage $\overline{V}$. The exact details of the iterative procedure are later described in section 4.3. The convergence of the procedure is very reliable and usually requires only two or three iterations.

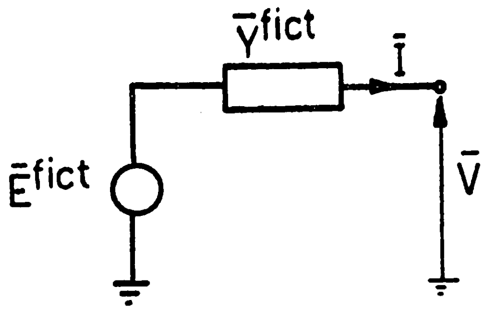


Figure 2: Equivalent Circuit for Generator Armature Equations

Figure 3: Graphical Interpretation of Approximate Integration Formula

2.4 Modelling of Machines and their Controllers

2.4.1 Introductory Remarks

This section describes the approach which is adopted for solving the dynamic equations of the machines and their controllers. Essentially, we are concerned with calculating the time derivatives of the machine and controller state variables ($\dot{y}$) when the state variables (y) and the auxiliary network variables (z) are known. This function forms a basic part of the step by step simulation procedure which was outlined in section 2.2.

For a purpose of providing an uncomplicated overall description of the problem, the machine and controller equations were presented in section 2.1 as:

$$\dot{y} = g(y,z) \tag{2.2}$$

However, in practice the mathematical representations of the machines and their controllers will usually take a more general form. In the following treatment we do not assume any specific form for the model equations but instead we consider that the model is described by a control system block diagram. From this basis the problem is approached in the same way as if it were to be solved using an analog computer.[1] An analog computer diagram is drawn to correspond with the control system block diagram. Then, the analog computer diagram is used as a guide for applying a simple procedure for generating the digital computer statements which solve the model equations.

2.4.2 Advantages of Analog Computer Approach

Several advantages are obtained in using the analog computer type of approach as compared with a purely mathematical approach such as the state space one:

i) The analog computer formulation requires a minimal amount of manipulation on the original model.

ii) The solution is always calculated in terms of variables which have physical meaning.

iii) All of the model variables are readily available for processing operations such as saturation, switch condition testing and data output.

iv) Changes to the model can be isolated in the analog diagram and included very easily in the program. On the other hand even simple model changes can often require wide spread program changes when a state space formulation is used.

v) Complex operations such as non-linear function generation, time delays and variable differentiation are easily included.

vi) The simplicity of the approach makes it quite easy for the engineer to develop program subroutines which represent his own particular models.

2.4.3 Subdivision of Machine and Controller Models

The auxiliary network variables provide the only source of interconnection between the individual generating units. Because these variables are initially calculated by solution of the network equations and thereafter are considered as fixed over the integration time step, the dynamic equations of each generating unit may be solved separately. In other words, for each integration time step the terminal voltage and current of each machine are temporarily considered as fixed inputs on the machine and controller block diagrams.

The model of each complete generating unit can be divided into models of machine, excitation system and turbine-governor. These models can also be treated separately provided that their interconnecting variables are accounted for.

The subdivision of the machine and controller models facilitates the development of separate subroutines for modelling the different types of generator, excitation system and turbine-governing apparatus. Each equipment subroutine has a standard set of input and output variables and the subroutines which model the same class of equipment are interchangeable. Each subroutine can perform the required calculations for any number of devices of the same type. In this way, the equipment subroutines form the basic building blocks which can be combined in various ways to model the different generating units in a particular study.

2.4.4 Procedure for Development of Equipment Subroutines

This section describes a general procedure for generating the digital computer statements for modelling the various types of generators, excitation systems and turbine-governors. The procedure is illustrated by generating the FORTRAN statements for modelling a static excitation system.

The essence of the procedure is contained in the following statement. On the analog computer diagram of the machine and controller models, the integrator outputs correspond to the state variables and the net integrator inputs correspond to the time derivatives of the state variables. Therefore, the calculation of the time derivatives of the state variables with known state variables is equivalent to calculating the values of the integrator inputs with the integrator outputs held constant.

The procedure for developing the equipment model subroutines consists of the following steps:

i) An analog computer diagram for the model is drawn. On the diagram generator terminal voltage and current are shown as inputs.

ii) Names are assigned to the variables at integrator outputs and various other points on the diagram. Sufficient variables should be named so that the relationships between the variables are quite obvious.

iii) Expressions for the initial values of the integrator outputs and the controller set points are obtained by considering that the net integrator inputs must be zero during the steady state.

iv) The subroutine for the solution of the machine and controller model is written. The subroutine consists of two sections; one calculates the initial conditions of the state variables and controller set points prior to the simulation and the other calculates the time derivatives of the state variables at each integration step during the simulation. The statements for calculating the initial conditions are generated directly from the expressions derived in step iii). The statements for calculating the time derivatives of the state variables are generated to execute the following steps:

I Definition of variables at integrator outputs.

II Calculation of variables at intermediate points in analog diagram. These statements must be ordered so that each variable is calculated only after its dependent variables are known.

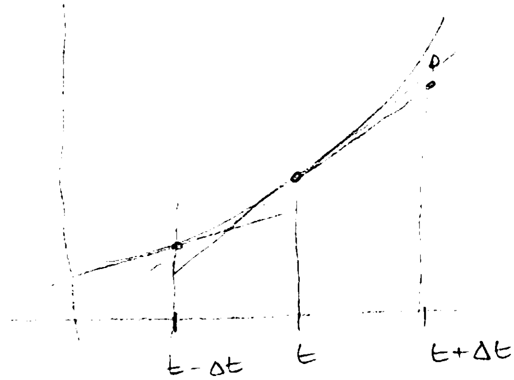
III Calculation of the net integrator inputs.

The procedure is now illustrated by generating the FORTRAN statements for modelling a static excitation system. A block diagram of the excitation system being modelled is shown in Figure 6 and the corresponding analog computer diagram is shown in Figure 14. Note that on the analog diagrams which are shown the integrators do not produce a sign change.

The integrator outputs are contained in the array OUT(I), and the integrator inputs are contained in the array PLUG(I). Thus, the variables OUT(I) are known at the start of the procedure and the variables PLUG(I) are calculated. The statements for the solution of the excitation system equations are shown below. The names of the FORTRAN variables can easily be linked with the names of the variables on the analog diagram in Figure 14.

```
C  DEFINE VARIABLES AT INTEGRATOR OUTPUTS
      X5 = OUT(5)
      X3 = OUT(6)
C  CALCULATE VARIABLES AT INTERMEDIATE POINTS
      X1 = VREF - VT
      EF = X5
      IF(EF .LT. -KP*VT) EF = -KP*VT
      IF(EF .GT. KP*VT) EF = KP*VT
      X2 = EF*KF/TF - X3
      X4 = X1 - X2
C  CALCULATE INTEGRATOR INPUTS
      PLUG(5) = X4*KA/TA - X5/TA
      PLUG(6) = X2/TF
```

The initial values of the integrator outputs and the reference voltage set point are calculated assuming the field voltage EF and terminal voltage VT are given:




```
C  CALCULATE INITIAL INTEGRATOR OUTPUTS
      OUT(5) = EF
      OUT(6) = EF*KF/TF
C  CALCULATE VOLTAGE REFERENCE SET POINT
      VREF = VT + EF/KA
```

2.5 Step by Step Integration of Dynamic Equations

2.5.1 Integration Formula

The program uses a simple one pass integration procedure which is flexible and easily programmed. Consider a general set of differential equations with state variables y and assume that the time derivatives of the state variables can be calculated at any instant in time. Now,

$$y_{t + \Delta t} = y_t + \int_t^{t + \Delta t} \dot{y} \, dt \qquad (2.14)$$

An approximation for the integral in equation (2.14) is graphically represented in Figure 3. In the approximation we assume that the slope of $\dot{y}$ for the interval t to t + Δt is equal to the slope of $\dot{y}$ for the previous interval t - Δt to t. Thus, from equation (2.14):

$$y_{t + \Delta t} = \dot{y}_t \Delta t + (\dot{y}_t - \dot{y}_{t - \Delta t}) \Delta t/2 + y_t \qquad (2.15)$$

Equation (2.15) provides the required formula for calculating the state variables for a new time step from the state variables and their derivatives in the preceding time steps.

2.5.2 Selection of a Suitable Integration Step

To obtain a solution which is numerically stable the integration step should obey the condition[1]:

$$\Delta t < T_{min}/5$$

where T_{min} is the period of the highest frequency which is present in the solution. In practice, T_{min} will not usually be known before the solution but satisfactory performance can usually be obtained by selecting the integration step such that:

$$\Delta t < TC_{min}/5$$

where TC_{min} is the smallest time constant in the system.

Any instability which may arise in the solution is quickly shown up as an oscillation with a period equal to twice the integration step. The accuracy of the solution increases as the integration step is decreased and in practice a quick check on the solution accuracy can be made by repeating a run with the integration step halved.

3. MACHINE AND CONTROLLER REPRESENTATIONS

3.1 Synchronous Generator Models

The synchronous generator model is based upon the two axis representation of a machine with a d axis field winding and a q axis damper on the rotor. By appropriate selection of data this model is used for the following generator representations:

i) Round rotor machine including transient saliency. Saturation is represented in both axes as a function of the total air gap flux.

ii) Salient Pole Machine including transient effects. In this case the q axis damper winding is not utilized. Saturation is represented only in the d axis as a function of the d axis air gap flux.

iii) Voltage behind transient reactance.

iv) Infinite Bus.

A block diagram of the synchronous generator model is shown in Figure 4. The model equations are summarized below: The generator electrical equations are: (Appendix 8.1)

q axis stator equation,

$$V_q = E_q' - X_d' I_d - R_a I_q \tag{3.1}$$

d axis stator equation,

$$V_d = E_d' + X_q' I_q - R_a I_d \tag{3.2}$$

d axis field equation,

$$\dot{E}_q' = (kE_f - E_q' - (X_d - X_d') I_d)/T_{do}' \tag{3.3}$$

q axis damper equation,

$$\dot{E}_d' = (-E_d' + (X_q - X_q') I_q)/T_{qo}' \tag{3.4}$$

The rotor mechanical equations are:

$$\dot{\omega} = (P_m - P_e - P_1 - D\omega)/2H \tag{3.5}$$

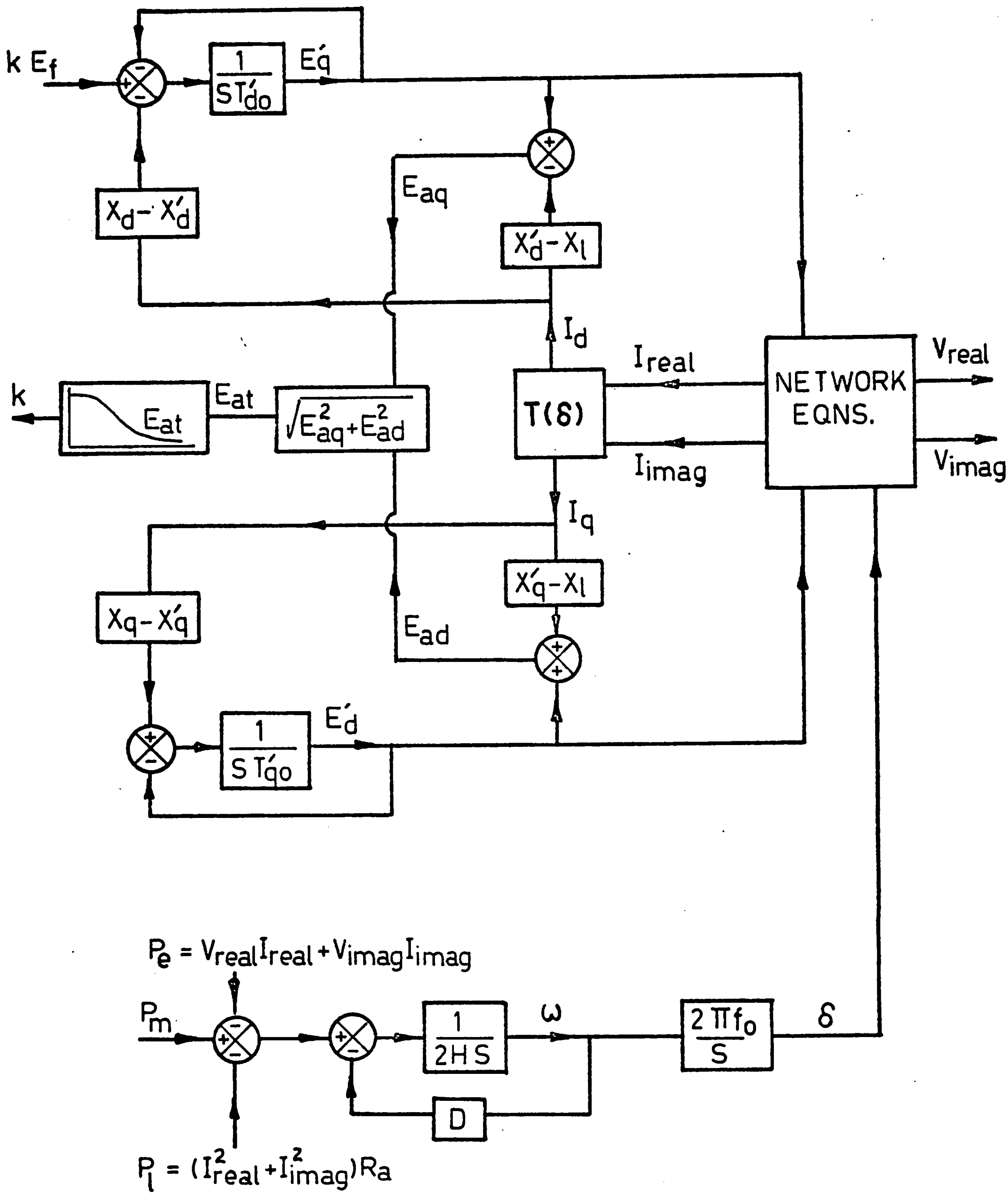


Figure 4: Synchronous Generator Model.

$$\dot{\delta} = 2\Pi f_o \omega \tag{3.6}$$

The machine saturation is represented by varying the reactances and time constants as a function of a saturation factor k, which is in turn a function of the air gap flux. (A method for obtaining an analytical expression for k from open circuit data is described in appendix 8.3.) We assume that the generator mutual reactances saturate according to:

$$X_{ad} = kX_{ad}^{o} \tag{3.7}$$

and

$$X_{aq} = kX_{aq}^{o} \tag{3.8}$$

It follows that the saturated synchronous reactances are given by (Appendix 8.1.6):

$$X_d = kX_d^o + (1 - k)X_\ell \tag{3.9}$$

and

$$X_q = kX_q^o + (1 - k)X_\ell \tag{3.10}$$

It also follows that the saturated open circuit transient time constants are given by (Appendix 8.1.6):

$$T'_{do} = T'^{o}_{do} \left[1 - (1 - k)(X_d^o - X'_d)/(X_d^o - X_\ell)\right] \tag{3.11}$$

and

$$T'_{qo} = T'^{o}_{qo} \left[1 - (1 - k)(X_q^o - X'_q)/(X_q^o - X_\ell)\right] \tag{3.12}$$

The transient reactances X'_d and X'_q are not sensitive to typical variations in X_{ad} and are assumed to be independent of the generator saturation level.

To determine the machine saturation level the air gap voltage is calculated using (Appendix 8.1.4):

$$E_{aq} = E'_q - (X'_d - X_\ell)I_d \tag{3.13}$$

$$E_{ad} = E_d' + (X_q' - X_\ell)I_q \tag{3.14}$$

and

$$E_{at} = \sqrt{E_{aq}^2 + E_{ad}^2} \tag{3.15}$$

In the preceding description saturation is represented in both axes as a function of the total air gap voltage. For a salient pole machine saturation is represented only in the d axis. In this case, only d axis quantities are adjusted and the saturation factor is calculated as a function of just E_{aq}.

3.2 Excitation System Models

The program allows for the following excitation system representations:

i) Fixed excitation

ii) IEEE Type 1 representation[4] of a rotating excitation system. Figure 5 shows the model block diagram.

iii) IEEE Type 1s representation[4] of a static excitation system. The model block diagram is shown in Figure 6.

iv) A model of a rotating excitation system with an auxiliary stabilizer. (This model was developed for a study of the stabilization of the generators at Squaw Rapids on the Saskatchewan Power Corporation's system). A block diagram of this model is shown in Figure 7.

Further details of the excitation system models listed under ii) and iii) may be found in reference 4.

3.3 Turbine and Governor Models

The program allows for the following turbine-governor models:

i) Constant mechanical input power.

ii) One, two or three stage tandem steam turbine and governor. A block diagram of the model is shown in Figure 8.

iii) A linear model of a hydraulic turbine and governor. The model block diagram is shown in Figure 9.

Further information on the turbine and governor models may be obtained from reference 5.

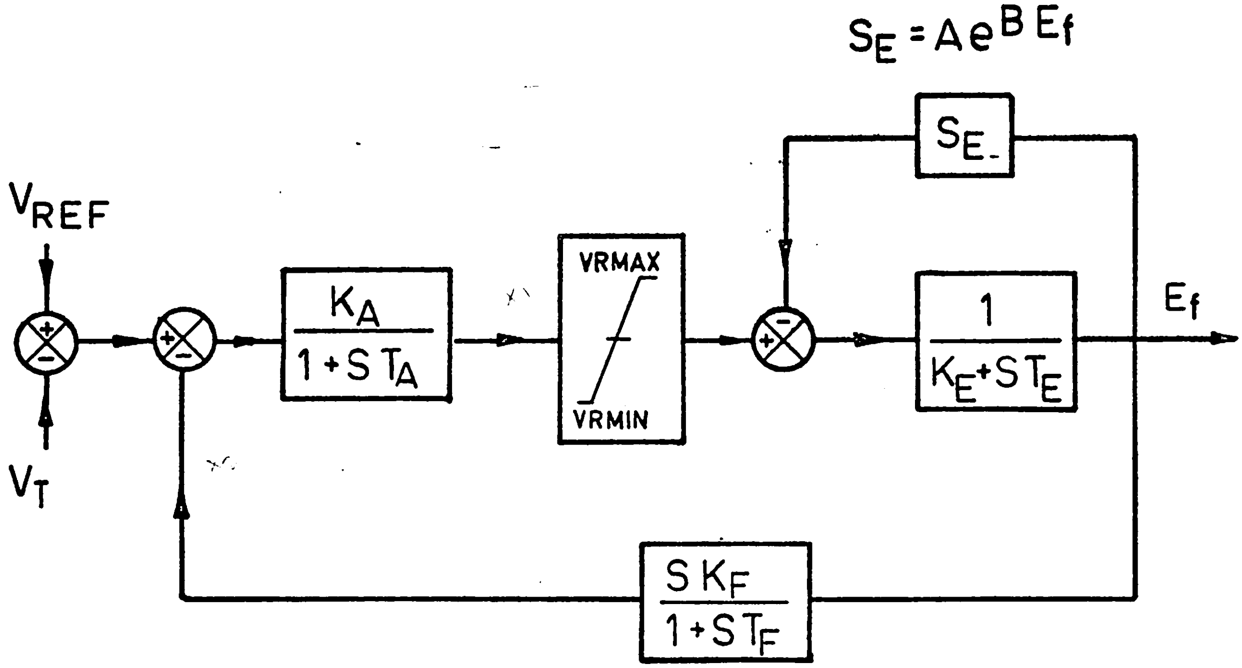


Figure 5: IEEE Type 1 Rotating Excitation System Model

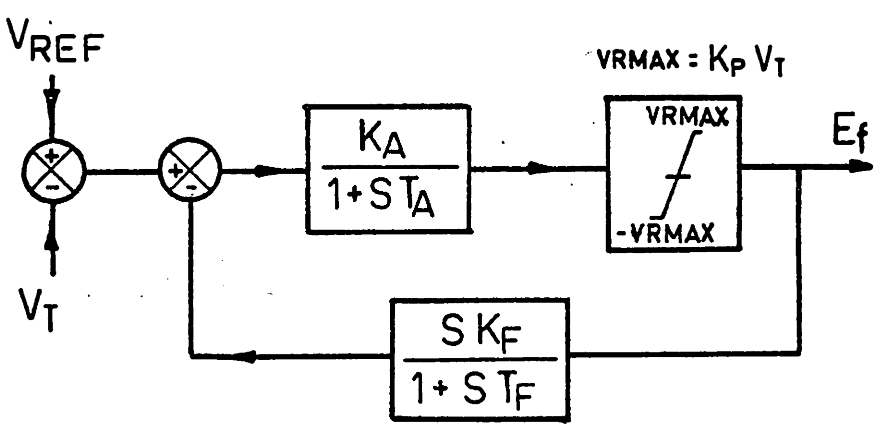


Figure 6: IEEE Type 1s Static Excitation System Model

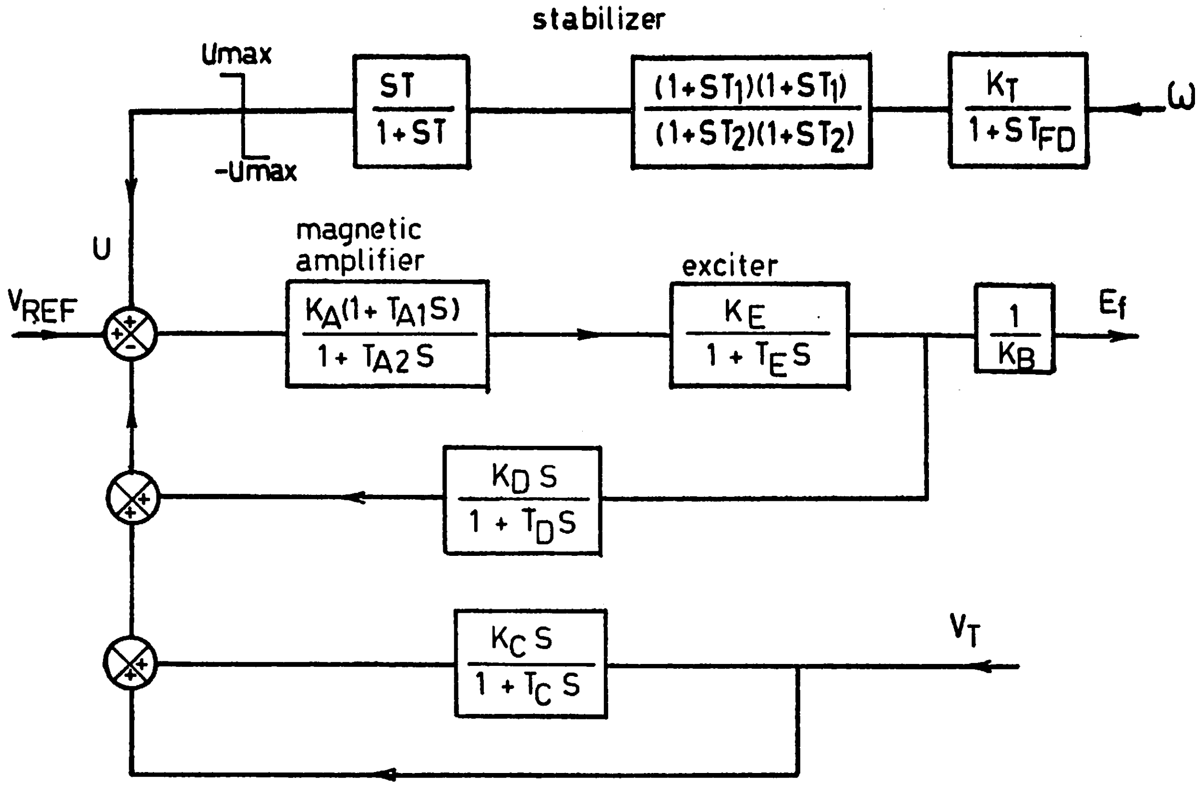


Figure 7: Model of Rotating Exciter with Auxiliary Stabilizer.

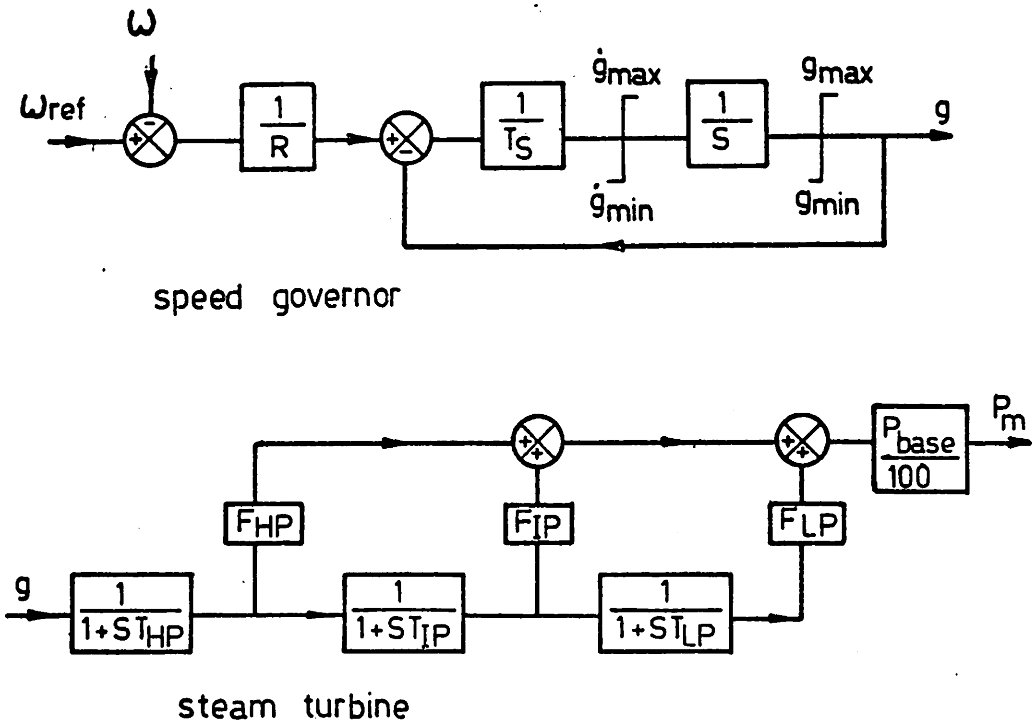


Figure 8: Steam Turbine and Governor Model

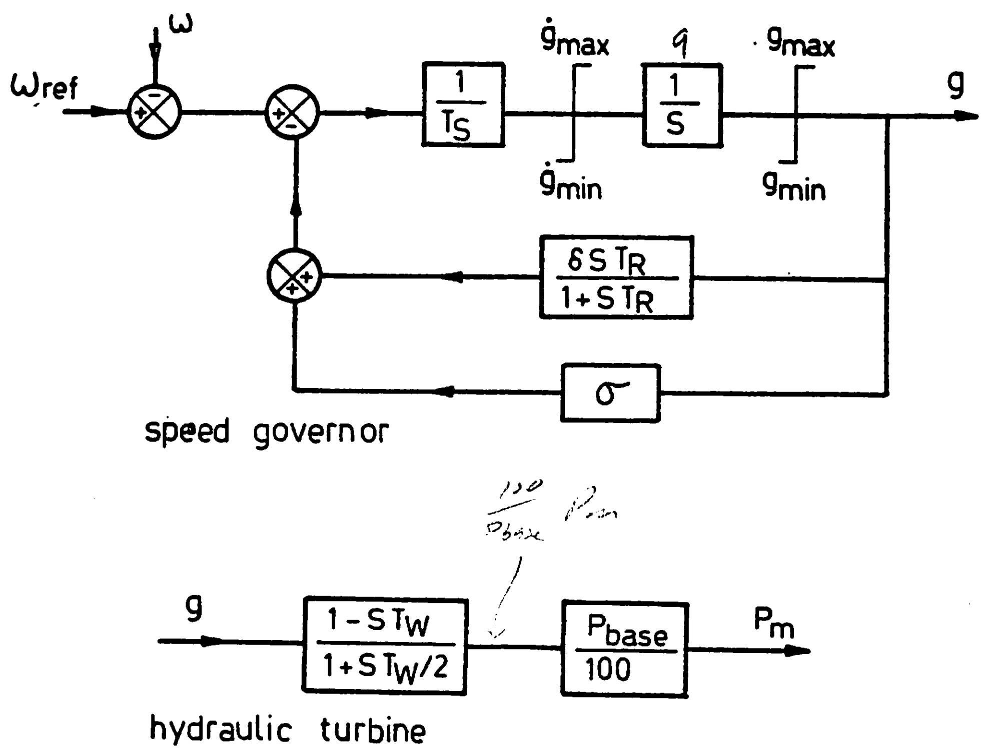


Figure 9: Hydraulic Turbine and Governor Model

4. PROGRAM DESCRIPTION

4.1 Overall Program Structure

A flow diagram of the overall program is shown in Figure 10. The program is divided into the following subroutines:

i) A mainline routine which performs data initialization and input and which provides overall control of the program.

ii) MATRIX - a subroutine for forming the equivalent admittance matrix of the internal generator buses.

iii) NWSOL - a subroutine for solving the network equations. This subroutine calculates the terminal voltages and currents as functions of the generator internal voltages.

iv) A library of equipment subroutines for modelling the various types of generators, excitation systems and turbine-governor systems. Each equipment subroutine consists of two parts; one calculates the initial conditions of the state variables and is executed once at the start of the program, the other calculates the time derivatives of the state variables for each integration time step.

v) INT - the integration subroutine calculates the state variables for a new time interval as a function of the state variables and state variable derivatives for the present and preceding intervals.

vi) OUTPUT and PLOT - these subroutines respectively give printouts and plots of the variables calculated during the simulation.

The function of the various subroutines is further illustrated by Figures 11 and 12 which show the relationships between subroutine input and output variables at two different stages in the program. In the following sections the algorithms of the less straightforward subroutines are described in detail.

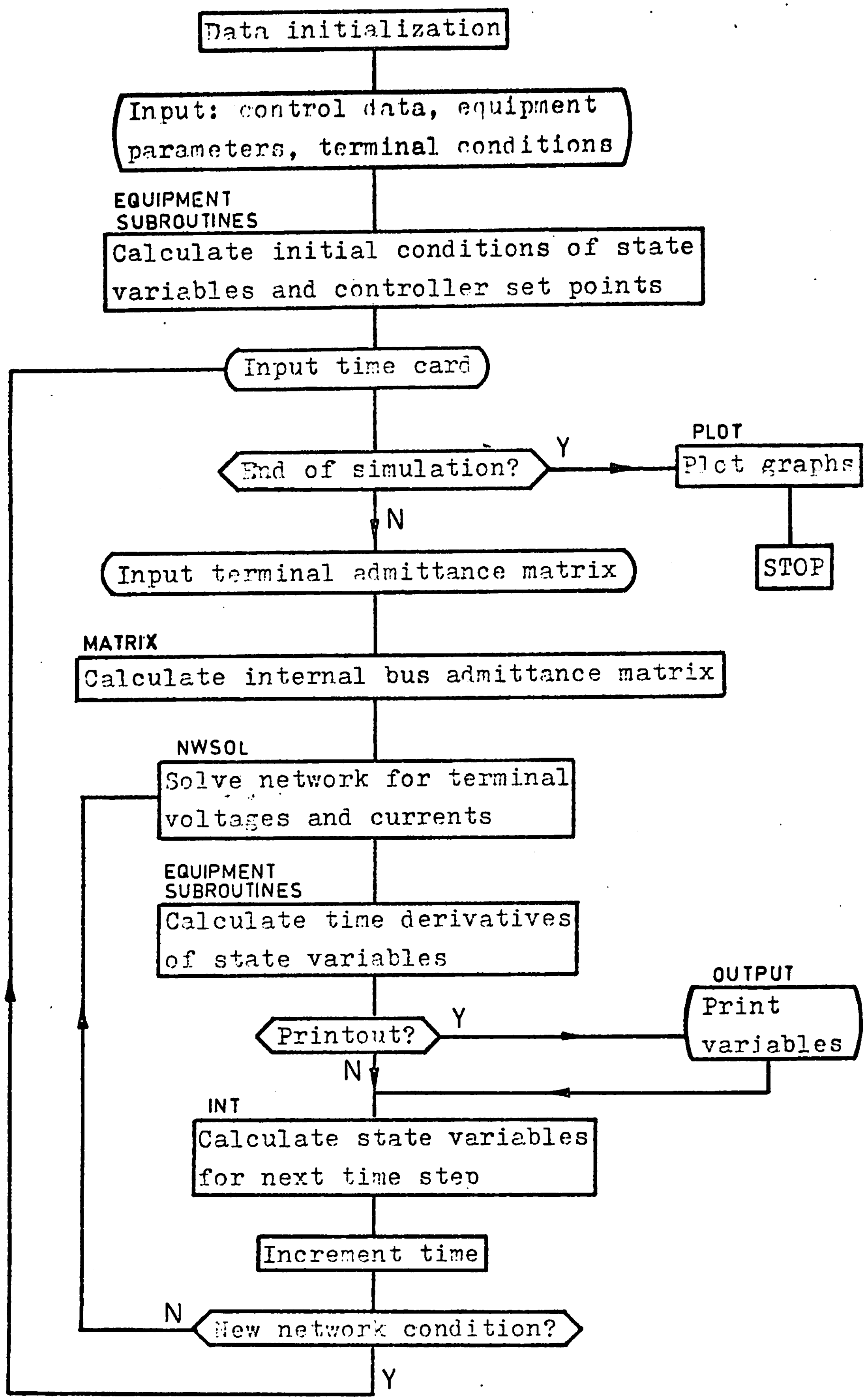


Figure 10: Overall Program Flow Diagram

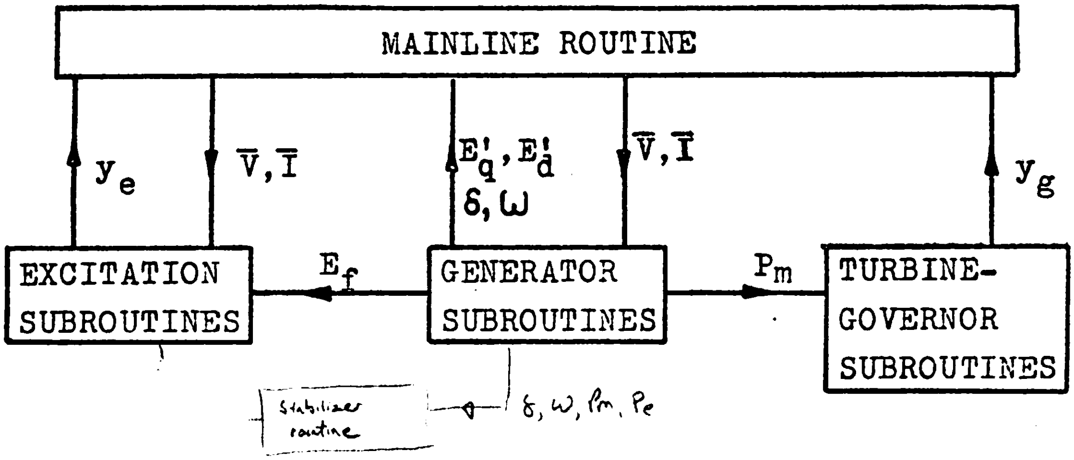


Figure 11: Relationship of Subroutine Variables for Calculating Initial Conditions

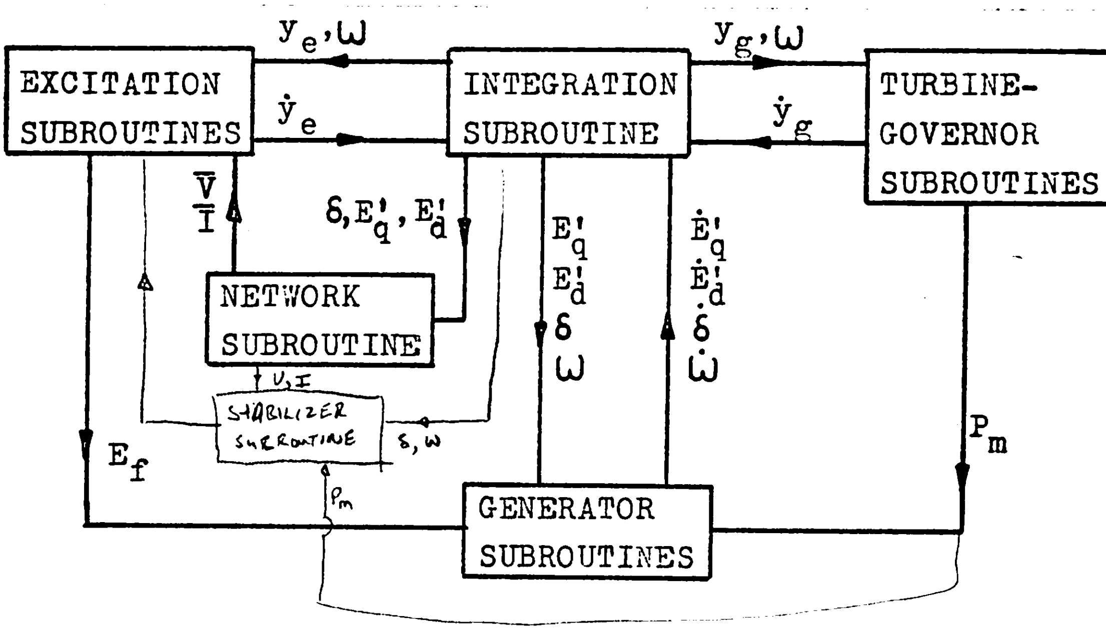


Figure 12: Relationship of Subroutine Variables During Step by Step Simulation

Order of execution - Turbine
Stabiliser.
Exciter.
Generator

4.2 Network Reduction Subroutine

The subroutine MATRIX adds each internal generator bus to the network admittance matrix and then reduces the new matrix by eliminating the corresponding generator terminal bus. For each generator bus the following steps are applied to a matrix which is initialized as the equivalent admittance matrix for the terminal buses.

i) The row and column for the terminal of the generator in concern is moved to the outside of the matrix.

ii) The generator internal bus is added to the matrix in the position of the recently vacated row and column.

iii) The terminal bus row and column is eliminated from the matrix using Kron's reduction formula.

When the procedure is completed for each generator, the final matrix is the equivalent admittance matrix for just the internal generator buses. The procedure requires only one extra row and column of storage apart from that used by the original terminal admittance matrix.

4.3 Network Solution Subroutine

The network solution subroutine calculates the terminal voltages and currents from the generator internal voltages and rotor angles. As described in section 2.3 an iterative procedure is used to account for transient saliency. The solution procedure is as follows:

i) Set initial estimate of $\overline{V}$ to the terminal voltage obtained in the previous time step.

ii) Transform generator internal voltages to the synchronous reference frame. For each generator calculate:

$$\theta = \delta - 90$$

and

$$\overline{E}' = (E_d' + jE_q')e^{j\theta}$$

iii) Calculate the fictitious internal voltage $\overline{E}^{fict}$ for each generator using equation (2.11).

iv) Calculate the generator currents $\overline{I}$ using equation (2.13).

v) Calculate new estimates of the terminal voltages from:

$$\overline{V} = \overline{E}^{fict} - \overline{I}/\overline{Y}^{fict}$$

vi) Check for convergence. For each generator calculate:

$$I_d + jI_q = \overline{I}\, e^{-j\theta}$$

and

$$V_d + jV_q = \overline{V}\, e^{-j\theta}$$

If $\qquad V_q = E_q' - X_d' I_d - R_a I_q$

and $\qquad V_d = E_d' + X_q' I_q - R_a I_d$

for every generator, then the solution has converged. If solution has not converged return to step iii).

4.4 Equipment Subroutines

4.4.1 Introductory Remarks

This section describes the subroutines which solve the dynamic equations of the machines and their controllers. The development of these subroutines is based upon the analog computer type of approach which was described in section 2.4.

As shown in Figure 10, the equipment subroutines perform two functions:

i) They calculate the initial conditions of the machine and controller state variables and the initial controller set points.

ii) They calculate the time derivatives of the machine and controller state variables at each integration time step.

The equipment subroutines are classed according to whether they model a generator, excitation system or turbine and governor. In developing the program it is assumed that all the synchronous generator models (see section 3.1) are described by a common set of state variables: namely ω, δ, E_q' and E_d'. On the other hand, no assumption is made concerning what particular state variables describe the various excitation system and turbine-governor models.

During the calculations it is necessary to account for the interconnections between the generator, excitation system and turbine-governor models. Figure 11 shows the relationship between the subroutine input and output variables during the calculation of the initial conditions. In this case, the field voltage and mechanical power are calculated by the generator subroutine and consequently this is executed before the excitation system and turbine-governor subroutines. The relationship which exists between the subroutine input and output variables during the step by step simulation is shown in Figure 12. In this case, the new values of field voltage and mechanical power are respectively calculated by the excitation system and turbine-governor subroutines. Consequently, the generator subroutine is now executed after the excitation system and turbine-governor subroutines.

4.4.2 Synchronous Generator Subroutine

The various synchronous generator representations are modelled with a single subroutine; GEN1. However, the program allows for the addition of other generator modelling subroutines if the need for this should arise.

A block diagram of the synchronous generator model is shown in

Figure 4. The synchronous generator subroutine was developed from the block diagram by following the modelling procedure which was outlined in section 2.4. The generator modelling algorithm is somewhat complicated by the representation of saturation and it is, therefore, described in detail.

The input and output variables of the generator subroutine during the initial condition calculations are shown in Figure 11. The generator subroutine receives the terminal voltage and current in terms of the synchronous reference frame as input and provides the initial conditions of the generator state variables ω, δ, E_q' and E_d' as well as the initial field voltage and mechanical power as output. The algorithm for calculating the generator initial conditions is described below. The procedure is iterative since the machine reactances are dependent upon the saturation factor and this is initially unknown.

i) Set initial estimate of saturation factor $k = 1$.

ii) Calculate saturated reactances X_d and X_q from equations (3.9) and (3.10).

iii) Calculate the angle of the generator rotor with respect to the synchronous reference. I.e. calculate:

$$\overline{E}_{qd} = \overline{V} + (R_a + jX_q)\,\overline{I}$$

then,
$$\delta = \mathrm{Arg}(\overline{E}_{qd})$$

and,
$$\theta = \delta - 90$$

iv) Calculate d and q axis components of generator current:

$$I_d + jI_q = \overline{I}\,e^{-j\theta}$$

v) Calculate the generator transient voltages and the field voltage:

$$E_q' = \mathrm{Mag}(\overline{E}_{qd}) - (X_q - X_d')I_d$$

$$E_d' = (X_q - X_q')I_q$$

$$E_f = (E_q' + (X_d - X_d')I_d)/k$$

vi) Calculate the air gap voltage from equations (3.13), (3.14) and (3.15).

vii) Calculate a new estimate of the saturation factor k as a function of the air gap voltage.

viii) If the saturation factor has converged continue to step ix), otherwise return to step ii).

ix) Calculate the initial mechanical power as the sum of the generator output and armature loss.

x) Exit from the subroutine.

During the step by step simulation, the function of the generator subroutine is to calculate the time derivatives of the generator state variables. Figure 12 shows the generator subroutine input and output variables for this calculation. The algorithm which is used to calculate the time derivatives of the generator state variables is as follows:

i) Define generator state variables ω, δ, E_q' and E_d' in terms of integrator outputs.

ii) Calculate d and q axis components of generator current. Set,

$$\theta = \delta - 90$$

$$I_d + jI_q = \overline{I}\, e^{-j\theta}$$

iii) Calculate generator electrical output and losses:

$$P_e = V_{real}I_{real} + V_{imag}I_{imag}$$

$$P_1 = (I_{real}^2 + I_{imag}^2)R_a$$

iv) Calculate saturated reactances X_d, X_q and time constants T'_{do}, T'_{qo} from equations (3.9), (3.10), (3.11) and (3.12) respectively.

v) Calculate the air gap voltage from equations (3.13), (3.14) and (3.15) and the saturation factor k.

vi) Calculate the time derivatives of the generator state variables using equations (3.3), (3.4), (3.5), (3.6) and store them as integrator inputs.

vii) Exit from the subroutine.

4.4.3 Controller Subroutines

Three excitation system modelling subroutines are presently included in the program library:

AVR1: IEEE Type 1 representation of rotating excitation system.

AVR2: IEEE Type 1s representation of a static excitation system.

AVR3: Rotating excitation system with an auxiliary stabilizer.

The turbine-governor modelling subroutines in the program library consist of:

TUR1: One, two or three stage steam turbine and governor.

TUR2: Hydraulic turbine and governor.

No subroutines are required for representing constant excitation voltage or constant mechanical input power.

The algorithms for the excitation system and turbine-governor subroutines are not described in detail. However, the analog computer diagrams which were used in generating the controller subroutines according to the procedure presented in section 2.2 are given in Figures 13 to 17. The names of the variables on the analog diagrams correspond closely with the names of the program variables and the subroutine statements can be easily understood by reference to the corresponding analog diagram.

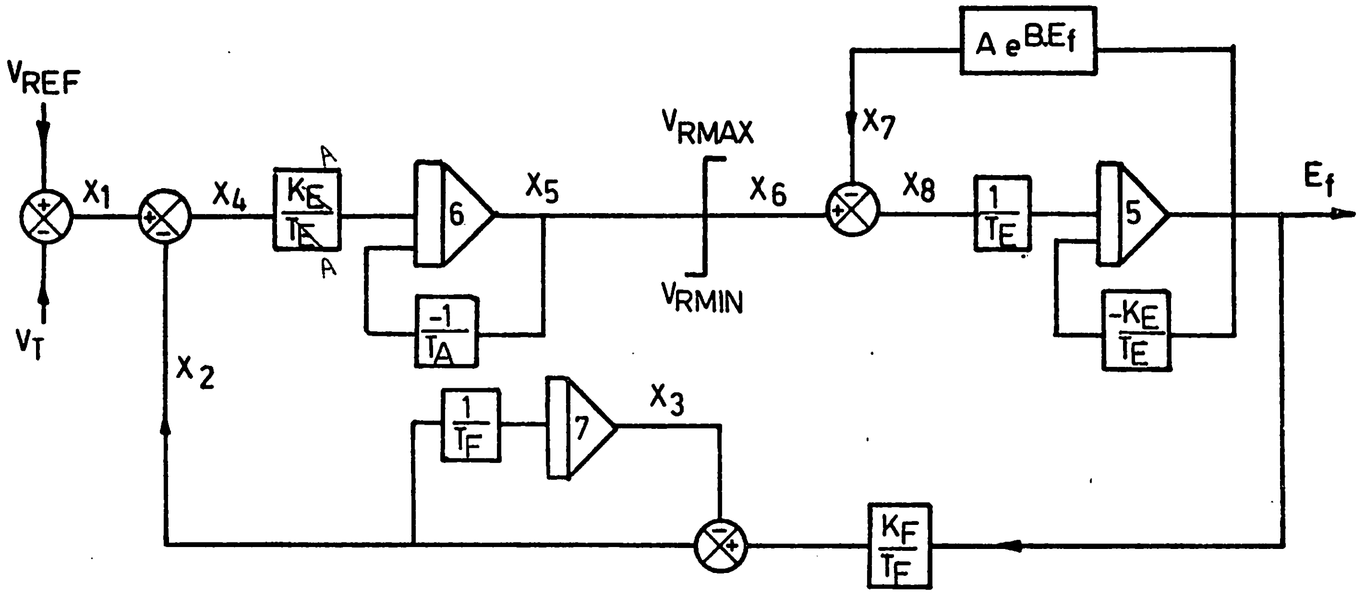


Figure 13: Analog Diagram of IEEE Type 1 Rotating Excitation System Model

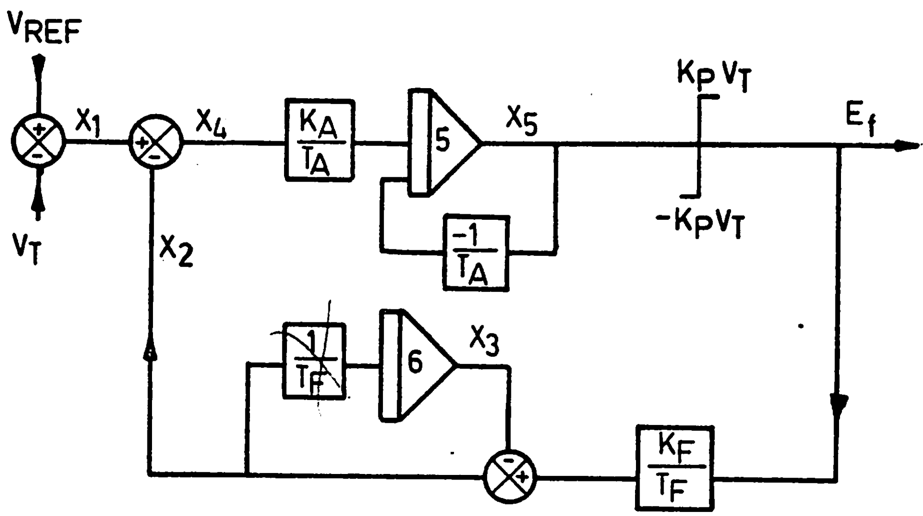


Figure 14: Analog Diagram of IEEE Type 1s Static Excitation System Model

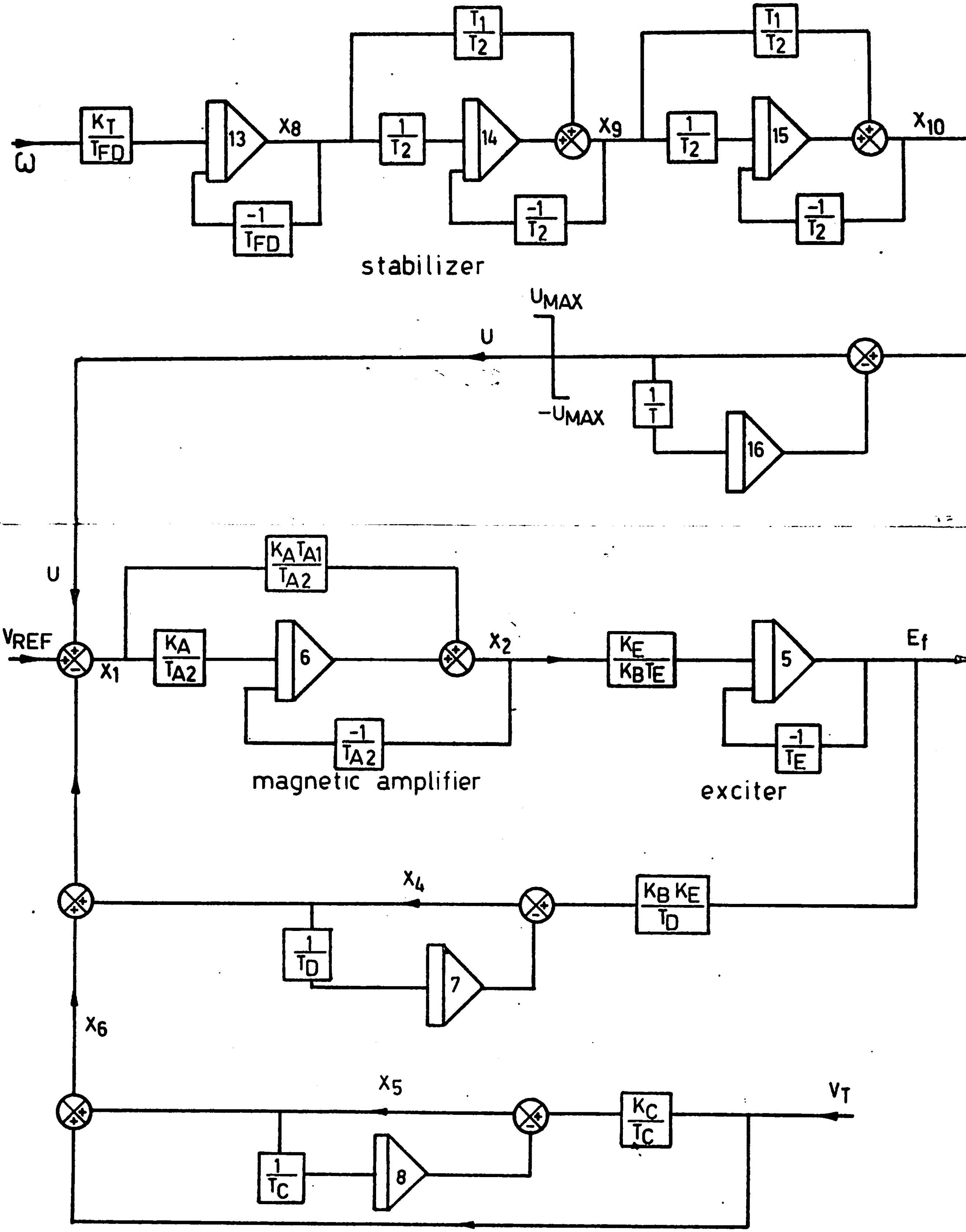


Figure 15: Analog Diagram of Rotating Exciter with Auxiliary Stabilizer

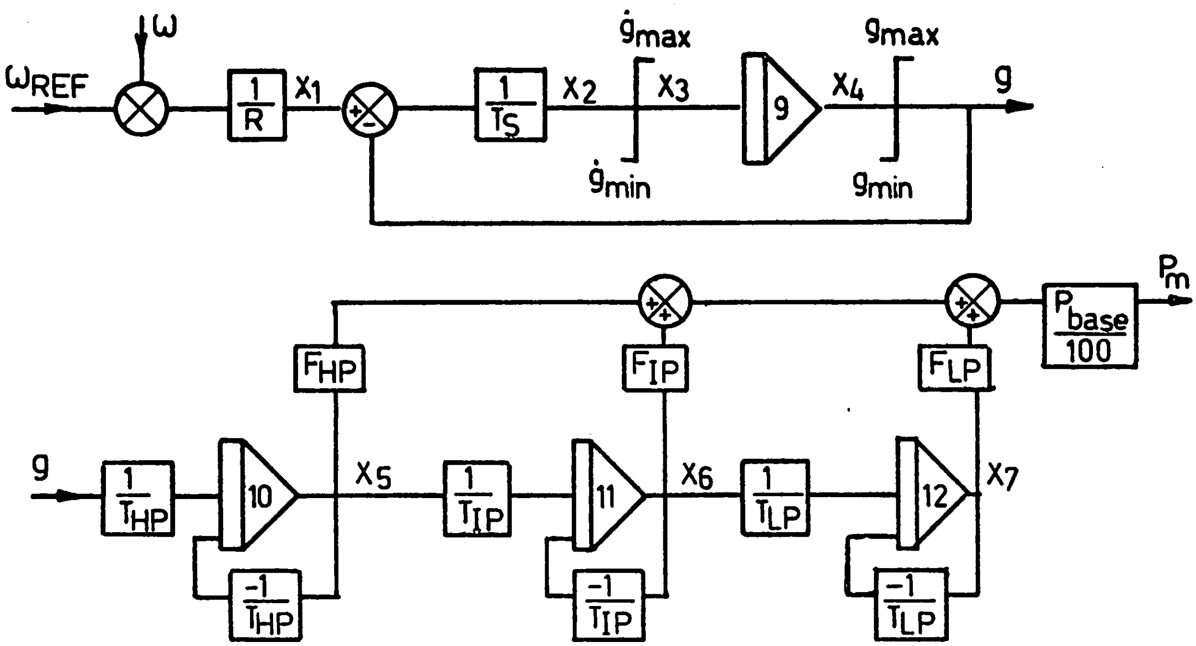


Figure 16: Analog Diagram of Steam Turbine and Governor

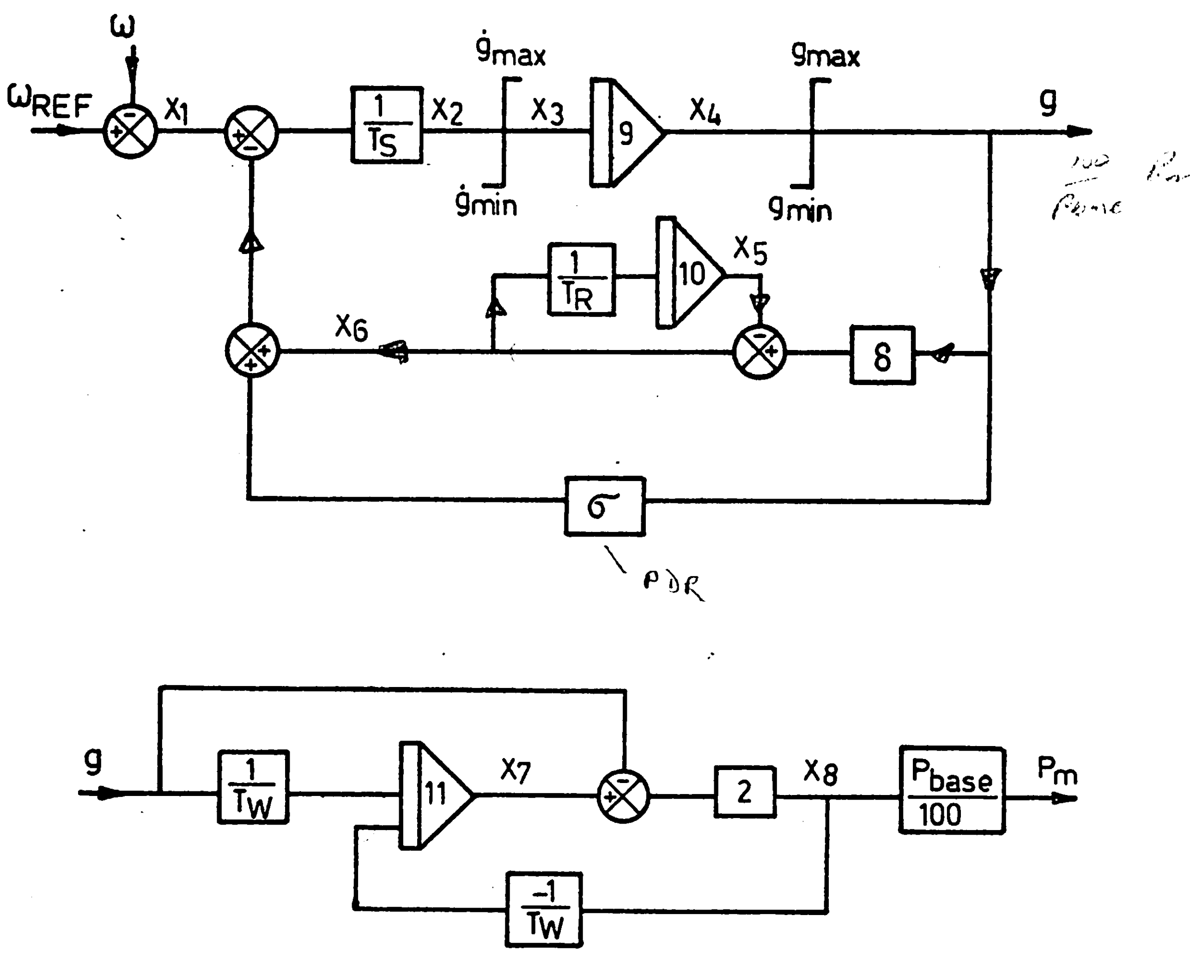


Figure 17: Analog Diagram of Hydraulic Turbine and Governor

4.5 Data Organization

The program variables which are common to several subroutines are divided into nine blocks which are listed below:

i) Time and integration step.

ii) Generator parameters.

iii) Excitation system parameters.

iv) Turbine and governor parameters.

v) Terminal voltages and currents, field voltages and mechanical input powers.

vi) Integrator input and output variables.

vii) Network admittance matrix.

viii) Graph plotting variables.

ix) Printout variables.

The program arrays are dimensioned for a maximum of 10 generating units. A maximum of 16 state variables or integrators are available for modelling each complete generating unit. The integrator variables are contained in the arrays:

PLUG (I,J) - value of integrator input (state variable derivative),

OUT(I,J) - value of integrator output (state variable),

SAVE(I,J) - value of integrator input for previous integration time step.

Where, I = no. of generating unit and J = no. of integrator. Integrators 1, 2, 3, and 4 are respectively assigned to specific generator state variables ω, δ, E_q' and E_d'. Integrators 5, 6, 7, and 8 are assigned to arbitrary excitation system state variables and 9, 10, 11, and 12 to arbitrary turbine-governor state variables. Integrators 13, 14, 15, and 16 may be assigned as required to either generator, excitation system or turbine-governor state variables.

At a specified simulation interval up to 14 variables may be printed for each generating unit. The variables to be printed are stored in the array PRTVAR(I,J), where I = no. of generator and J - no. of printout variable. The user controls the variables to be printed by inserting statements which specify the elements of PRTVAR(I,J) in the machine and controller subroutines.

A maximum of 6 variable time responses may be plotted on the line printer. Each plot may have a maximum of 200 points along the time axis. The variables to be plotted are stored in the array VAR(I,J) where I = no. of point on plot and J = no. of plot. The variables plotted are controlled by inserting statements which specify the elements of VAR(I,J) in the output subroutine.

The generator input parameters have the same definition irrespective of the type of generator representation and are defined within the mainline program and within the subroutines. On the other hand, the definition of the excitation system and turbine-governor input parameters depends upon the type of model being used. Consequently, these parameters are defined only within the controller subroutines and are considered as arbitrary parameters within the mainline program.

5. PREPARATION OF PROGRAM AND DATA

5.1 Data Preparation

The program data can be divided into the following items:

i) Control parameters.

ii) Generator parameters.

iii) Excitation system parameters.

iv) Turbine and governor parameters.

v) Initial terminal conditions.

vi) Network switching time parameter.

vii) Network terminal admittance matrix.

viii) Graph plotting parameters.

The items of data are input in the order in which they are listed. All items are entered once except for items vi) and vii). These items are re-entered for each different network configuration. The data parameters and their card format are now described for each item in turn. Unless stated otherwise, the parameters are entered as floating point data.

i) Control Parameters

5	10	20	30	
NGEN		TSTEP	TPRINT	

NGEN: Number of generators. (Integer data)

TSTEP: Integration time step.

TPRINT: Time interval for printout and plotting of variables.

ii) Generator Parameters:

The data for each generator is entered on two consecutive cards:

1 10	20	30	40	50	60	70	80
P_{base}	H	R_a	X_l	X_d	X_d'	X_q	X_q'

10	20	30	40	50	
T_{do}'	T_{qo}'	D	A	B	

P_{base} MW base for p.u. generator parameters

H Inertia constant.

R_a Armature resistance.

X_ℓ Armature leakage reactance.

X_d, X_q d and q axis synchronous reactances.

X_d', X_q' d and q axis transient reactances.

T_{do}', T_{qo}' d and q axis open circuit transient time constants.

D Mechanical damping factor.

A,B Saturation constants.

All the generator models use a common set of data. For some representations several of the above parameters are superfluous, however, they must still be input according to the following rules.

Round rotor machine: All the parameters apply and actual values should be used.

Salient pole machine: Set $X_q' = X_q$ and $T_{qo}' =$ large value (1000.0).

Constant voltage behind transient reactance: Set $X_d = X_q = X_q' = X_d'$ where the actual value of X_d' is used. Set $T_{do}' = T_{qo}' =$ large value. For an infinite bus generator also assume a very small value of X_d' and a very large value of H.

iii) Excitation System Parameters:

The data for each excitation system is entered on two consecutive cards. The data parameters are dependent upon the type of excitation system being represented.

a) IEEE Type 1 Rotating Excitation System:

10	20	30	40	50	60	70	80
K_A	K_E	K_F	T_A	T_E	T_F	V_{RMIN}	V_{RMAX}

10	20	
A S_{max}	B $S_{.75}$	

The input parameters are defined in the block diagram in Figure 5.

b) IEEE Type 1s Static Excitation System:

10	20	30	40	50		
K_A	~~K_F~~ X	~~T_A~~ K_F	~~T_F~~ T_A	~~K_P~~	T_F	K_P

The input parameters are defined in the block diagram in Figure 6.

c) Rotating Excitation System with Auxiliary Stabilizer:

10	20	30	40	50	60	70	80
K_A	T_{A1}	T_{A2}	K_B	K_E	T_E	K_D	T_D

10	20	30	40	50	60	70	80
K_C	T_C	T_1	T_2	K_T	T_{FD}	T	U_{MAX}

The input parameters shown above are defined in the block diagram in Figure 7.

d) Constant Excitation:

No input parameters are required and, therefore, two blank cards are entered.

iv) Turbine and Governor Parameters:

The data for each turbine-governor set is entered on two consecutive cards. The data parameters are dependent upon the type of turbine-governor being modelled.

a) Steam Turbine and Governor:

10	20	30	40	50	60	70	80
P_{base}	T_{HP}	T_{IP}	T_{LP}	F_{HP}	F_{IP}	F_{LP}	R

10	20	30	40	50	
T_S	$\dot{g}_{min}$	$\dot{g}_{max}$	g_{min}	g_{max}	

The steam turbine and governor input parameters are defined in the block diagram in Figure 8. For a three stage turbine all the parameters are applicable. For a two stage turbine set $F_{LP} = 0$ and T_{LP} = large value. For a single stage turbine set $F_{IP} = F_{LP} = 0$ and $T_{IP} = T_{LP}$ = large value.

b) Hydraulic Turbine and Governor:

10	20	30	40	50	60	70	80
P_{base}	T_W	T_S	T_R	σ	δ	$\cancel{\dot{g}_{min}}$	$\cancel{\dot{g}_{max}}$

10	20	
$\cancel{g_{min}}$	$\cancel{g_{max}}$ $\dot{g}_{min}$	$\dot{g}_{max}$ g_{min} g_{max}

The hydraulic turbine and governor input parameters are defined in the block diagram in Figure 9.

c) Constant Mechanical Input Power:

No input parameters are required and two blank cards are entered.

v) <u>Initial Terminal Conditions:</u>

A single card is read to specify the initial terminal conditions for each generator.

10	20	30	40	
P	Q	V	δ	

P Generator MW output.

Q Generator MVAR output.

V Bus voltage in p.u.

δ Bus angle in degrees.

vi) Network Switching Time Parameter:

For each network condition a time card is read:

10	
TFIN	

TFIN Time at which the network condition about to be input terminates. The end of the simulation is indicated when a time card with TFIN = 0.0 is read.

vii) Network Terminal Admittance Matrix:

The complex elements of the terminal admittance matrix are entered row by row. Each card can contain four complex elements. Each row of the matrix is commenced on a new card.

10	20	30	40	50	60	70	80
Y_{REAL}	Y_{IMAG}	Y_{REAL}	Y_{IMAG}	Y_{REAL}	Y_{IMAG}	Y_{REAL}	Y_{IMAG}

viii) Graph Plotting Data:

A single card is read for each graph that is plotted:

40	50	60	
LABEL	MIN	MAX	

LABEL Up to 40 alphanumeric characters to name the plotted variable.

MIN Lower range of plot variable.

MAX Upper range of plot variable.

5.2 Program Preparation

The following steps are required in preparing the program source for a study.

i) The required subroutines for modelling the generators, excitation systems and turbine-governors are selected and included in the program.

ii) The statements which call the appropriate generator, excitation system and turbine-governor subroutines for modelling each generating unit in the system are inserted in the mainline program. These consist of statements which call the subroutines firstly to calculate the initial conditions e.g. CALL GEN1IC(2) and secondly, to calculate the state variable derivatives, e.g. CALL GEN1(2). With the example calling statements, subroutine GEN1 would be used for modelling generator number 2. The subroutine names for the different machine and controller representations were previously defined in section 4.

iii) Statements are inserted within the machine and controller subroutines to set the elements of PRTVAR(I,J) so that the desired variables are printed.

iv) Statements are inserted into the subroutine OUTPUT to set the elements of VAR(I,J) so that the desired variables are plotted.

6. EXAMPLE STUDY

6.1 Purpose of Study

This section illustrates the application of the program with a transient stability study of a simple one machine infinite bus system. This system has been previously studied[6,7] with the General Electric Simulation Program[2] and has been used as a benchmark for the debugging and testing of the developed program.

The description of the study also clarifies the instructions given in section 5 for preparing the program and the data. The program source listing which is given in section 11 is set up for the study which is described here. The program results which were obtained from the study are also included in this section. The example system and the results presented can be conveniently used for an initial check out of the program when it is set up on another computer.

6.2 Description of System

The configuration of the example system and its initial loadflow condition is shown in Figure 18. The generator is a round rotor machine. It has a rotating excitation system which is represented by the IEEE Type 1 model. The mechanical input to the generator is considered constant. The system disturbance is created by applying a three phase stub fault on the intermediate load bus for .066s.

The data input sheets for the study are shown in Figures 19a, 19b and 19c. The equivalent admittance matrices for the faulted and unfaulted transmission networks were obtained using the network reduction program described in appendix 8.4.

6.3 Simulation Results

The simulated responses of the generator rotor angle, terminal voltage, electric power and field voltage are shown in Figures 21, 22, 23 and 24 respectively. The results obtained with the developed program are compared with those obtained with the General Electric Simulation program and presented in reference 7. The validity of the developed program is demonstrated by the excellent agreement between the two sets of results.

The program printout consists of three sections:

i) Image printout of the input data.

ii) Numerical printout of variable responses.

iii) Graphical printout of variable responses.

Samples of these sections of program printout which were obtained in the example study are respectively shown in Figures 20a, 20b and 20c.

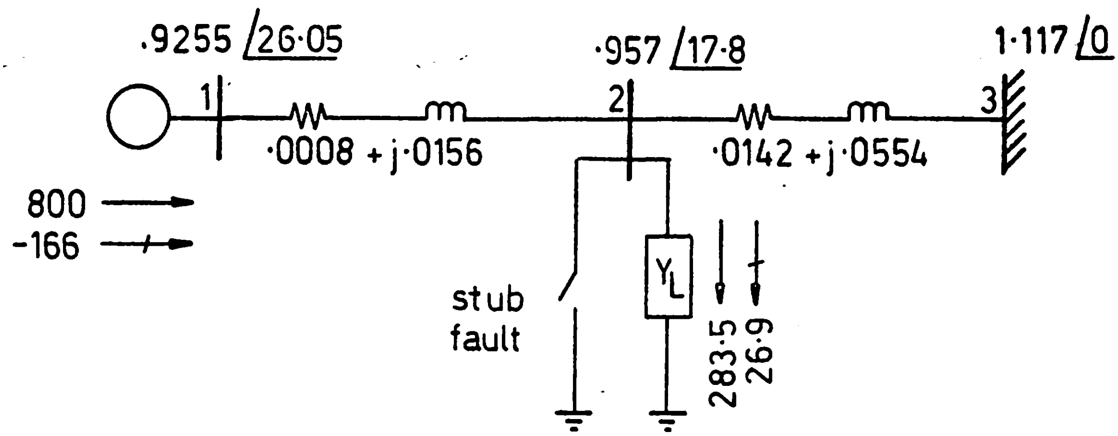


Figure 18: Example System Configuration and Initial Loadflow Condition

NO.								
	control parameters							
	INGEN	TSTEP	TPRINT					
1	2	.001	.025					
	generator parameters							
	P_{base}	H	R_a	X_l	X_d	X'_d	X_q	X'_q
	T'_{do}	T'_{qo}	D	A	B			
2	800.0	3.82	.0037	.188	1.75	.275	1.68	.47
3	5.2	1.965	.0	.00007978	7.1918			
4	10000.0	1000.0	.0	.0	.0001	.0001	.0001	.0001
5	1000.0	1000.0	.0	.0	.0			
	excitation system parameters							
	K_A	K_E	K_F	T_A	T_E	T_F	V_{RMIN}	V_{RMAX}
	A	B						
6	25.0	-.0445	.16	.06	0.5	1.0	-1.0	1.0
7	.0016	1.456						
8	b̸							
9	b̸							

Figure 19a: Data Cards for Example Study

NO.				
	turbine-governor parameters			
10	ø			
11	ø			
12	ø			
13	ø			
	terminal conditions			
	P	Q	V	δ
14	800.0	-166.0	.9255	26.05
15	-454.7	533.2	1.117	0.0
	time card			
	TFIN			
16	.066			
	admittance matrix - fault on			
17	3.2787	-63.9344	0.0	0.0
18	0.0	0.0	4.3414	-16.9377

Figure 19b: Data Cards for Example Study

```
NO.
    time card
       TFIN
19      3.0

    admittance matrix - fault off
20      4.7196  -13.8959   -2.3252   13.4735
21     -2.3252   13.4735    2.9890  -13.4568

    time card
       TFIN
22      0.0

    plotting parameters
           LABEL                                MIN       MAX
23  ROTOR ANGLE, DEG                          -20.0     180.0
24  TERM VOLTAGE, PU                            0.0       2.0
25  REAL POWER, PU                              0.0       2.0
26  FIELD VOLTAGE,  PU                          0.0       5.0
```

Figure 19c: Data Cards for Example Study

```
                         POWER  SYSTEM  RESEARCH  GROUP
                          UNIVERSITY  OF  SASKATCHEWAN
                   POWER  SYSTEM  DYNAMIC  SIMULATION  PROGRAM

NO. OF GENERATORS      2
TIME STEP          0.001
PRINT INTERVAL     0.025

GENERATOR PARAMETERS
   1   800.0         3.820       0.3700E-02  0.1880      1.750       0.2750      1.680       0.4700
       5.200         1.965       0.0000      0.7978E-04  7.192
   2  0.1000E 05     1000.       0.0000      0.0000      0.1000E-03  0.1000E-03  0.1000E-03  0.1000E-03
       1000.         1000.       0.0000      0.0000      0.0000

 EXCITATION SYSTEM PARAMETERS
   1   25.00       -0.4450E-01   0.1600      0.6000E-01  0.5000      1.000      -1.000       1.000
      0.1600E-02     1.465       0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
   2  0.0000         0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
      0.0000         0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000

 TURBINE-GOVERNOR PARAMETERS
   1  0.0000         0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
      0.0000         0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
   2  0.0000         0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000
      0.0000         0.0000      0.0000      0.0000      0.0000      0.0000      0.0000      0.0000

INITIAL GENERATOR TERMINAL CONDITIONS
 GEN          MW          MVAR        VOLTS       ANGLE
   1        800.000    -166.000     0.92550    26.05000
   2       -454.700     533.200     1.11700     0.00000

      TERMINAL ADMITTANCE MATRIX FROM    0.000 TO    0.066 SECS.
         3.2787   -63.9344     0.0000     0.0000
         0.0000     0.0000     4.3414   -16.9377
```

Figure 20a: Sample Printout for Example Study

SIMULATED RESPONSES

TIME SECS	GEN NO	ROTOR ANGLE	ROTOR SPEED	EQ' VOLTS	ED' VOLTS	TERM VOLTS	ELEC-POWER REAL	IMAG	FIELD VOLTS	MECH POWER	SATN FACTOR	AIR GAP VOLTS
0.000												
	1	97.797	0.0000	0.5546	0.6235	0.2181	0.0195	0.3802	1.9910	1.0045	0.9960	0.5462
	2	-0.000	0.0000	1.1170	0.0000	1.1170	0.0542	0.2113	1.1170	-0.0455	1.0000	1.1170
0.025												
	1	98.655	0.0032	0.5517	0.6002	0.2144	0.0188	0.3675	2.0388	1.0045	0.9962	0.5369
	2	-0.001	-0.0000	1.1170	0.0000	1.1170	0.0542	0.2113	1.1170	-0.0455	1.0000	1.1170
0.050												
	1	101.237	0.0064	0.5490	0.5779	0.2110	0.0182	0.3558	2.0903	1.0045	0.9964	0.5283
	2	-0.002	-0.0000	1.1170	0.0000	1.1170	0.0542	0.2113	1.1170	-0.0455	1.0000	1.1170

TERMINAL ADMITTANCE MATRIX FROM 0.066 TO 0.350 SECS.

4.7196	-13.8959	-2.3252	13.4735
-2.3252	13.4735	2.9890	-13.4568

TIME SECS	GEN NO	ROTOR ANGLE	ROTOR SPEED	EQ' VOLTS	ED' VOLTS	TERM VOLTS	ELEC-POWER REAL	IMAG	FIELD VOLTS	MECH POWER	SATN FACTOR	AIR GAP VOLTS
0.075												
	1	105.414	0.0083	0.5474	0.5650	0.8720	1.0661	-0.2252	2.1419	1.0045	0.9630	0.8590
	2	-0.003	-0.0000	1.1170	0.0000	1.1170	-0.0507	0.0673	1.1170	-0.0455	1.0000	1.1170
0.100												
	1	109.813	0.0080	0.5470	0.5664	0.8566	1.1149	-0.2117	2.1935	1.0045	0.9650	0.8507
	2	-0.005	-0.0000	1.1170	0.0000	1.1170	-0.0540	0.0735	1.1170	-0.0455	1.0000	1.1170
0.125												
	1	114.008	0.0075	0.5461	0.5674	0.8400	1.1542	-0.1981	2.2451	1.0045	0.9673	0.8412
	2	-0.007	-0.0000	1.1170	0.0000	1.1170	-0.0565	0.0797	1.1170	-0.0455	1.0000	1.1170
0.150												
	1	117.926	0.0070	0.5450	0.5680	0.8227	1.1841	-0.1848	2.2968	1.0045	0.9695	0.8309
	2	-0.008	-0.0000	1.1170	0.0000	1.1170	-0.0583	0.0858	1.1170	-0.0455	1.0000	1.1170
0.175												
	1	121.514	0.0063	0.5436	0.5682	0.8053	1.2053	-0.1722	2.3485	1.0045	0.9717	0.8203
	2	-0.010	-0.0000	1.1170	0.0000	1.1170	-0.0595	0.0914	1.1170	-0.0455	1.0000	1.1170
0.200												
	1	124.734	0.0056	0.5419	0.5680	0.7884	1.2190	-0.1609	2.4003	1.0045	0.9738	0.8096
	2	-0.011	-0.0000	1.1170	0.0000	1.1170	-0.0602	0.0966	1.1170	-0.0455	1.0000	1.1170
0.225												
	1	127.561	0.0049	0.5402	0.5673	0.7725	1.2264	-0.1510	2.4521	1.0045	0.9756	0.7994
	2	-0.012	-0.0000	1.1170	0.0000	1.1170	-0.0605	0.1012	1.1170	-0.0455	1.0000	1.1170
0.250												
	1	129.979	0.0041	0.5384	0.5663	0.7581	1.2292	-0.1428	2.5039	1.0045	0.9772	0.7899
	2	-0.014	-0.0000	1.1170	0.0000	1.1170	-0.0604	0.1051	1.1170	-0.0455	1.0000	1.1170
0.275												
	1	131.982	0.0033	0.5366	0.5650	0.7454	1.2288	-0.1363	2.5558	1.0045	0.9785	0.7813
	2	-0.015	-0.0000	1.1170	0.0000	1.1170	-0.0601	0.1084	1.1170	-0.0455	1.0000	1.1170
0.300												
	1	133.572	0.0026	0.5349	0.5634	0.7348	1.2262	-0.1317	2.6077	1.0045	0.9796	0.7738
	2	-0.016	-0.0000	1.1170	0.0000	1.1170	-0.0598	0.1110	1.1170	-0.0455	1.0000	1.1170
0.325												
	1	134.752	0.0018	0.5333	0.5617	0.7264	1.2225	-0.1288	2.6597	1.0045	0.9805	0.7677
	2	-0.017	-0.0000	1.1170	0.0000	1.1170	-0.0594	0.1129	1.1170	-0.0455	1.0000	1.1170
0.350												
	1	135.529	0.0011	0.5318	0.5598	0.7204	1.2185	-0.1276	2.7117	1.0045	0.9811	0.7630
	2	-0.018	-0.0000	1.1170	0.0000	1.1170	-0.0591	0.1142	1.1170	-0.0455	1.0000	1.1170

Figure 20b: Sample Printout for Example Study

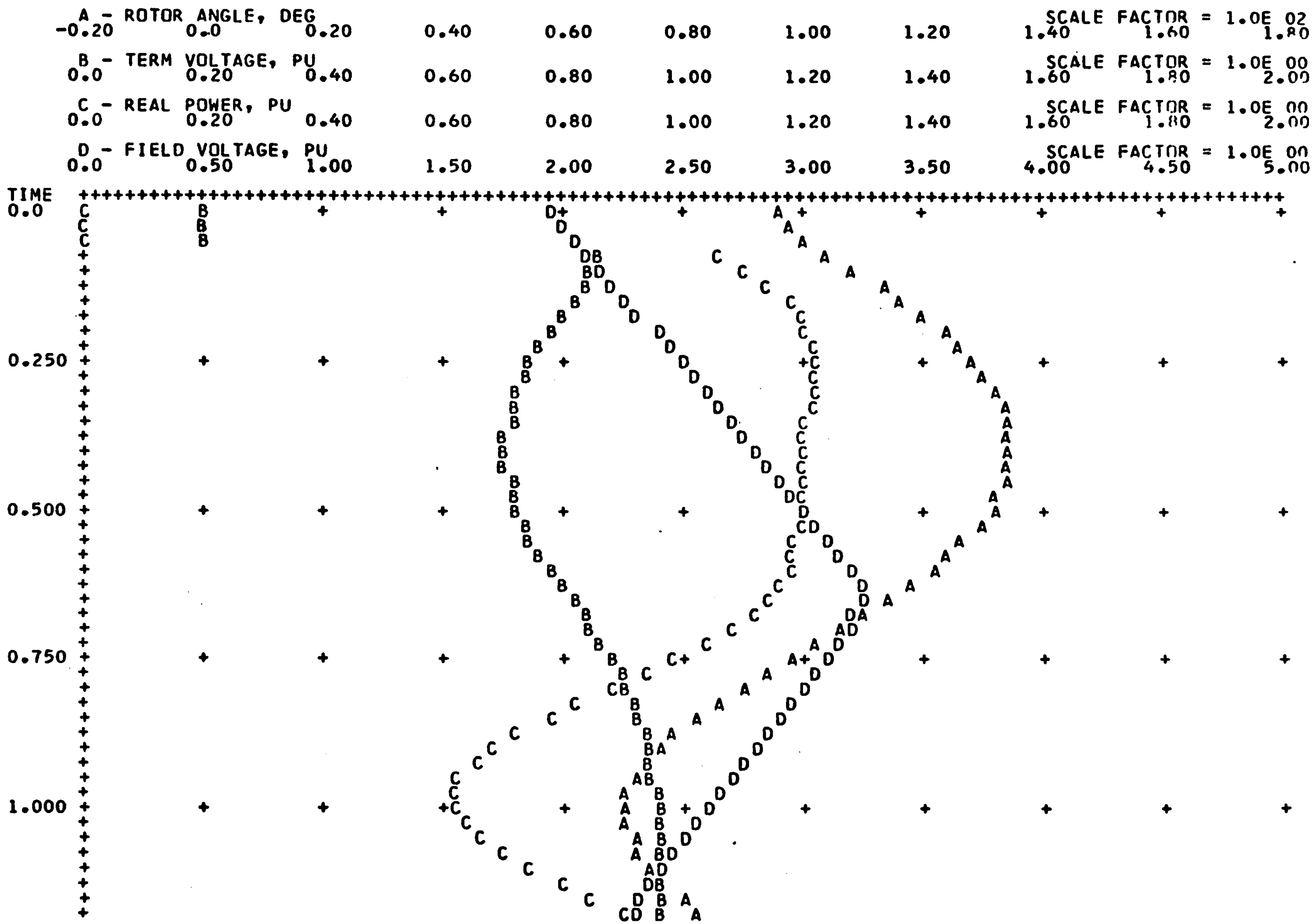


Figure 20c: Sample Printout for Example Study

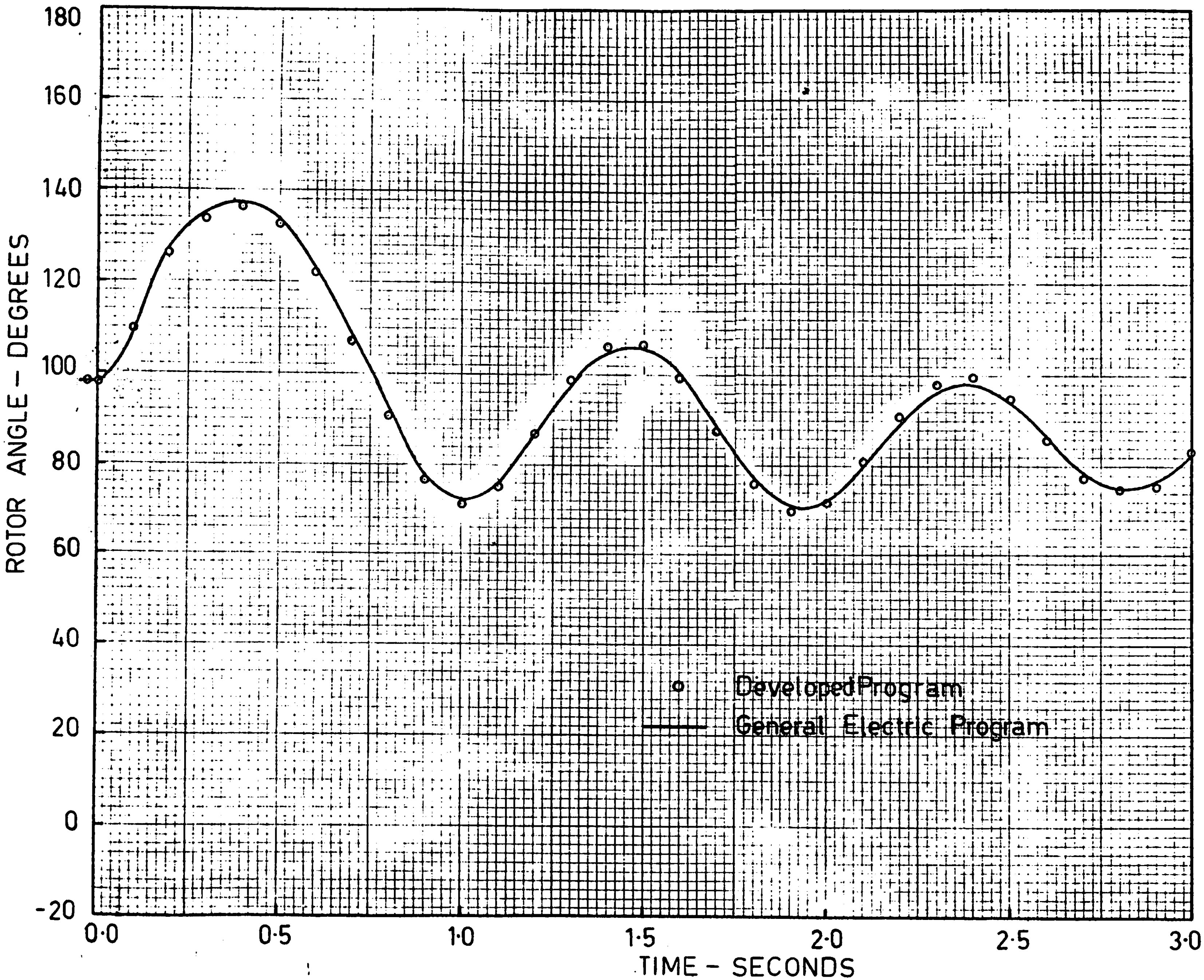


Figure 21: Comparison of Rotor Angle Responses

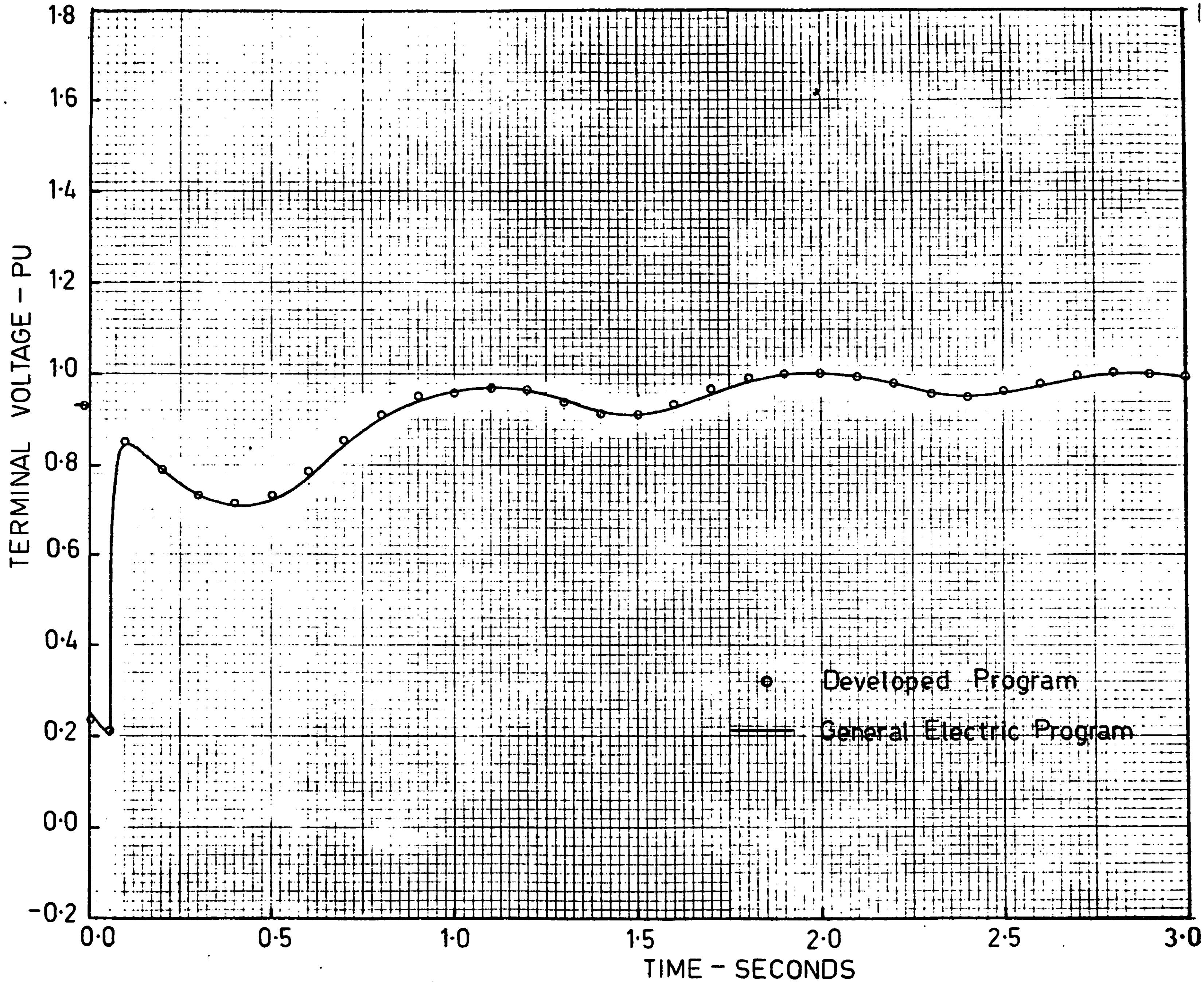


Figure 22: Comparison of Terminal Voltage Responses

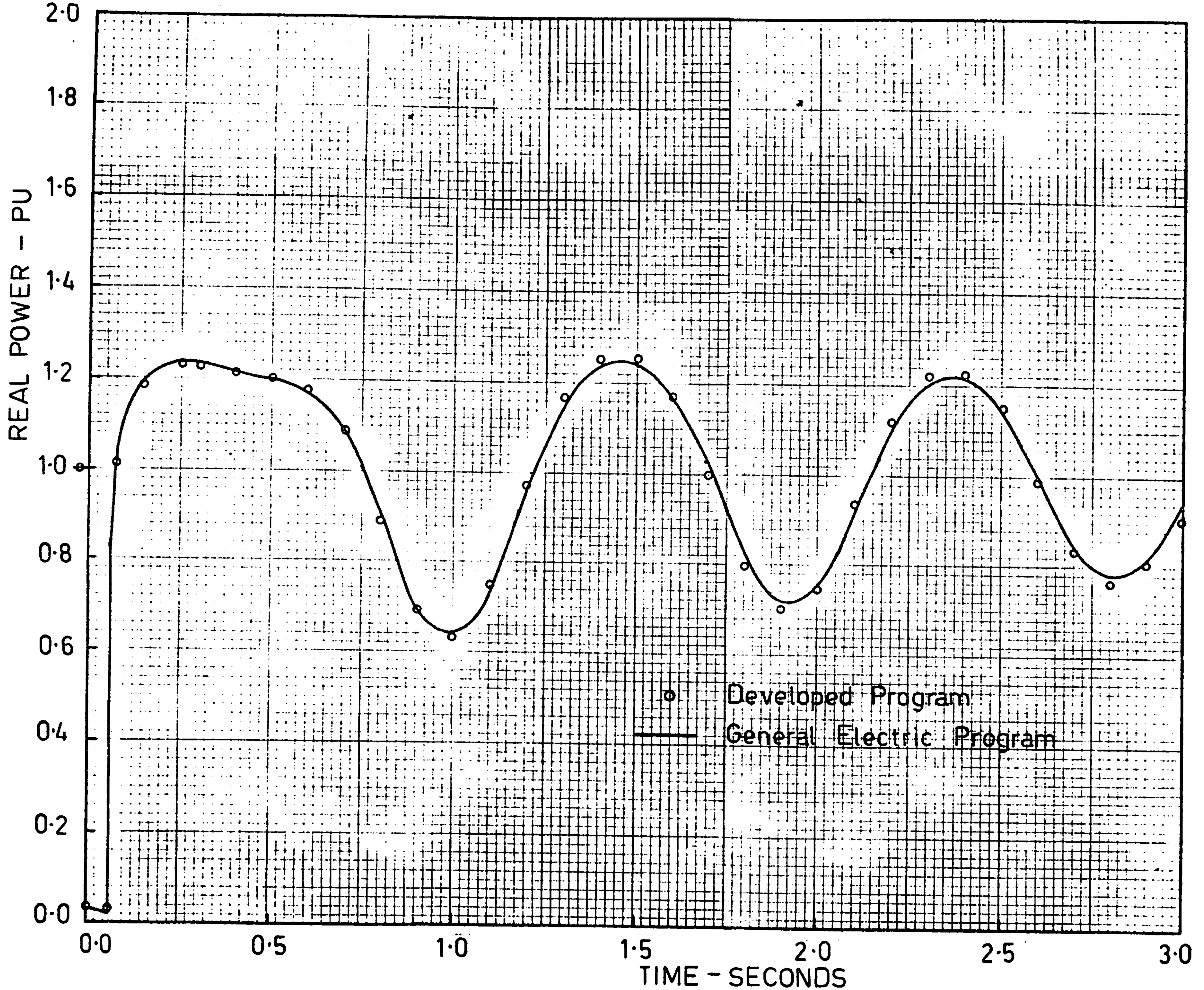


Figure 23: Comparison of Electric Power Responses

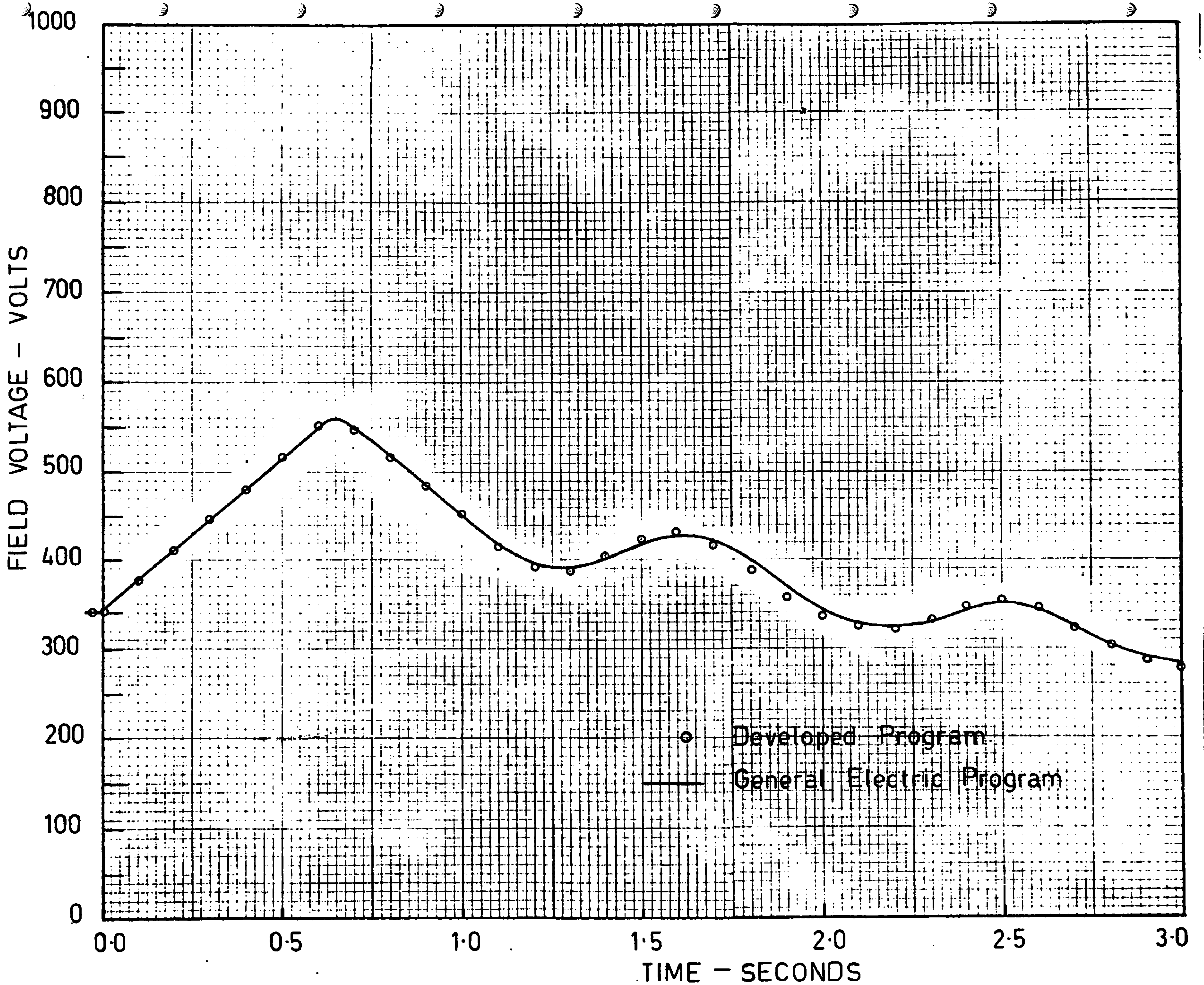


Figure 24: Comparison of Field Voltage Responses

7. PROGRAM SPECIFICATION AND PERFORMANCE DETAILS

Language: FORTRAN IV

Computer used: IBM 370/155

Core requirement: 32,000 bytes.

Capacity: 10 generators

Number of statements: 553

Input medium: cards.

Output medium: line printer

Work or Data files needed: none.

Execution Time: 13 secs. to simulate 1 sec. of real time on a 10 generator system using an integration step of .005s.

8. APPENDICES

8.1 Derivation of Synchronous Generator Equations

8.1.1 Park Equations

The Park equations of a synchronous generator with a d axis field winding and a q axis damper winding are summarized below:

Direct axis:

$$\psi_d = L_{ad}I_f - L_dI_d \tag{8.1}$$

$$\psi_f = L_{ff}I_f - L_{ad}I_d \tag{8.2}$$

$$V_d = \dot{\psi}_d - R_aI_d - \psi_q\omega \tag{8.3}$$

$$V_f = \dot{\psi}_f + R_fI_f \tag{8.4}$$

Quadrature axis:

$$\psi_q = L_{aq}I_{kq} - L_qI_q \tag{8.5}$$

$$\psi_{kq} = L_{kkq}I_{kq} - L_{aq}I_q \tag{8.6}$$

$$V_q = \dot{\psi}_q - R_aI_q + \psi_d\omega \tag{8.7}$$

$$0 = \dot{\psi}_{kq} + R_{kq}I_{kq} \tag{8.8}$$

8.1.2 Stator Equations

In this section the stator equations which are suited for use in the generator model are derived. For the purpose of transient and dynamic stability studies the stator transformer voltages can be neglected. Equations (8.3) and (8.7), therefore, become:

$$V_d = -R_aI_d - \omega\psi_q \tag{8.9}$$

$$V_q = -R_aI_q + \omega\psi_d \tag{8.10}$$

The q axis stator equation is firstly derived by eliminating I_f from (8.1) and (8.2),

$$\omega\psi_d = \omega\psi_fL_{ad}/L_{ff} - \omega(L_d - L_{ad}^2/Lff)I_d \tag{8.11}$$

Define tho d axis transient inductance,

$$L_d' = L_d - L_{ad}^2/L_{ff} \tag{8.12}$$

and the q axis transient voltage,

$$E_q' = \omega\psi_f L_{ad}/L_{ff} \tag{8.13}$$

and substitute into (8.11),

$$\omega\psi_d = E_q' - \omega L_d' I_d \tag{8.14}$$

Substitute (8.14) into (8.10),

$$V_q = E_q' - \omega L_d' I_d - R_a I_q \tag{8.15}$$

The d axis stator equation is now derived by firstly eliminating I_{kq} from (8.5) and (8.6),

$$\omega\psi_q = \omega\psi_{kq} L_{ad}/L_{kkq} - \omega(L_q - L_{aq}^2/L_{kkq}) I_q \tag{8.16}$$

Define the q axis transient inductance,

$$L_q' = L_q - L_{aq}^2/L_{kkq} \tag{8.17}$$

and the d axis transient voltage,

$$E_d' = -\omega\psi_{kq} L_{aq}/L_{kkq} \tag{8.18}$$

and substitute into (8.16)

$$\omega\psi_q = -E_d' - \omega L_q' I_q \tag{8.19}$$

Substitute (8.19) into (8.9),

$$V_d = E_d' + \omega L_q' I_q - R_a I_d \tag{8.20}$$

8.1.3 Rotor Winding Equations

The d axis field winding equation is derived as follows. Multiply (8.4) by $\omega L_{ad}/R_f$,

$$V_f \frac{\omega L_{ad}}{R_f} = \omega L_{ad} I_f + \frac{\omega L_{ad}}{R_f} \dot{\psi}_f \tag{8.21}$$

Define,

$$E_f = \frac{\omega L_{ad}^o}{R_f} V_f = \frac{\omega L_{ad}}{kR_f} V_f \tag{8.22}$$

From (8.2),

$$I_f = \psi_f/L_{ff} + L_{ad}I_d/L_{ff} \tag{8.23}$$

Substitute (8.22) and (8.23) into (8.21),

$$kE_f = \frac{\omega L_{ad}}{L_{ff}}\psi_f + \frac{\omega L_{ad}^2}{L_{ff}} I_d + \frac{\omega L_{ad}}{R_f}\dot{\psi}_f \tag{8.24}$$

Define the d axis open circuit transient time constant,

$$T_{do}' = L_{ff}/R_f \tag{8.25}$$

Combine (8.12), (8.13), (8.25) into (8.24),

$$kE_f = E_q' + \omega(L_d - L_d')I_d + T_{do}'\dot{E}_q' \tag{8.26}$$

The q axis damper equation is derived as follows.

Multiply (8.8) by $\omega L_{aq}/R_{kq}$,

$$0 = \omega L_{aq}I_{kq} + \frac{\omega L_{aq}}{R_{kq}}\dot{\psi}_{kq} \tag{8.27}$$

From (8.6),

$$I_{kq} = \psi_{kq}/L_{kkq} + L_{aq}I_q/L_{kkq} \tag{8.28}$$

Substitute (8.28) into (8.27),

$$0 = \frac{\omega L_{aq}}{L_{kkq}}\psi_{kq} + \frac{\omega L_{aq}^2}{L_{kkq}} I_q + \frac{\omega L_{aq}}{R_{kq}}\dot{\psi}_{kq} \tag{8.29}$$

Define the q axis open circuit transient time constant,

$$T_{qo}' = L_{kkq}/R_{kq} \tag{8.30}$$

Combine (8.18), (8.19) and (8.30) into (8.29),

$$0 = +E_d' \mp \omega(L_q - L_q')I_q + T_{qo}'\dot{E}_d' \tag{8.31}$$

8.1.4 Air Gap Voltage Equations

The d and q axis air gap voltages are defined by,

$$E_{aq} = \omega\psi_{ad} = \omega(\psi_d + L_\ell I_d) \tag{8.32}$$

and,

$$E_{ad} = -\omega\psi_{aq} = -\omega(\psi_q + L_\ell I_q) \tag{8.33}$$

The equations used in the generator model are obtained as follows.

Substitute (8.14) into (8.32),

$$E_{aq} = E_q' - \omega(L_d' - L_\ell)I_d \tag{8.34}$$

Substitute (8.20) into (8.33),

$$E_{ad} = E_d' + \omega(L_q' - L_\ell)I_q \tag{8.35}$$

In sections 8.1.1 to 8.1.4 the generator model equations have been developed in terms of inductances. For transient and dynamic stability studies the deviations in frequency are small and we are justified in substituting $\omega L = X$ (X is the reactance at synchronous frequency) in the equations derived.

8.1.5 Steady State Generator Equations

The vector diagram for a synchronous generator in the steady state is shown in Figure 25. The voltage $\overline{E}_{qd}$ is defined as:

$$\overline{E}_{qd} = (V_d + jV_q) + (R_a + jX_q)(I_d + jI_q) \tag{8.36}$$

During the steady state $\overline{E}_{qd}$ lies on the quadrature axis since,

$$R(\overline{E}_{qd}) = V_d + R_a I_d - X_q I_q \tag{8.37}$$

and from (8.5) and (8.9),

$$V_d + R_a I_d - X_q I_q = -X_{aq} I_{kq} \tag{8.38}$$

In the steady state, $I_{kq} = 0$ and, therefore, $R(\overline{E}_{qd}) = 0$. This property of $\overline{E}_{qd}$ is used in calculating the generator initial conditions since the terminal voltage and current are given in terms of the synchronous reference frame. The position of the generator q axis is given by calculating,

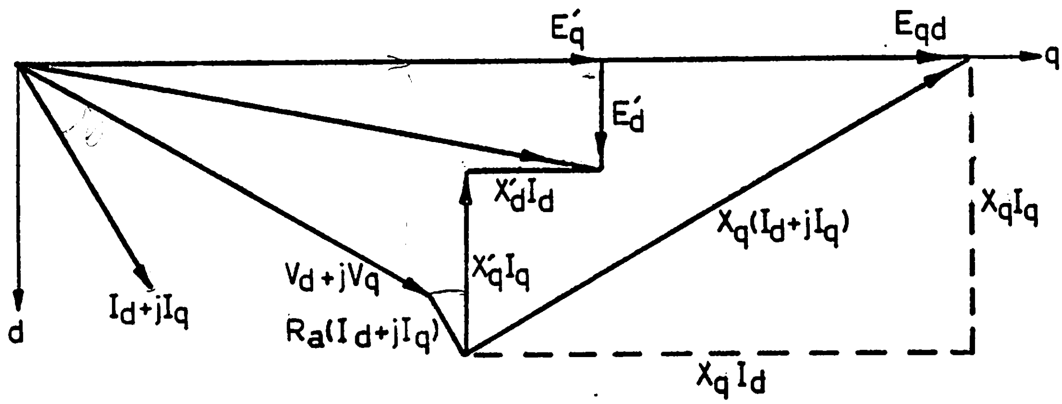


Figure 25: Steady State Synchronous Generator Vector Diagram

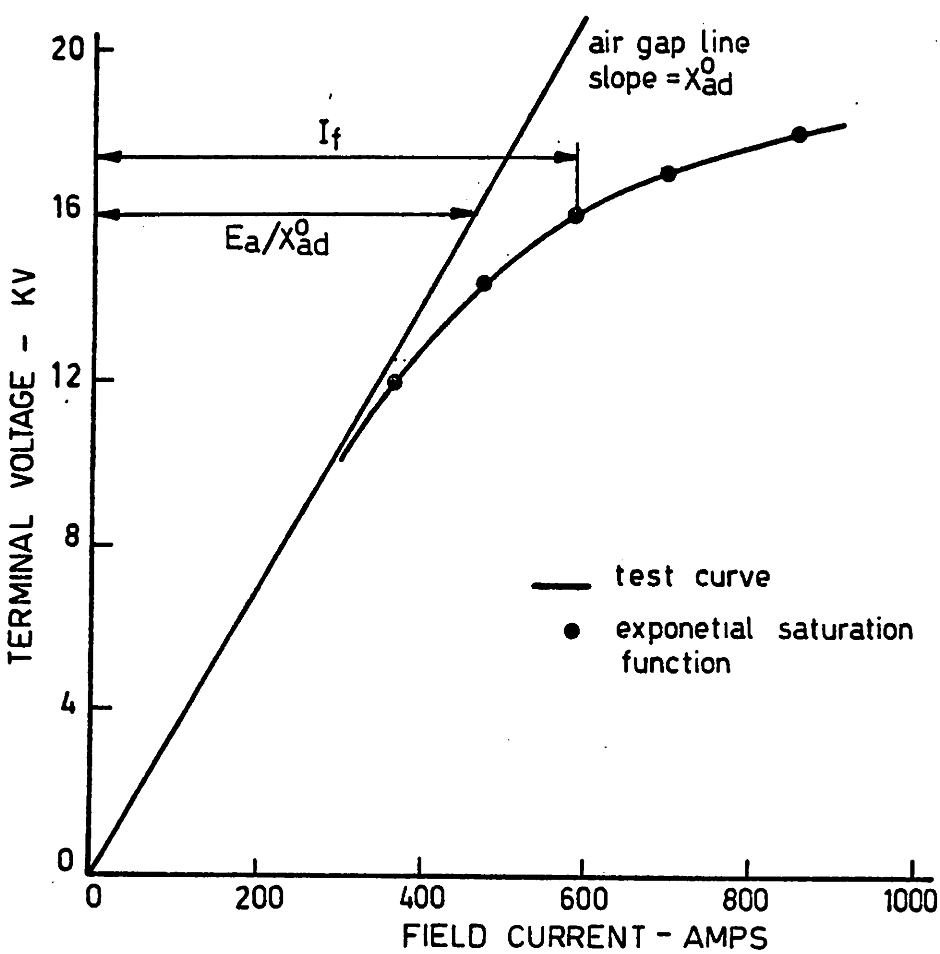


Figure 26: Open Circuit Saturation Curve for Squaw Rapids Generators on Saskatchewan Power System

$$\overline{E}_{qd} = \overline{V} + (R_a + jX_q)\overline{I} \tag{8.39}$$

and,

$$\delta = \mathrm{Arg}(\overline{E}_{qd}) \tag{8.40}$$

From the vector diagram it can also be seen that in the steady state,

$$E_q' = \mathrm{Mag}(\overline{E}_{qd}) - (X_q - X_d')I_d \tag{8.41}$$

Two other equations which are used in calculating the generator initial conditions are obtained by setting derivatives to zero in (8.26) and (8.31),

$$E_f = (E_q' + (X_d - X_d')I_d)/k \tag{8.42}$$

$$E_d' = (X_q - X_q')I_q \tag{8.43}$$

8.1.6 Saturated Generator Reactances and Time Constants

This section derives the expressions for the saturated values of generator reactances and time constants in terms of the unsaturated values. The generator mutual reactances vary with the saturation factor according to,

$$X_{ad} = kX_{ad}^{o} \tag{8.44}$$

$$X_{aq} = kX_{aq}^{o} \tag{8.45}$$

Expressions for the saturated synchronous reactances are derived as follows,

$$X_d = X_{ad} + X_\ell \tag{8.46}$$

and,

$$X_d^o = X_{ad}^o + X_\ell \tag{8.48}$$

Substituting (8.44) and (8.47) into (8.46),

$$X_d = kX_d^o + (1 - k)X_\ell \tag{8.48}$$

Similarly, it follows that,

$$X_q = kX_q^o + (1 - k)X_\ell \tag{8.49}$$

Expressions for the saturated open circuit transient time constants are derived as follows. From the definition of $T_{do}^{'}$ in (8.25),

$$T_{do}^{'}/T_{do}^{'o} = X_{ff}/X_{ff}^o \tag{8.50}$$

Now,

$$X_{ff}^o = X_{ad}^o + X_{f\ell} \tag{8.51}$$

and,

$$X_{ff} = X_{ad} + X_{f\ell} \tag{8.52}$$

Substituting (8.44) and (8.51) into (8.52),

$$X_{ff} = X_{ff}^o \left(k + \frac{X_{f\ell}}{X_{ff}^o} (1 - k)\right) \tag{8.53}$$

From definition of $X_d^{'}$ in (8.12),

$$X_{ff}^o = X_{ad}^{o2}/(X_d^o - X_d^{'}) \tag{8.54}$$

Substituting (8.51) and (8.55) into (8.53),

$$X_{ff} = X_{ff}^o\left(k + \frac{X_d^o - X_d^{'}}{X_d^o - X_\ell} (1 - k)\right) \tag{8.55}$$

and from (8.50),

$$T_{do}^{'} = T_{do}^{'o} \left(k + \frac{X_d^o - X_d^{'}}{X_d^o - X_\ell} (1 - k)\right) \tag{8.56}$$

Similarly,

$$T_{qo}^{'} = T_{qo}^{'o} \left(k + \frac{X_q^o - X_q^{'}}{X_q^o - X_\ell} (1 - k)\right) \tag{8.57}$$

8.2 <u>Derivation of Iterative Equation for Transient Saliency</u>

From (2.4) the generator armature equations may be arranged as,

$$\begin{bmatrix} I_d \\ I_q \end{bmatrix} = \frac{1}{R_a^2 + X_q'X_d'} \begin{bmatrix} R_a & X_q' \\ -X_d' & R_a \end{bmatrix} \begin{bmatrix} E_d'-V_d \\ E_q'-V_q \end{bmatrix} \qquad (8.58)$$

Equation (8.~~4~~58) can be expressed as a single complex equation,

$$I_d + jI_q = \frac{1}{R_a^2 + X_q'X_d'} [R_a(E_d' + jE_q') - R_a(V_d + jV_q) + X_q'(E_q' - V_q) - jX_d'(E_d' - V_d)] \qquad (8.59)$$

From (2.10) the current produced by the generator equivalent circuit in Figure 2 is,

$$I_d + jI_q = \frac{R_a - j\frac{1}{2}(X_d' + X_q')}{R_a^2 + X_d'X_q'} [(E_d^{fict} + jE_q^{fict}) - (V_d + jV_q)] \qquad (8.60)$$

By equating the two expressions for generator current in (8.59) and (8.60), we obtain,

$$E_d^{fict} + jE_q^{fict} = E_d' + jE_q' + \frac{\frac{j}{2}(X_q' - X_d')}{R_a - \frac{j}{2}(X_d' + jX_q')} [(E_d' - jE_q') - (V_d - jV_q)] \qquad (8.61)$$

Now,

$$E_d^{fict} + jE_q^{fict} = \bar{E}^{fict} e^{-j\theta} \qquad (8.62)$$

and, therefore, the iterative equation for calculating $\bar{E}^{fict}$ is,

$$\bar{E}^{fict} = \bar{E}' + \frac{\frac{j}{2}(X_q' - X_d')}{R_a - \frac{j}{2}(X_d' + jX_q')} [\bar{E}'^* - \bar{V}^*] e^{j2\theta} \qquad (8.63)$$

8.3 Mathematical Representation of Open Circuit Saturation Curve

This appendix describes the mathematical function which is used for representing the generator saturation curves. The function is used for representing saturation effects in the synchronous generator and in the rotating exciter in the IEEE type 1 excitation system model. Consider the typical open circuit saturation curve shown in Figure 26. The saturation function is defined by:

$$S(E_a) = \frac{I_f}{E_a/X_{ad}} - 1 \tag{8.64}$$

We assume S to be mathematically expressed as an exponential function of E_a,

$$S(E_a) = A\, e^{B.E_a} \tag{8.65}$$

A and B are constants which can be calculated given two points on the open circuit saturation curve. Consider that points S_1, E_{a1} and S_2, E_{a2} are known. Then,

$$B = \ln(S_1/S_2)/(E_{a1} - E_{a2}) \tag{8.66}$$

and,

$$A = S_1/e^{B.E_{a1}} \tag{8.67}$$

The application of the saturation function is illustrated by representing the open circuit saturation test curve shown in Figure 26. The saturation constants A and B have been calculated using the test points at 14.4 and 18.0 kV. and the mathematical saturation curve is constructed using,

$$I_f = E_a(1 + Ae^{B.E_a})/X^o_{ad} \tag{8.68}$$

Figure 26 shows that the mathematical representation duplicates the test curve very closely.

8.4 Network Reduction Program Manual

8.4.1 Summary

This appendix describes the specifications of a Power System Network Reduction Program and gives instructions for its use. The program is intended primarily for use in conjunction with the Power System Dynamic Simulation Program described in the main body of this report.

8.4.2 Program Function

The function of the program is to determine the matrix of driving point and transfer admittances of a given network as seen from the terminals of a selected group of generator buses. The system MW and MVAR loads are converted to constant admittances using the bus voltages obtained from a previous loadflow. The program allows for a three phase short circuit to be placed on any bus in the system. Faults with finite impedance are represented as shunt admittances.

8.4.3 Procedure

The program procedure is described below:

i) Input bus, line and transformer data.

ii) Convert bus loads to admittances.

iii) Form bus admittance matrix.

iv) Eliminate buses which are not specified as generator buses.

v) Calculate complex powers on generator buses to check reduction:

$$S = \overline{V}\ \overline{Y}_{TT}\ \overline{V}$$

vi) Printout generator powers and the equivalent terminal admittance matrix.

vii) If requested output admittance matrix on punched cards.

8.4.4 Program Specification and Performance

Language: FORTRAN IV

Computer used: IBM 370/155

Core requirement: 24,584 Bytes

Capacity: 40 buses, 10 generators

Number of Statements: 109

Input medium: cards

Output medium: line printer and card punch

Work of data files needed: none

Execution Time: 2.3 secs for a network with 39 buses and 10 generators.

8.4.5 Preparation of Data

The program input data is divided into the following items:

i) Control parameters.

ii) Bus parameters.

iii) Line parameters.

iv) Transformer parameters.

The data items are entered in the order listed. The data parameters are defined and their card format is illustrated for each item in turn. A sample data sheet for the example system which was described in section 6 is also shown in Figure 27.

i) Control parameters

5	10	15	20	
NBUS	NGEN	NFLT	NPUN	

NBUS Number of buses.

NGEN Number of generator buses.

NFLT Number of bus on which three phase short circuit is placed. If no short circuit is present NFLT = 0.

NPUN Punched card output of equivalent admittance matrix is enabled if NPUN = 1 and is suppressed if NPUN = 0.

ii) Bus parameters

5	17	18	24	30	36	42	48	54	66	72	78	
BUS			MAG	ANGLE	PL	QL	PG	QG		G	B	

BUS Bus number in range 1-40. Buses in network must be number consecutively.

TYPE TYPE = 0 for load bus, TYPE = 1 for generator bus.

MAG Magnitude of bus voltage in pu.

ANGLE Angle of bus voltage in degrees.

PL MW load.

QL MVAR load.

PG MW generation.

QG MVAR generation.

G Shunt conductance.

B Shunt susceptance (+ve for capacitance).

The end of the bus data cards is signified by a card with BUS = 0.

iii) Line cards

5	10	20	30	40	
FROM BUS	TO BUS	R	X	B	

FROM BUS Bus number.

TO BUS Bus number.

R Line resistance in pu.

X Line reactance in pu.

B Total line charging susceptance in pu.

The end of the line data is signified by a card with FROM BUS = 0.

iv) Transformer cards

5	10	20	30	36	
FROM BUS	TO BUS	R	X	TAP	

FROM BUS Bus number

TO BUS Bus number

R Transformer resistance in pu.

X Transformer reactance in pu.

TAP Tap ratio. The tap must be on the FROM BUS side of the transformer.

The end of the transformer data is signified by a card with FROM BUS = 0.

The program calculates the equivalent admittance matrix for one network condition at a time and does not allow for change cases. If equivalent admittances are required for different network configurations the card data must be modified and the program re-executed for each configuration.

8.4.6 Program Output

The program output consists of the following items.

i) Image printout of the input data.

ii) Printout of calculated generator MW and MVAR outputs. When a prefault network is reduced these outputs should check with the prefault loadflow generation.

iii) Printout of equivalent admittance matrix.

iv) If requested, punched card output of the equivalent admittance matrix in the format accepted by the Power System Dynamic Simulation Program.

A sample of the program printout which was obtained for the example study described in section 6 is shown in Figure 28.

control parameters

NO	NBUS	NGEN	NFLT	NPUN
1	3	2	0	0

bus parameters

NO	BUS	TYPE	MAG	ANGLE	PL	QL	PG	QG	G	B
2	1	1	.9255	26.05			800.0	-166.0	0.0	0.0
3	2	0	.9570	17.80	283.5	26.9			0.0	0.0
4	3	1	1.1170	0.0					0.0	0.0
5	0							-99999		

line parameters

NO	FROM BUS	TO BUS	R	X	B	TAP
6	1	2	.0008	.0150	0.0	
7	2	3	.0142	.0554	0.0	
8	0					

transformer parameters

NO	FROM BUS	TO BUS	R	X	TAP
9	0				

Figure 27: Sample Data Cards for Network Reduction Program

NETWORK REDUCTION PROGRAM

```
NO. OF BUSES          3
NO. OF GENERATORS     2
FAULT BUS             0
PUNCH FLAG            0
```

BUS DATA

BUS	TYPE	VOLTS	ANGLE	LOAD MW	LOAD MVAR	GEN MW	GEN MVAR	SHUNT G	SHUNT B
1	1	0.9255	26.050	0.00	0.00	800.00	-166.00	0.0000	0.0000
2	0	0.9570	17.800	283.50	26.90	0.00	0.00	0.0000	0.0000
3	1	1.1170	0.000	0.00	0.00	0.00	0.00	0.0000	0.0000

LINE DATA

BUS	BUS	RESISTANCE	REACTANCE	SUSCEPTANCE
1	2	0.00080	0.01560	0.00000
2	3	0.01420	0.05540	0.00000

TRANSFORMER DATA

BUS	BUS	RESISTANCE	REACTANCE	TAP

COMPUTED GENERATION

BUS	MW	MVAR
1	799.99	-166.67
3	-454.70	533.19

EQUIVALENT GENERATOR BUS ADMITTANCE MATRIX

4.7196	-13.8959	-2.3252	13.4735
-2.3252	13.4735	2.9890	-13.4568

Figure 28: Sample Printout from Network Reduction Program

9. NOMENCLATURE

The following notation is generally used unless a specific alternative definition is made within the text.

y	state variables
z	auxiliary variables
E	generator internal voltage
I	generator current
V	generator terminal voltage
θ	angle of generator d axis to synchronous reference
δ	angle of generator q axis to synchronous reference
X	reactance
L	inductance
ω	rotor speed deviation
Y	admittance
R	resistance
T	time constant
D	damping factor
A,B	coefficients of generator saturation function
ψ	flux
S	saturation function
k	saturation factor
H	inertia constant
f	frequency

Superscripts

$-$	complex
.	time derivative
'	transient
o	unsaturated

fict fictitious

Subscripts

d	direct axis
q	quadrature axis
real	real axis of synchronous reference
imag	imaginary axis of synchronous reference
a	armature or air gap
o	open circuit or base value
k	damper winding
f	field winding
ℓ	leakage
e	electrical or exciter
m	mechanical
G	generator internal bus
T	generator terminal bus
1	loss

10. REFERENCES

1. D.N. Ewart, F.P. deMello, "FACE - A Digital Dynamic Analysis Program", Proc. 1967 PICA Conference, pp. 83-94.

2. D.N. Ewart, R.P. Schulz, "FACE Multi-machine Power System Simulator Program", Proc. 1969 PICA Conference, pp. 133-153.

3. H.W. Dommel, N. Sato, "Fast Transient Stability Solutions", IEEE PAS-91, pp. 1643-1650.

4. IEEE Committee Report, "Computer Representation of Excitation Systems", IEEE PAS-87, June 1968.

5. IEEE Committee Report, "Dynamic Models for Steam and Hydro Turbines in System Studies", IEEE PAS-92, pp. 1904-1915.

6. P.L. Dandeno, R.L. Hauth, R.P. Schulz, "Effects of Synchronous Machine Modelling in Large Scale System Studies", IEEE PAS-92, March 1973, pp. 574-582.

7. R.L. Hauth, R.P. Schulz, "Effects of Synchronous Machine Modelling in Large Scale System Studies", final report for Northeast Power Co-ordinating Council (NPCC) Task Force on System Studies, System Dynamic Techniques Working Group, Dec. 1, 1971.

8. R. Podmore, "Loadflow Program Users Manual", Elec. Eng. Departmental Memo, Univ. of Canterbury, New Zealand, 1971.

11. PROGRAM SOURCE LISTINGS

```
C   ******************************************************************
C   *                                                                *
C   *           POWER SYSTEM DYNAMIC SIMULATION PROGRAM              *
C   *                                                                *
C   ******************************************************************
C
C   MAINLINE ROUTINE.
      COMPLEX VT,CT,Y,YFICT
      COMPLEX CMPLX,CONJG
      COMMON /BLOCK1/ TIME,TSTEP
      COMMON /BLOCK2/ PBASE(10),H(10),R(10),XL(10),XD(10),XD1(10),
     1XQ(10),XQ1(10),TD1(10),TQ1(10),DAMP(10),C1(10),C2(10)
      COMMON /BLOCK3/ AVRPRM(10,16)
      COMMON /BLOCK4/ TURPRM(10,16)
      COMMON/BLOCK5/ VT(10),CT(10),EF(10),PM(10)
      COMMON/BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      COMMON/BLOCK7/ Y(11,11)
      COMMON /BLOCK8/ TYM(200),VAR(200,6),NT,NVAR
      COMMON /BLOCK9/ PRTVAR(10,20)
      NT=0
      TIME=0.0
      TFIN=0.0
      NSTEP=0
      NPRINT=0
C   CLEAR INTEGRATOR ARRAYS
      DO 10 I=1,10
      DO 10 J=1,16
      PLUG(I,J)=0.0
      OUT(I,J)=0.0
      SAVE(I,J)=0.0
10    CONTINUE
      DO 12 I=1,10
      DO 12 J=1,20
12    PRTVAR(I,J)=0.0
      WRITE(6,990)
990   FORMAT('1',T39,'POWER  SYSTEM  RESEARCH  GROUP'//
     1T40,'UNIVERSITY  OF  SASKATCHEWAN'//
     2T32,'POWER  SYSTEM  DYNAMIC  SIMULATION  PROGRAM'//)
      READ(5,1000) NGEN,TSTEP,TPRINT
1000  FORMAT(I5,5X,2F10.4)
      WRITE(6,1005) NGEN,TSTEP,TPRINT
1005  FORMAT('0NO. OF GENERATORS',T20,I5/' TIME STEP',T20,F6.3/
     1' PRINT INTERVAL',T20,F6.3)
      WRITE(6,1008)
1008  FORMAT('0GENERATOR PARAMETERS')
C   READ GENERATOR PARAMETERS.
      DO 20 I=1,NGEN
      READ(5,1010)          PBASE(I),H(I),R(I),XL(I),XD(I),
     1XD1(I),XQ(I),XQ1(I),TD1(I),TQ1(I),DAMP(I),C1(I),C2(I)
1010  FORMAT(8F10.4)
      WRITE(6,1015) I,      PBASE(I),H(I),R(I),XL(I),XD(I),
     1XD1(I),XQ(I),XQ1(I),TD1(I),TQ1(I),DAMP(I),C1(I),C2(I)
1015  FORMAT(1X,I5,8G12.4/6X,8G12.4)
C     CONVERT DATA TO 100 MW BASE.
      C=100.0/PBASE(I)
      H(I)=H(I)/C
      R(I)=R(I)*C
      XL(I)=XL(I)*C
      XD(I)=XD(I)*C
      XD1(I)=XD1(I)*C
      XQ(I)=XQ(I)*C
      XQ1(I)=XQ1(I)*C
      DAMP(I)=DAMP(I)/C
20    CONTINUE
      WRITE(6,1018)
1018  FORMAT('0 EXCITATION SYSTEM PARAMETERS')
C   READ EXCITATION SYSTEM PARAMETERS.
      DO 30 I=1,NGEN
      READ(5,1020) (AVRPRM(I,J),J=1,16)
1020  FORMAT(8F10.4)
      WRITE(6,1025) I,(AVRPRM(I,J),J=1,16)
1025  FORMAT(1X,I5,8G12.4/6X,8G12.4)
30    CONTINUE
      WRITE(6,1028)
1028  FORMAT('0 TURBINE-GOVERNOR PARAMETERS')
C   READ TURBINE AND GOVERNOR PARAMETERS.
      DO 40 I=1,NGEN
      READ(5,1030) (TURPRM(I,J),J=1,16)
1030  FORMAT(8F10.4)
      WRITE(6,1035) I,(TURPRM(I,J),J=1,16)
1035  FORMAT(1X,I5,8G12.4/6X,8G12.4)
40    CONTINUE
      WRITE(6,1038)
1038  FORMAT('0INITIAL GENERATOR TERMINAL CONDITIONS'/
     1T3,'GEN',T15,'MW',T26,'MVAR',T36,'VOLTS',T46,'ANGLE')
C   READ CONDITIONS ON TERMINAL BUSES.
      DO 50 I=1,NGEN
      READ(5,1040)    PT,QT,VMAG,VARG
```

```
1040  FORMAT(        2P2F10.4,0P2F10.4)
      WRITE(6,1045) I,PT,QT,VMAG,VARG
1045  FORMAT(1X,I5,5X,2P2F10.3,0P2F10.5)
      VARG=VARG*3.1416/180.0
      VT(I)=VMAG*CMPLX(COS(VARG),SIN(VARG))
      CT(I)=CONJG(CMPLX(PT,QT)/VT(I))
50    CONTINUE
C  CALL EQUIPMENT SUBROUTINES TO CALCULATE INITIAL CONDITIONS.
C  CALL STATEMENTS MUST BE GENERATED BY USER.
C  *****************************************************************
      CALL GEN1IC(1)
      CALL GEN1IC(2)
      CALL AVR1IC(1)
C  *****************************************************************
C  LOOP HERE FOR EACH NEW NETWORK CONDITION.
70    CONTINUE
      TOLD=TFIN
C  READ THE CONTROL CARD.
      READ(5,1050) TFIN
1050  FORMAT(F10.4)
      IF(TFIN .EQ. 0.0) GO TO 150
      WRITE(6,1055) TOLD,TFIN
1055  FORMAT('0',T9,'TERMINAL ADMITTANCE MATRIX FROM',F8.3,' TO',F8.3,
     1' SECS.')
C  READ THE NEW ADMITTANCE MATRIX.
      DO 72 I=1,NGEN
      READ(5,1060) (Y(I,J),J=1,NGEN)
1060  FORMAT(8F10.4)
      WRITE(6,1065) (Y(I,J),J=1,NGEN)
1065  FORMAT((T9 ,8F10.4))
72    CONTINUE
      CALL MATRIX(NGEN)
      WRITE(6,1076)
1076  FORMAT('0')
C  LOOP HERE FOR EACH INTEGRATION STEP.
100   CONTINUE
C  SOLVE THE NETWORK.
      CALL NWSOL(NGEN)
C  CALL EQUIPMENT SUBROUTINES TO CALCULATE STATE VARIABLE DERIVATIVES.
C  CALL STATEMENTS MUST BE GENERATED BY USER.
C  *****************************************************************
      CALL AVR1(1)
      CALL GEN1(1)
      CALL GEN1(2)
C  *****************************************************************
C  CHECK FOR OUTPUT.
      IF(NSTEP .EQ. 0) CALL OUTPUT(NGEN)
      IF(NPRINT*TSTEP .LT. TPRINT-.0001) GO TO 125
      CALL OUTPUT(NGEN)
      NPRINT=0
125   CONTINUE
C  PERFORM INTEGRATION STEP.
      CALL INT(NGEN)
      NSTEP=NSTEP+1
      TIME=NSTEP*TSTEP
      NPRINT=NPRINT+1
C  CHECK FOR NEW NETWORK CONDITION.
      IF(TIME .LT. TFIN) GO TO 100
C  LOOP BACK FOR NEW NETWORK CONDITION.
      GO TO 70
C  COME HERE WHEN RUN IS COMPLETED.
150   CONTINUE
      CALL PLOT
      STOP
      END


C
C  SUBROUTINE TO CALCULATE EQUIVALENT Y MATRIX FOR INTERNAL
C  GENERATOR BUSES.
      SUBROUTINE MATRIX(NGEN)
      COMMON /BLOCK2/ PBASE(10),H(10),R(10),XL(10),XD(10),XD1(10),
     1XQ(10),XQ1(10),TD1(10),TQ1(10),DAMP(10),C1(10),C2(10)
      COMMON/BLOCK7/ Y(11,11)
      COMPLEX Y,YFICT,CMPLX,CONJG
C  AUGMENT Y MATRIX WITH GENERATOR BUSES AND ELIMINATE THE
C  TERMINAL BUSES.
      N1=NGEN+1
      DO 80 I=1,NGEN
C  MOVE TERMINAL BUS TO OUTSIDE OF MATRIX.
      Y(N1,N1)=Y(I,I)
      Y(I,I)=(0.0,0.0)
      DO 75 J=1,NGEN
C  MOVE ROW
      Y(N1,J)=Y(I,J)
      Y(I,J)=(0.0,0.0)
C  MOVE COLUMN
      Y(J,N1)=Y(J,I)
```

```
      Y(J,I)=(0.0,0.0)
75    CONTINUE
C     ADD IN GENERATOR BUS
      YFICT=CMPLX(R(I),-(XD1(I)+XQ1(I))/2.0)/(R(I)*R(I)+XD1(I)*XQ1(I))
      Y(I,I)=YFICT
C     CHECK IF TERMINAL BUS IS GROUNDED.
      IF(CABS(Y(N1,N1)) .EQ. 0.0) GO TO 80
      Y(N1,N1)=Y(N1,N1)+YFICT
      Y(I,N1)=-YFICT
      Y(N1,I)=-YFICT
C     ELIMINATE THE TERMINAL BUS.
      DO 76 M=1,NGEN
      DO 76 N=M,NGEN
      Y(M,N)=Y(M,N)-Y(M,N1)*Y(N1,N)/Y(N1,N1)
      Y(N,M)=Y(M,N)
76    CONTINUE
80    CONTINUE
      RETURN
      END


C
C     SUBROUTINE TO SOLVE NETWORK AND ARMATURE EQUATIONS.
      SUBROUTINE NWSOL(NGEN)
      COMPLEX SCALE,ROTATE
      REAL ID,IQ
      COMPLEX CMPLX,CONJG,Y,VT,CT
      COMPLEX VOLD,YFICT
      COMMON /BLOCK2/ PBASE(10),H(10),R(10),XL(10),XD(10),XD1(10),
     1XQ(10),XQ1(10),TD1(10),TQ1(10),DAMP(10),C1(10),C2(10)
      COMMON/BLOCK5/ VT(10),CT(10),EF(10),PM(10)
      COMMON/BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      COMMON /BLOCK7/ Y(11,11)
      COMPLEX EFICT(10),E(10)
       REAL DEL(10)
      DO 10 I=1,NGEN
      DEL(I)=OUT(I,2)
      EQ=OUT(I,3)
      ED=OUT(I,4)
C     TRANSFORM VOLTAGE TO SYNCHRONOUS REFERENCE.
      THETA=DEL(I)-3.1416/2.0
      E(I)=CMPLX(ED,EQ)*CMPLX(COS(THETA ),SIN(THETA ))
10    CONTINUE
      ITER=0
C     LOOP HERE FOR EACH ITERATION.
15    CONTINUE
      ITER=ITER+1
      DO 20 I=1,NGEN
      THETA=DEL(I)-3.1416/2.0
      SCALE=CMPLX(0.0,(XQ1(I)-XD1(I))*0.5)/CMPLX(R(I),-(XQ1(I)+XD1(I))*
     10.5)
      ROTATE=CMPLX(COS(2.0*THETA),SIN(2.0*THETA))
      EFICT(I)=E(I)+SCALE*CONJG(E(I)-VT(I))*ROTATE    YFICT = CMPLX(R(I), -
                                                      CFICT(I)= EFICT(I) * YFICT
20    CONTINUE
      DO 30 I=1,NGEN
      CT(I)=(0.0,0.0)
      DO 25 J=1,NGEN
25    CT(I)=CT(I)+Y(I,J)*EFICT(J)
30    CONTINUE
      DO 40 I=1,NGEN
      YFICT=CMPLX(R(I),-(XD1(I)+XQ1(I))/2.0)/(R(I)*R(I)+XD1(I)*XQ1(I))
      VT(I)=EFICT(I)-CT(I)/YFICT
40    CONTINUE
C     CHECK FOR CONVERGENCE.
      NFLAG=0
      DO 50 I=1,NGEN
      EQ=OUT(I,3)
      ED=OUT(I,4)
C     TRANSFORM TERMINAL VOLTAGE AND CURRENT TO MACHINE REFERENCE.
      THETA=DEL(I)-3.1416/2.0
      ROTATE=CMPLX(COS(THETA),-SIN(THETA))
      ID=REAL(CT(I)*ROTATE)
      IQ=AIMAG(CT(I)*ROTATE)
      VD=REAL(VT(I)*ROTATE)
      VQ=AIMAG(VT(I)*ROTATE)
      IF(ABS(EQ-R(I)*IQ-XD1(I)*ID-VQ) .GT. .001) NFLAG=1
      IF(ABS(ED-R(I)*ID+XQ1(I)*IQ-VD) .GT. .001) NFLAG=1
      VD1=ED-R(I)*ID+XQ1(I)*IQ
      VQ1=EQ-R(I)*IQ-XD1(I)*ID
50    CONTINUE
      IF(ITER .GE. 10) GO TO 60
      IF(NFLAG .EQ. 1) GO TO 15
      RETURN
60    WRITE(6,1025)
1025  FORMAT('0 SALIENCY ITERATIONS NOT CONVERGED')
      DO 70 I=1,NGEN
      WRITE(6,1010) I,VT(I),CT(I),VD,VQ,VD1,VQ1,ID,IQ
1010  FORMAT(' TERM',I5,10F10.4)
```

```
70     CONTINUE
       STOP
       END

C
C  MODEL OF SYNCHRONOUS GENERATOR WITH FIELD WINDING IN D AXIS AND
C  DAMPER WINDING IN Q AXIS.
       SUBROUTINE GEN1(I)
       COMMON /BLOCK1/ TIME,TSTEP
       COMMON /BLOCK2/ PBASE(10),H(10),R(10),XL(10),XD(10),XD1(10),
      1XQ(10),XQ1(10),TD1(10),TQ1(10),DAMP(10),C1(10),C2(10)
       COMMON/BLOCK5/ VT(10),CT(10),EF(10),PM(10)
       COMMON/BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
       COMMON /BLOCK9/ PRTVAR(10,20)
       REAL CSAT(10)
       COMPLEX CONJG,CMPLX,VT,CT,EQD,CURR
       REAL ID,IQ
C  ENTER HERE FOR EACH INTEGRATION STEP.
C  DEFINE INTEGRATOR OUTPUTS.
       OME=OUT(I,1)
       DEL=OUT(I,2)
       EQ=OUT(I,3)
       ED=OUT(I,4)
C  TRANSFORM CURRENT TO MACHINE REFERENCE.
       CURR=CT(I)*CMPLX(SIN(DEL),COS(DEL))
       ID=REAL(CURR)
       IQ=AIMAG(CURR)
C  CALCULATE GENERATOR OUTPUT PLUS LOSSES.
       PE=REAL(VT(I)*CONJG(CT(I)))
       QE=AIMAG(VT(I)*CONJG(CT(I)))
       PL=CABS(CURR)**2*R(I)
C  ADJUST REACTANCES AND TIME CONSTANTS TO ACCOUNT FOR SATURATION.
       XDS=CSAT(I)*XD(I)+(1.0-CSAT(I))*XL(I)
       XQS=CSAT(I)*XQ(I)+(1.0-CSAT(I))*XL(I)
       IF(XQ1(I) .EQ. XQ(I))XQS=XQ(I)
       TD1S=TD1(I)*(1.0-(1.0-CSAT(I))*(XD(I)-XD1(I))/(XD(I)-XL(I)))
       TQ1S=TQ1(I)*(1.0-(1.0-CSAT(I))*(XQ(I)-XQ1(I))/(XQ(I)-XL(I)))
       EAQ=EQ-(XD1(I)-XL(I))*ID
       EAD=ED+(XQ1(I)-XL(I))*IQ
       IF(XQ1(I) .EQ. XQ(I))EAD=0.0
       EAT=SQRT(EAQ**2+EAD**2)
       CSAT(I)=1.0/(1.0+C1(I)*EXP(C2(I)*EAT))
C  DEFINE INTEGRATOR INPUTS.
C  SET UP PRINTOUT VARIABLES.
       PLUG(I,1)=(PM(I)-PE-PL-DAMP(I)*OME)/(2.0*H(I))
       PLUG(I,2)=377.0*OME
       PLUG(I,3)=(CSAT(I)*EF(I)-EQ-(XDS-XD1(I))*ID)/TD1S
       PLUG(I,4)=(-ED+(XQS   -XQ1(I))*IQ)/TQ1S
       PRTVAR(I,1)=DEL*180.0/3.142
       PRTVAR(I,2)=OME
       PRTVAR(I,3)=EQ
       PRTVAR(I,4)=ED
       PRTVAR(I,5)=CABS(VT(I))
       PRTVAR(I,6)=PE*100.0/PBASE(I)
       PRTVAR(I,7)=QE*100.0/PBASE(I)
       PRTVAR(I,8)=EF(I)
       PRTVAR(I,9)=PM(I)*100.0/PBASE(I)
       PRTVAR(I,10)=CSAT(I)
       PRTVAR(I,11)=EAT
       RETURN
C  ENTER HERE TO CALCULATE INITIAL CONDITIONS.
       ENTRY GEN1IC(I)
       CSAT(I)=1.0
       DO 100 J=1,3
       XDS=CSAT(I)*XD(I)+(1.0-CSAT(I))*XL(I)
       XQS=CSAT(I)*XQ(I)+(1.0-CSAT(I))*XL(I)
       IF(XQ1(I) .EQ. XQ(I))XQS=XQ(I)
C  CALCULATE ANGLE OF GENERATOR Q AXIS.
       EQD=VT(I)+CMPLX(R(I),XQS   )*CT(I)
       DEL=ATAN2(AIMAG(EQD),REAL(EQD))
C  TRANSFORM CURRENT ONTO GENERATOR REFERENCE.
       CURR=CT(I)*CMPLX(SIN(DEL),COS(DEL))
       ID=REAL(CURR)
       IQ=AIMAG(CURR)
       EQ=CABS(EQD)-(XQS    -XD1(I))*ID
       EF(I)=(EQ+(XDS-XD1(I))*ID)/CSAT(I)
       ED=(XQS   -XQ1(I))*IQ
       EAQ=EQ-(XD1(I)-XL(I))*ID
       EAD=ED+(XQ1(I)-XL(I))*IQ
       IF(XQ1(I) .EQ. XQ(I))EAD=0.0
       EAT=SQRT(EAQ**2+EAD**2)
       CSAT(I)=1.0/(1.0+C1(I)*EXP(C2(I)*EAT))
100    CONTINUE
       PE=REAL(VT(I)*CONJG(CT(I)))+CABS(CURR)**2*R(I)
       PM(I)=PE
       OUT(I,1)=0.0
       OUT(I,2)=DEL
```

I=1,200

```
      OUT(I,3)=E0
      OUT(I,4)=E0
      RETURN
      END


C
C  ROTATING EXCITATION SYSTEM - IEEE TYPE 1 MODEL.
      SUBROUTINE AVR1(I)
      COMMON /BLOCK1/ TIME,TSTEP
      COMMON/BLOCK3/         KA(10),KE(10),KF(10),TA(10),TE(10),
     1TF(10),VRMIN(10),VRMAX(10),C1(10),C2(10),DUM(10,6)
      COMMON/BLOCK5/ VT(10),CT(10),EF(10),PM(10)
      COMMON/BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      COMMON /BLOCK9/ PRTVAR(10,20)
      COMPLEX VT,CT
      REAL KA,KE,KF,VREF(10),SE(10)
C  ENTER HERE FOR EACH INTEGRATION STEP.
C  DEFINE INTEGRATOR OUTPUTS.
      X5=OUT(I,6)
      EF(I)=OUT(I,5)
      X3=OUT(I,7)
C  CALCULATE INTERMEDIATE VARIABLES.
      X1=VREF(I)-CABS(VT(I))
      X2=KF(I)/TF(I)*EF(I)-X3
      X4=X1-X2
      X6=X5
      IF(X6 .LT. VRMIN(I)) X6=VRMIN(I)
      IF(X6 .GT. VRMAX(I)) X6=VRMAX(I)
      X7=SE(I)*EF(I)
      X8=X6-X7
C  DEFINE INTEGRATOR INPUTS.
      PLUG(I,5)=X8/TE(I)-KE(I)/TE(I)*EF(I)
      PLUG(I,6)=KA(I)/TA(I)*X4-X5/TA(I)
      PLUG(I,7)=X2/TF(I)
      RETURN
C  ENTER HERE TO CALCULATE INITIAL CONDITIONS.
      ENTRY AVR1IC(I)
      SE(I)=C1(I)*EXP(C2(I)*EF(I))
      VREF(I)=CABS(VT(I))+(KE(I)+SE(I))*EF(I)/KA(I)
      OUT(I,5)=EF(I)
      OUT(I,6)=EF(I)*(KE(I)+SE(I))
      OUT(I,7)=EF(I)*KF(I)/TF(I)
      IF(OUT(I,6) .LT. VRMIN(I)) WRITE(6,1020) I
      IF(OUT(I,6) .GT. VRMAX(I)) WRITE(6,1020) I
 1020 FORMAT('0**** AVR VOLTAGE LIMIT IS EXCEEDED BY INITIAL FIELD ON',
     1' UNIT',I3/)
      RETURN
      END


C
C  STATIC EXCITATION SYSTEM - IEEE TYPE 1S MODEL.
      SUBROUTINE AVR2(I)
      COMMON /BLOCK1/ TIME,TSTEP
      COMMON /BLOCK3/ KA(10),KF(10),TA(10),TF(10),KP(10),DUM(10,11)
      COMMON /BLOCK5/ VT(10),CT(10),EF(10),PM(10)
      COMMON /BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      COMMON /BLOCK9/ PRTVAR(10,20)
      COMPLEX VT,CT
      REAL KA,KF,KP,VREF(10)
C  ENTER HERE FOR EACH INTEGRATION STEP.
C  DEFINE INTEGRATOR OUTPUTS.
      X5=OUT(I,5)
      X3=OUT(I,6)
C  CALCULATE INTERMEDIATE VARIABLES.
      EF(I)=X5
      VMAG=CABS(VT(I))
      IF(X5 .GT. KP(I)*VMAG) EF(I)=KP(I)*VMAG
      IF(X5 .LT. -KP(I)*VMAG) EF(I)=-KP(I)*VMAG
      X2=EF(I)*KF(I)/TF(I)-X3
      X1=VREF(I)-VMAG
      X4=X1-X2
C  CALCULATE INTEGRATOR INPUTS.
      PLUG(I,5)=X4*KA(I)/TA(I)-X5/TA(I)
      PLUG(I,6)=X2/TF(I)
      RETURN
C  ENTER HERE TO CALCULATE INITIAL CONDITIONS.
      ENTRY AVR2IC(I)
      OUT(I,5)=EF(I)
      OUT(I,6)=EF(I)*KF(I)/TF(I)
      VREF(I)=CABS(VT(I))+EF(I)/KA(I)
C  CHECK IF INITIAL CONDITIONS ARE WITHIN LIMITS.
      VMAG=CABS(VT(I))
      IF(EF(I) .GT. KP(I)*VMAG) WRITE(6,1020) I
      IF(EF(I) .LT. -KP(I)*VMAG) WRITE(6,1020) I
 1020 FORMAT('0**** AVR VOLTAGE LIMIT IS EXCEEDED BY INITIAL FIELD ON',
     1' UNIT',I3/)
```

```
      RETURN
      END
C
C  MODEL OF ROTATING EXCITER WITH AUXILIARY STABILIZER.
      SUBROUTINE AVR3(I)
      COMMON /BLOCK1/ TIME,TSTEP
      COMMON /BLOCK2/ PBASE(10),H(10),R(10),XL(10),XD(10),XD1(10),
     1XQ(10),XQ1(10),TD1(10),TQ1(10),DAMP(10),C1(10),C2(10)
      COMMON /BLOCK3/ KA(10),TA1(10),TA2(10),KB(10),KE(10),TE(10),
     1KD(10),TD(10),KC(10),TC(10),T1(10),T3(10),KT(10),TFD(10),T(10),
     2UMAX(10)
      REAL KA,KB,KE,KD,KC,KT,VREF(10)
      COMMON /BLOCK5/ VT(10),CT(10),EF(10),PM(10)
      COMMON /BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      COMMON /BLOCK9/ PRTVAR(10,20)
      COMPLEX CMPLX,CONJG
      COMPLEX VT,CT,EQD
C  ENTER HERE FOR EACH INTEGRATION STEP.
C  DEFINE STABILIZER VARIABLES.
      OME=OUT(I,1)
      X7=OME
      X8=OUT(I,13)
      X9=OUT(I,14)+T1(I)/T3(I)*X8
      X10=OUT(I,15)+T1(I)/T3(I)*X9
      U=X10-OUT(I,16)
      IF(U .GT. UMAX(I)) U=UMAX(I)
      IF(U .LT. -UMAX(I)) U=-UMAX(I)
C  DEFINE STABILIZER INTEGRATOR INPUTS.
      PLUG(I,13)=(KT(I)*X7-X8)/TFD(I)
      PLUG(I,14)=(X8-X9)/T3(I)
      PLUG(I,15)=(X9-X10)/T3(I)
      PLUG(I,16)=U/T(I)
C  DEFINE AVR VARIABLES.
      EF(I)=OUT(I,5)
      X5=CABS(VT(I))*KC(I)/TC(I)-OUT(I,8)
      X6=CABS(VT(I))+X5
      X4=EF(I)*KB(I)*KD(I)/TD(I)-OUT(I,7)
      X1=VREF(I)+U-X4-X6
      X2=X1*KA(I)*TA1(I)/TA2(I)+OUT(I,6)
C  DEFINE AVR INTEGRATOR INPUTS.
      PLUG(I,5)=X2*KE(I)/KB(I)/TE(I)-EF(I)/TE(I)
      PLUG(I,6)=X1*KA(I)/TA2(I)-X2/TA2(I)
      PLUG(I,7)=X4/TD(I)
      PLUG(I,8)=X5/TC(I)
      PRTVAR(I,12)=U
      RETURN
C  ENTER HERE TO CALCULATE INITIAL CONDITIONS.
      ENTRY AVR3IC(I)
C  SET INITIAL CONDITIONS FOR STABILIZER INTEGRATORS.
      OUT(I,13)=0.0
      OUT(I,14)=0.0
      OUT(I,15)=0.0
      OUT(I,16)=0.0
C  SET INITIAL CONDITIONS FOR AVR.
      VREF(I)=EF(I)*KB(I)/KA(I)/KE(I)+CABS(VT(I))
      OUT(I,5)=EF(I)
      OUT(I,6)=EF(I)*KB(I)/KE(I)*(1.0-TA1(I)/TA2(I))
      OUT(I,7)=EF(I)*KB(I)*KD(I)/TD(I)
      OUT(I,8)=CABS(VT(I))*KC(I)/TC(I)
      RETURN
      END
C
C  STEAM TURBINE AND GOVERNOR MODEL.
      SUBROUTINE TUR1(I)
      COMMON /BLOCK1/ TIME,TSTEP
      COMMON /BLOCK4/ PBASE(10),THP(10),TIP(10),TLP(10),FHP(10),FIP(10),
     1 FLP(10),R(10),TS(10),DGMIN(10),DGMAX(10),GMIN(10),GMAX(10),
     1 DUM(10,3)
      COMMON /BLOCK5/ VT(10),CT(10),EF(10),PM(10)
      COMMON /BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      COMMON /BLOCK9/ PRTVAR(10,20)
      COMPLEX VT,CT
      REAL WREF(10)
C  ENTER HERE FOR EACH TIME STEP.
C  DEFINE INTEGRATOR OUTPUTS.
      OME=OUT(I,1)
      X4=OUT(I,9)
      X5=OUT(I,10)
      X6=OUT(I,11)
      X7=OUT(I,12)
C  CALCULATE INTERMEDIATE VARIABLES.
      G=X4
      IF(X4 .GT. GMAX(I)) G=GMAX(I)
      IF(X4 .LT. GMIN(I)) G=GMIN(I)
```

```
      X1=(WREF(I)-OME)/R(I)
      X2=(X1-G)/TS(I)
      X3=X2
      IF(X2 .GT. DGMAX(I)) X3=DGMAX(I)
      IF(X2 .LT. DGMIN(I)) X3=DGMIN(I)
      PM(I)=(FHP(I)*X5+FIP(I)*X6+FLP(I)*X7)*PBASE(I)/100.0
C   CALCULATE INTEGRATOR INPUTS.
      PLUG(I,9)=X3
      PLUG(I,10)=(G-X5)/THP(I)
      PLUG(I,11)=(X5-X6)/TIP(I)
      PLUG(I,12)=(X6-X7)/TLP(I)
      PRTVAR(I,12)=G
      RETURN
C   ENTER HERE TO CALCULATE INITIAL CONDITIONS.
      ENTRY TUR1IC(I)
      OME=OUT(I,1)
      G=PM(I)*100.0/PBASE(I)/(FHP(I)+FIP(I)+FLP(I))
      OUT(I,9)=G
      OUT(I,10)=G
      OUT(I,11)=G
      OUT(I,12)=G
C   CALCULATE SET POINT.
      WREF(I)=OME+OUT(I,9)*R(I)
C   CHECK IF INITIAL CONDITIONS ARE WITHIN LIMITS.
      IF(OUT(I,9) .GT. GMAX(I)) WRITE(6,1020) I
      IF(OUT(I,9) .LT. GMIN(I)) WRITE(6,1020) I
1020  FORMAT('0**** TURBINE GATE LIMIT IS EXCEEDED BY INITIAL POWER',
     1' ON UNIT',I3/)
      RETURN
      END


C
C   HYDRAULIC TURBINE AND GOVERNOR MODEL.
      SUBROUTINE TUR2(I)
      COMMON /BLOCK1/ TIME,TSTEP
      COMMON /BLOCK4/ PBASE(10),TW(10),TS(10),TR(10),PDR(10),TDR(10),
     1 DGMIN(10),DGMAX(10),GMIN(10),GMAX(10),DUM(10,6)
      COMMON /BLOCK5/ VT(10),CT(10),EF(10),PM(10)
      COMMON /BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      COMMON /BLOCK9/ PRTVAR(10,20)
      COMPLEX VT,CT
      REAL WREF(10)
C   ENTER HERE FOR EACH TIME STEP.
C   DEFINE INTEGRATOR OUTPUTS.
      OME=OUT(I,1)
      X4=OUT(I,9)
      X5=OUT(I,10)
      X7=OUT(I,11)
C   CALCULATE INTERMEDIATE VARIABLES.
      G=X4
      IF(G .GT. GMAX(I)) G=GMAX(I)              LIMITER
      IF(G .LT. GMIN(I)) G=GMIN(I)
      X6=G*TDR(I)-X5
      X1=WREF(I)-OME
      X2=(X1-X6-G*PDR(I))/TS(I)
      X3=X2
      IF(X2 .GT. DGMAX(I)) X3=DGMAX(I)
      IF(X2 .LT. DGMIN(I)) X3=DGMIN(I)
      X8=2.0*(X7-G)
      PM(I)=X8*PBASE(I)/100.0
C   CALCULATE INTEGRATOR INPUTS.
      PLUG(I,9)=X3
      PLUG(I,10)=X6/TR(I)
      PLUG(I,11)=(G-X8)/TW(I)
      PRTVAR(I,12)=G
      RETURN
C   ENTER HERE TO CALCULATE INITIAL CONDITIONS.
      ENTRY TUR2IC(I)
      OME=OUT(I,1)
      OUT(I,9)=PM(I)*100.0/PBASE(I)
      OUT(I,10)=PM(I)*100.0 /PBASE(I)*TDR(I)
      OUT(I,11)=1.5*PM(I)*100.0/PBASE(I)
C   CALCULATE SET POINT.
      WREF(I)=OME+OUT(I,9)*PDR(I)
C   CHECK IF INITIAL CONDITIONS ARE WITHIN LIMITS.
      IF(OUT(I,9) .GT. GMAX(I)) WRITE(6,1020) I
      IF(OUT(I,9) .LT. GMIN(I)) WRITE(6,1020) I
1020  FORMAT('0**** TURBINE GATE LIMIT IS EXCEEDED BY INITIAL POWER',
     1' ON UNIT',I3/)
      RETURN
      END


C
C   SUBROUTINE TO CALCULATE STATE VARIABLES FOR NEXT INCREMENT IN TIME.
      SUBROUTINE INT(NGEN)
      COMMON /BLOCK1/ TIME,TSTEP
```

Increase to 100

```
      COMMON/BLOCK6/ PLUG(10,16),OUT(10,16),SAVE(10,16)
      IF(TIME .GT. 0.0) GO TO 20
      DO 10 I=1,NGEN
      DO 10 J=1,16
10    SAVE(I,J)=PLUG(I,J)
      NFLAG=1
20    CONTINUE
      DO 30 I=1,NGEN
      DO 30 J=1,16
      OUT(I,J)=OUT(I,J)+PLUG(I,J)*TSTEP+(PLUG(I,J)-SAVE(I,J))*0.5*TSTEP
30    SAVE(I,J)=PLUG(I,J)
      RETURN
      END


C
C  SUBROUTINE TO PRINTOUT SYSTEM VARIABLES.
      SUBROUTINE OUTPUT(NGEN)
      COMMON /BLOCK1/ TIME,TSTEP
      COMMON /BLOCK8/ TYM(200),VAR(200,6),NT,NVAR
      COMMON /BLOCK9/ PRTVAR(10,20)
      IF(TIME .EQ. 0.0) WRITE(6,2000)
2000  FORMAT('1'//T42,'SIMULATED RESPONSES',//T4,'TIME   GEN   ROTOR    ',
     1'ROTOR     EQ''       ED''      TERM      ELEC-POWER      FIELD   ',
     2'MECH     SATN    AIR GAP'/T4,'SECS    NO   ANGLE     SPEED    ',
     3'VOLTS    VOLTS   VOLTS      REAL     IMAG     VOLTS    POWER   ',
     4'FACTOR    VOLTS'/)
      WRITE(6,1010) TIME
1010  FORMAT(' ',F7.3)
      DO 10 I=1,NGEN
      WRITE(6,1000) I,(PRTVAR(I,J),J=1,11)
1000  FORMAT(' ',8X,I3,F8.3,13F8.4)
10    CONTINUE
C  SET UP THE VARIABLES TO BE PLOTTED.
      NT=NT+1
      TYM(NT)=TIME
      VAR(NT,1)=PRTVAR(1,1)
      VAR(NT,2)=PRTVAR(1,5)
      VAR(NT,3)=PRTVAR(1,6)
      VAR(NT,4)=PRTVAR(1,8)
      RETURN
      END


C
C  SUBROUTINE TO PLOT GRAPHS OF SYSTEM VARIABLES.
      SUBROUTINE PLOT
      COMMON /BLOCK8/ TYM(200),VAR(200,6),NT,NVAR
      LOGICAL SYMBOL(6),PLUS,NAME(6,40),ALINE(132),BLANK
      DATA SYMBOL/'A','B','C','D','E','F'/
      DATA PLUS/'+'/,BLANK/' '/
      REAL VMAX(6),VMIN(6)
      REAL YVAL(11)
C  READ GRAPH NAMES AND MINIMUM AND MAXIMUM VALUES.
      DO 3 I=1,7
      IF(I .EQ. 7) GO TO 5
3     READ(5,1000,END=5)   (NAME(I,J),J=1,40),VMIN(I),VMAX(I)
1000  FORMAT(40A1,F10.5,F10.5)
5     NVAR=I-1
      IF(NVAR .EQ. 0) RETURN
      WRITE(6,1060)
1060  FORMAT('1')
C  WRITE HEADINGS FOR EACH GRAPH.
      DO 20 I=1,NVAR
      RANGE=VMAX(I)-VMIN(I)
      LOG=ALOG10(RANGE)
      SCALE=1.0*10.0**LOG
      DO 10J=1,11
10    YVAL(J)=(VMIN(I)+(J-1)*RANGE/10.0)/SCALE
      WRITE(6,1010) SYMBOL(I),(NAME(I,J),J=1,40),SCALE
1010  FORMAT('0',T12,A1,' - ',40A1,T93,'SCALE FACTOR = ',1PE7.1)
      WRITE(6,1020) (YVAL(J),J=1,11)
1020  FORMAT(' ',T10,11(F5.2,5X))
20    CONTINUE
C  WRITE Y AXIS.
      WRITE(6,1030) (PLUS,LOC=1,101)
1030  FORMAT('0     TIME   ',101A1)
C  PLOT GRAPHS.
      DO 50 J=1,NT
      DO 28 LOC=1,101
28    ALINE(LOC)=BLANK
      ALINE(1)=PLUS
      IF(MOD( J,10) .NE. 1) GO TO 32
C  INCLUDE GRID POINTS.
      DO 30 LOC=1,101,10
30    ALINE(LOC)=PLUS
32    CONTINUE
C  INCLUDE GRAPH POINTS.
```

```
      DO 35 I=1,NVAR
      LOC=(VAR(J,I)-VMIN(I))/(VMAX(I)-VMIN(I))*100.0+1.0
      IF(LOC .LT. 1) LOC=1
      IF(LOC .GT. 101) LOC=101
      ALINE(LOC)=SYMBOL(I)
35    CONTINUE
      IF(MOD(J,10) .EQ. 1) WRITE(6,1040) TYM(J),(ALINE(LOC),LOC=1,101)
1040  FORMAT(T4,F7.3,1X,101A1)
      IF(MOD( J,10) .NE. 1) WRITE(6,1045) (ALINE(LOC),LOC=1,101)
1045  FORMAT(T12,101A1)
50    CONTINUE
60    CONTINUE
      WRITE(6,1060)
      RETURN
      END
```

```
C     ******************************************************************
C     *                                                                *
C     *                   NETWORK REDUCTION PROGRAM                    *
C     *                                                                *
C     ******************************************************************
      COMPLEX Y(40,40),E(40)
      INTEGER TYPE(40)
      INTEGER GBUS(10)
      COMPLEX YSHUNT,YIJ,ZIJ,YII,YJJ,CURR,S
      COMPLEX CMPLX
      COMPLEX CONJG
C   INITIALIZE VARIABLES
      DO 10 I=1,40
      E(I)=(0.0,0.0)
      TYPE(I)=0
      DO 10 J=1,40
   10 Y(I,J)=(0.0,0.0)
      WRITE(6,2000)
 2000 FORMAT('1',T33,'NETWORK  REDUCTION  PROGRAM')
      READ(5,990) NBUS,NGEN,NFAULT,NPUNCH
  990 FORMAT(4I5)
      WRITE(6,995) NBUS,NGEN,NFAULT,NPUNCH
  995 FORMAT('0'/' NO. OF BUSES',T19,I4 /' NO. OF GENERATORS',T19,I4/
     1' FAULT BUS',T19,I4/' PUNCH FLAG',T19,I4)
      WRITE(6,2005)
 2005 FORMAT('0'/' BUS  DATA'/T37,'LOAD         LOAD        GEN        GEN',
     1'        SHUNT      SHUNT'/T4,'BUS  TYPE   VOLTS      ANGLE       ',
     2'MW       MVAR      MW        MVAR        G          B'/)
C   LOOP HERE FOR EACH BUS
      N=0
   16 READ(5,1001) I,NTYPE,EMAG,ARG,PL,QL,PG,QG,YSHUNT
 1001 FORMAT(I5,T18,I1,F6.4,F6.2,4F6.1,T67,2F6.3)
      IF(I.EQ.0) GO TO 17
      WRITE(6,1002) I,NTYPE,EMAG,ARG,PL,QL,PG,QG,YSHUNT
 1002 FORMAT(' ',2I5,F10.4,F10.3,4F10.2,2F10.4)
      ARG=ARG*3.14159/180.0
      Y(I,I)=Y(I,I)+YSHUNT+CMPLX(PL,-QL)*.01/EMAG**2
      TYPE(I)=NTYPE
      E(I)=EMAG*CMPLX(COS(ARG),SIN(ARG))
      IF(TYPE(I) .EQ. 0) GO TO 16
      N=N+1
      GBUS(N)=I
      GO TO 16
   17 CONTINUE
      NGEN=N
      WRITE(6,2010)
 2010 FORMAT('0'/' LINE  DATA'/'   BUS  BUS  RESISTANCE  REACTANCE',
     1'  SUSCEPTANCE'/)
C
C   LOOP HERE FOR EACH LINE TO BE READ
   20 READ(5,1003) I,J,ZIJ,B
 1003 FORMAT(2I5,3F10.5)
      IF(I.EQ.0) GO TO 25
      WRITE(6,1005)I,J,ZIJ,B
 1005 FORMAT(' ',2I5,F10.5,2F12.5)
      YIJ=(1.0,0.0)/ZIJ
      Y(I,I)=Y(I,I)+YIJ+CMPLX(0.0,B/2.0)
      Y(J,J)=Y(J,J)+YIJ+CMPLX(0.0,B/2.0)
      Y(I,J)=Y(I,J)-YIJ
      Y(J,I)=Y(J,I)-YIJ
      GO TO 20
   25 CONTINUE
      WRITE(6,2015)
 2015 FORMAT('0'/' TRANSFORMER  DATA'/'   BUS  BUS  RESISTANCE  ',
     1'REACTANCE    TAP'/)
C
C   LOOP HERE FOR EACH TRANSFORMER CARD TO BE READ
   30 READ(5,1004)I,J,ZIJ,RATIO
 1004 FORMAT(2I5,2F10.5,F6.4)
      IF(I.EQ.0) GO TO 35
      WRITE(6,1006) I,J,ZIJ,RATIO
 1006 FORMAT(2I5,F10.4,2X,F10.4,F10.4)
      YIJ=(1.0,0.0)/ZIJ
      YII=YIJ*(1.0/RATIO-1.0)/RATIO
      YJJ=YIJ*(1.0-1.0/RATIO)
      YIJ=YIJ/RATIO
      Y(I,I)=Y(I,I)+YII+YIJ
      Y(J,J)=Y(J,J)+YJJ+YIJ
      Y(I,J)=Y(I,J)-YIJ
      Y(J,I)=Y(J,I)-YIJ
      GO TO 30
   35 CONTINUE
      IF(NFAULT.EQ.0) GO TO 39
C   ZERO ROW AND COLUMN AT THE FAULTED BUS
      DO 38 I=1,NBUS
      Y(I,NFAULT)=(0.0,0.0)
      Y(NFAULT,I)=(0.0,0.0)
   38 CONTINUE
```

```
39    CONTINUE
C  PERFORM KRON ELIMINATION ON LOAD BUSES.
      DO 60 M=1,NBUS
      IF(M .EQ. NFAULT) GO TO 60
      IF(TYPE(M) .EQ. 1) GO TO 60
      IF(CABS(Y(M,M)) .EQ. 0.0) GO TO 60
      TYPE(M)=-1
      DO 50 I=1,NBUS
C  ONLY PROCESS ROWS AND COLUMNS WHICH HAVE NOT BEEN ELIMINATED.
      IF(TYPE(I) .EQ. -1) GO TO 50
      DO 40 J=1,NBUS
      IF(TYPE(J) .EQ. -1) GO TO 40
      Y(I,J)=Y(I,J)-Y(I,M)*Y(M,J)/Y(M,M)
40    CONTINUE
50    CONTINUE
60    CONTINUE
      WRITE(6,2020)
2020  FORMAT('0'/'   COMPUTED GENERATION'/'  BUS        MW          MVAR'/)
C   CALCULATE GENERATOR POWERS TO CHECK REDUCTION
      DO 80 I=1,NGEN
      CURR=(0.0,0.0)
      DO 75 J=1,NGEN
75    CURR=CURR+Y(GBUS(I),GBUS(J))*E(GBUS(J))
      S=E(GBUS(I))*CONJG(CURR)
      WRITE(6,1040) GBUS(I),S
1040  FORMAT(' ',I5,2P2F10.2)
   80 CONTINUE
      WRITE(6,2025)
2025  FORMAT('0'/'   EQUIVALENT  GENERATOR  BUS  ADMITTANCE  MATRIX'/)
      DO 85 I=1,NGEN
85    WRITE(6,1050) (Y(GBUS(I),GBUS(J)),J=1,NGEN)
1050  FORMAT((' ',8F10.4))
      IF(NPUNCH .NE. 1) GO TO 100
      DO 90 I=1,NGEN
90    WRITE(7,1060) (Y(GBUS(I),GBUS(J)),J=1,NGEN)
1060  FORMAT(8F10.4)
100   CONTINUE
      WRITE(6,2030)
2030  FORMAT('1')
      STOP
      END
```