

Large Power System Stability Analysis Using a FOSS-based tool: SciLab/Xcos

Vikas Singh, H. Girisha Navada and K. N. Shubhanga

Department of Electrical Engineering, NITK, Surathkal, 575025, INDIA, E-mail: knsa1234@yahoo.com

Abstract—This paper describes the usage of an open-source tool namely Scilab-package for development of a multi-machine small-signal stability programme. It is shown that the package has enough computational capabilities to carry out large power system analysis. Analytical and time-domain simulation results obtained for a well-known 4-machine, 10-bus, 10-machine, 39-bus and 50-machine, 145-bus power systems demonstrate that Scilab/Xcos can be an alternate open-source tool to conventional proprietary software.

Index Terms—FOSSEE, Open-source tools, SciLab, Load flow, Small-signal stability analysis, Time-domain simulation.

I. INTRODUCTION

In the recent years, there has been a continued effort to promote the usage of open-source packages for engineering education. In this context, Free and Open Source Software for Science and Engineering Education (FOSSEE) [1] project is a part of National Mission on Education through Information and Communication Technology (NMEICT) with the thrust area being “adaptation and development of open source simulation packages” equivalent to proprietary software, funded by MHRD govt. of India. One of the most important activities of FOSSEE is promotion of SciLab [2], which stands for Scientific Laboratory, a good alternative to MATLAB. Scilab is a cross-platform, free and open-source numerical computational package which is easy to use and offers high level, matrix-based programming language with a versatile inbuilt mathematical library. Scilab also provides Simulink like environment with Xcos as a graphical dynamic modeler and time-domain simulator. In contrast, the other open-source alternatives like Octave [3] and Modelica [4] do not have this feature.

Software packages for Power System Analysis are mainly classified into 2 classes of tools [5]: Commercial software and educational/research-aimed software. Commercial software like MATLAB are typically well-tested and computationally more efficient but these software are ‘closed’, their source code can neither be modified nor is freely distributed. On the other hand, there is a wide variety of open-source research tools which are mainly used for Power System Analysis like MATPOWER [6], Power System Analysis Toolbox (PSAT) [7], Voltage Stability Toolbox (VST) [8], SimPowerSystems (SPS) [10], Matsim [11] etc, are Matlab-based software packages. MATPOWER, PSAT can also run on GNU Octave. On the other hand Scilab programs and its modules are standalone and do not require any other software for compilation and modeling.

In this paper, the features of an open-source tool namely Scilab is elaborated by developing a small-signal stability programme for multi-machine system. The problem is formulated using an eigenvalue based approach in Scilab and power system components are modeled in Xcos to perform the time-domain simulation. The inferences made from eigenvalue analysis are verified through time-domain simulations.

In Section II, Scilab and Xcos features are illustrated. In Section III, the developed load-flow programmes in Scilab are tested with standard IEEE power systems. In Section IV, eigenvalue approach employed for small-signal stability analysis is briefed along with Xcos modeling of different power system components. In Section V, the results obtained for a well-known 2-area, 4-machine, 10-machine, 39-bus New England system and 50-machine, 145-bus IEEE power system are presented. The programmes have been developed and tested in Scilab-6.0.0 and Scilab-6.0.1 versions.

II. SCILAB & XCOS FEATURES

Scilab is a numerical computation software that is freely available for windows, Linux, Mac OS and can be downloaded at the following address: <http://www.scilab.org>

A. Scilab Commands

Some useful Scilab-commands are listed in Table I, comparing them with those available in MATLAB.

TABLE I
SOME MATLAB-SCILAB EQUIVALENTS

MATLAB Command	SCILAB Command	Command description
pi	%pi	π Pre-defined variable.
%	//	use to start a comment line.
a+j*b	a + %i * b	Complex number.
who	who_user	listing of user's variables.
zeros(3)	zeros(3,3)	3-by-3 matrix of zeros.
grid on	xgrid() / set(gca(),'grid',[1 1])	grid On for plots.
filename	exec("filename.sci",-1)	executes a file from Scilab console.
lookfor	apropos	search about something.
format short	format(7)	scale fixed point format with 5 digits.
axis([Xmin Xmax Ymin Ymax])	zoom_rect([Xmin Ymin Xmax Ymax])	set the axis limit.
sort	gsort	use to sort the elements.
load filename.dat	read('filename.dat',m,n)	load .dat file into the workspace.
fopen	mopen	For file open in 'fid' command.
pause	sleep (In Scilab 6.0.1)	it waits for the key press.
eig	spec	find eigenvalues & eigenvectors.

Additional Scilab-commands are listed below:

- 1) *Script file execution:* Any script file in the current path can be executed by writing: `exec("filename.sci",mode)`, where important modes are as follows:
 - 1: Nothing is printed in Scilab console.
 - 1 : Each command line is printed.
 - 0 : Only output (which is not suppressed) is printed.

- 2) *Creation and execution of a function file:* A function file in Scilab text editor is written as:

```
function[out] = convert(a,b)
//Help comments are written here.
out = a.^2 + b.^ 2;
endfunction
```

In Scilab, it is not necessary to save the 'filename' with the same name as the 'function name', `convert`. The function file should be executed first by typing: `exec("filename.sci",-1)` in the Scilab console. The function 'convert' can now be used in the Scilab console by typing: `n=convert(a,b)`. Workspace variables can be directly used in a function without global declaration.

- 3) *Sparse Operations:*

- a) Retrieves entries of sparse matrix: The MATLAB command `pvpq_buses=pq(pv==pq)` is used to retrieve common indices of pv and pq sparse matrices in pq matrix, whereas in Scilab, this can be done using `spget` command as following:

```
lf= (pv==pq);
[ij,v,mn]= spget(lf);
pvpq_buses= pq(ij(:,1))
```

In the above code, first line describes the array position where pv and pq entries are equal. Second line retrieves entries of sparse matrix 'lf' and third line finds the common elements from pq matrix which are in pv matrix as well.

- b) Row/ Column deletion of a Sparse Matrix: In Scilab, a square Sparse matrix has a tendency to form sparse square matrix after deletion of a row or a column from it. Therefore, first make the matrix full to perform the deletion and make the matrix sparse again.
- c) Subtraction from a sparse matrix : Subtracting a complex number from an element of a sparse matrix gives erroneous result, while addition operation performs correctly.
- 4) *'atan' elementary function:* `phi = atan(y,x)` computes the 4-quadrant arctangent, it returns the argument (angle) of the complex number $x + jy$ in the range of $(-\pi, \pi)$.

B. Xcos Usage

Xcos opens by default with 2 windows, A palette browser which provides a set of predefined blocks and an editing window where library blocks are assembled to build a model file. In the following lines, an Xcos model is built to perform time-domain solution of a RL series circuit. Here, the circuit

is excited by a unit-step input with R and L treated as vectors. State-space equation for the inductor current with $i_L(0) = 0$ is given by

$$\frac{di_L}{dt} = -\frac{R}{L} i_L + \frac{v}{L} \quad (1)$$

The above state-model is realized in Xcos as shown in Fig 1.

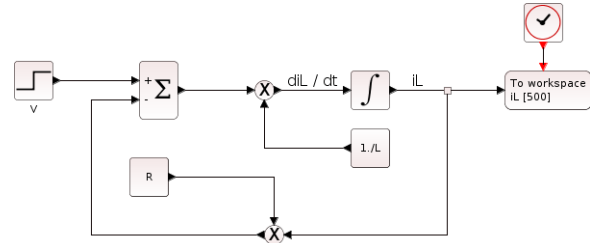


Fig. 1. RL-series circuit.

To run the simulation, perform the following:

- 1) Initialize $R = [1;0.2;0.5]$, $L = [0.1;0.2;0.25]$ in the Scilab console.
- 2) To connect any 2 blocks, output port size of one must match with input port size of other. For example, A block of output port size (-1,1) can only connect to other block of input port size (-1,1) where -1 could be any integer.
- 3) The Product blocks are employed to perform element-by-element operation effectively unlike gain blocks present in palette browser. Further, a period operator is used in the (1./L) block.
- 4) Set the simulation parameters as :
final integration time = 5 s and step size = 0.01 s.
- 5) Integrator block of type INTEGRAL_m is used for the simulation.
- 6) To plot the inductor current (i_L), either connect the scope (CSCOPE) or To workspace block at the output. Since it is an event driven block, the CLOCK_c block is connected to its red-port. In 'To workspace' block, set the variable name as i_L , fix the buffer size (total no. of samples recorded) to 500 (integration time/step-size). To access the stored variables in the scilab console, type ' $iL.time$ ' for the sampling time and ' $iL.values$ ' for the recorded iL values. The output waveforms of the inductor current are shown in Fig. 2.

NOTE:

- 1) *RK45-Runge-Kutta 4(5)*, a fixed-step method is chosen as a solver.
- 2) It should be noted that for three different combinations of R(i) and L(i), only one Xcos model is built, and in one time-domain simulation run, three corresponding $i_L(t)$ are plotted simultaneously. This shows the significance of vectorization of variables. This approach has been employed for modeling Swing-equations of all n_g -number of generators.

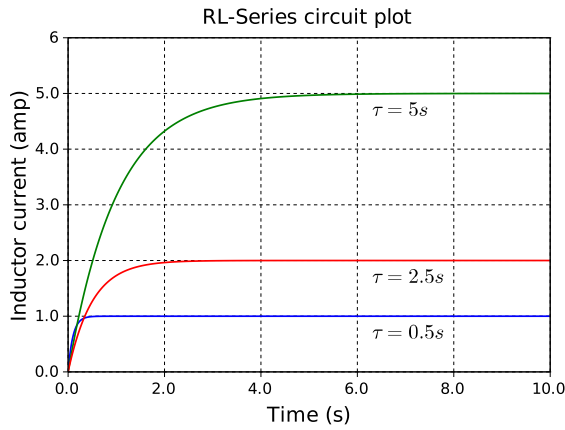


Fig. 2. Output Waveforms Of Inductor Current.

3) *Handling of Saturation and Threshold Switch*: The Saturation block in Xcos can take only scalar input of size (1,1). Therefore, for a vector input, the saturation function is performed using Scilab function block (*scifunc_block_m*). In the Function block, the following expression is written as :

$$y1 = (u1 \geq Efd_min \ \& \ u1 \leq Efd_max) .* u1 + (u1 > Efd_max) .* Efd_max + (u1 < Efd_min) .* Efd_min$$

where $u1$ is the input vector, Efd_min is lower limit, Efd_max is upper limit and $y1$ is the output vector of the function block. Similarly, a Threshold switch (*SWITCH2_m*) has been modeled.

III. LOAD FLOW STUDIES

In addition to other applications in power system [9], load flow analysis provides the initial operating values for the system variables, before any dynamic analysis is carried out. Thus, it avoids long duration time-domain simulation to arrive at a pre-fault operating condition. It is well known that load flow problem generally involves the simultaneous solution of many non-linear algebraic equations using numerical iterative procedure. Most commonly used iterative methods are the *Gauss-siedel*, the *Newton-Raphson (NR)* and *fast-decoupled load flow (FDLF)* methods. In this package, FDLF method [16] is employed for the load flow solution.

The corresponding equations are given by:

$$\begin{bmatrix} \Delta P \\ \Delta V \end{bmatrix} = [B'] [\Delta \delta] \quad (2)$$

$$\begin{bmatrix} \Delta Q \\ \Delta V \end{bmatrix} = [B''] [\Delta |V|] \quad (3)$$

where B' and B'' are susceptances matrices and are obtained from the bus admittance matrix. The correction vectors $\Delta \delta$ and $\Delta |V|$ are obtained by solving (2) and (3) respectively, in a sequential fashion [13]. FDLF method is implemented using *selective iteration scheme* in which ΔP and ΔQ are verified individually for convergence.

In Scilab, FDLF program is successfully tested with 4 machine, 10-bus, 10 machine, 39-bus and 50 machine, 145-bus power systems.

IV. SMALL-SIGNAL STABILITY ANALYSIS

A typical Classification of power system stability analysis is shown in Fig. 3.

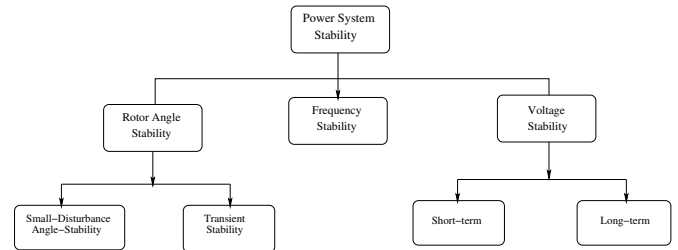


Fig. 3. Classification of power system stability

Of these, small-signal stability is the most fundamental requirement of any power system as it denotes the operating point stability. To analysis this, system equations given by

$$\dot{x} = F(x, y, u) \quad (4)$$

$$0 = g(x, y) \quad (5)$$

are linearized around an operating point to obtain final state-matrix, $[A]$ [12]. To identify the nature of oscillatory modes, the following procedure is employed:

- 1) Using 'spec' command, eigenvalues and eigenvectors are determined.
- 2) Employing a systematic participation factor (PF) analysis [14], the eigenvalues are classified as *Swing or Non-swing modes*.
- 3) Once the eigen predictions are recorded, a time-domain simulation [18] with detailed models for power system components are run to validate the predictions.

A. Xcos Modeling of Power Systems

Synchronous Generators are modeled employing 2.2 model [12]. For illustration purpose, swing equations and single-time constant static exciter are modeled below.

1) *Swing Equations*:: Swing equations are given by

$$\frac{d\delta}{dt} = S_m \omega_B \quad (6)$$

$$2H \frac{dS_m}{dt} = -\bar{D} S_m + \bar{T}'_m - \bar{T}_{eg} \quad (7)$$

The above equations are implemented in Xcos as shown in Fig. 5

2) *Single-time constant static exciter*:: The Xcos model is shown in Fig. 6

The complete Xcos modeling of the interfacing of different power system components employing current injection method is shown in Fig. 4

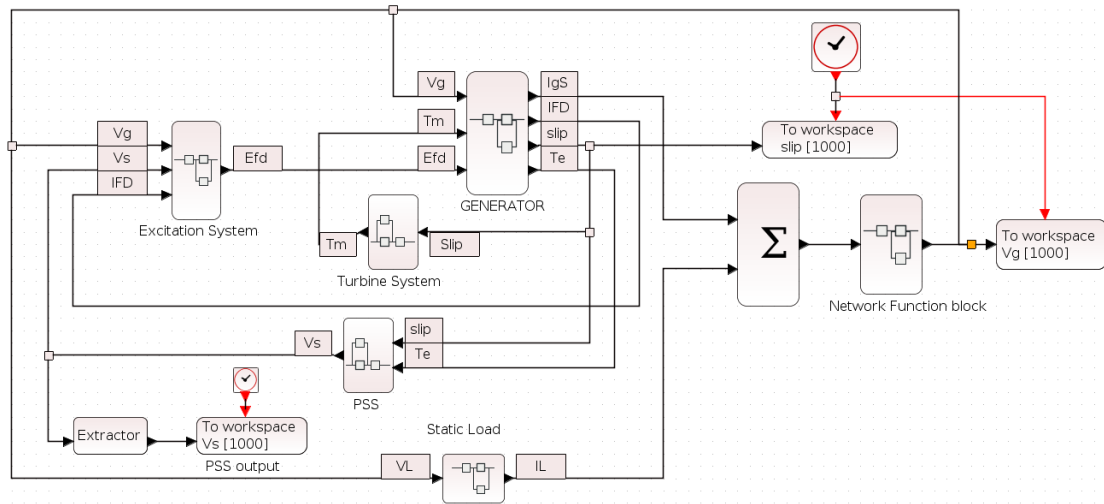


Fig. 4. Xcos modelling of power system components

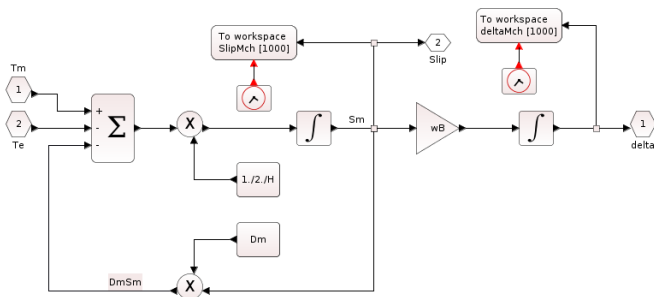


Fig. 5. Swing equations modeling of generator

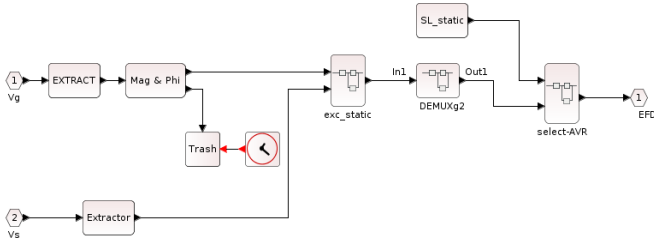


Fig. 6. Static exciter modelling

V. CASE STUDIES

For 4-machine and 10-machine power-systems, the system details are adopted from [12], and the data for the 50-machine IEEE system has been obtained from [19].

A. 4-machine, 10-bus power system:

In this case, generators are provided with a single-time constant static exciter with no PSS, and turbines are not considered. Further, constant impedance type load model has been employed for both real and reactive components of loads. The system diagram is shown in Fig. 7

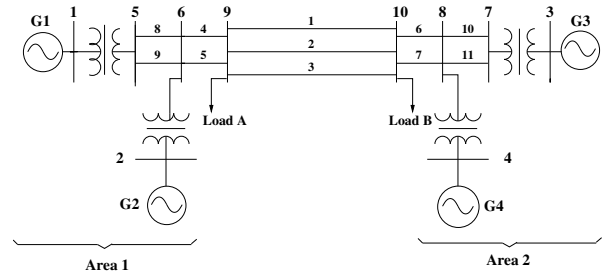


Fig. 7. Four Machine Power System

A function file is written in Scilab to perform the Network solution using back-slash commands as follows :

```
function[Vr,Vim]= network_solution(u1,u2)
u3=u1+%i*u2;
if abs(u3(nb+1)) < Tfault then
    Yuse=Y // pre-fault Ybus
    V1=(Yuse)\(u3(1:nb,1))
    Vr=real(V1)
    Vim=imag(V1)
end
endfunction
```

By running Scilab script code, eigenvalues are obtained at the operating point. Out of the 3 swing modes, 2 modes are pertaining to local-area mode and 1 mode is pertaining to inter-area mode as shown in Table II.

TABLE II
OSCILLATORY MODES- 4 MACHINE SYSTEM (BASE CASE)

SL No.	Eigen values	Damping factor	Frequency	Nature of Modes
1	$-1.1037 \pm j7.4473$	0.146603	1.185277	Swing- mode (m/cs. 1 and 2)
2	$-1.0487 \pm j6.7981$	0.152466	1.081954	Swing- mode (m/cs. 3 and 4)
3	$-0.0372 \pm j4.4582$	0.008349	0.709547	Swing- mode (m/cs. (1,2) and (3,4))

A compass plot similar to MATLAB is plotted here also as in Fig. 8 for inter-area swing mode. A time-domain simulation is performed for the above system at the base-case by perturbing the V_{ref} of generator-1. In this regard, slip-COI signals obtained are taken up for analysis as shown in Fig. 9.

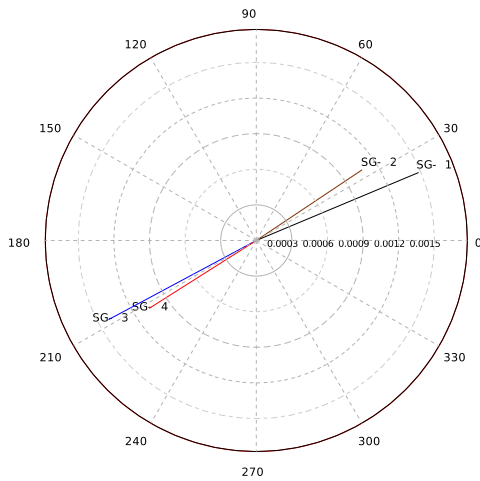


Fig. 8. Plot of slip-right eigenvectors for machines (1,2) and (3,4).

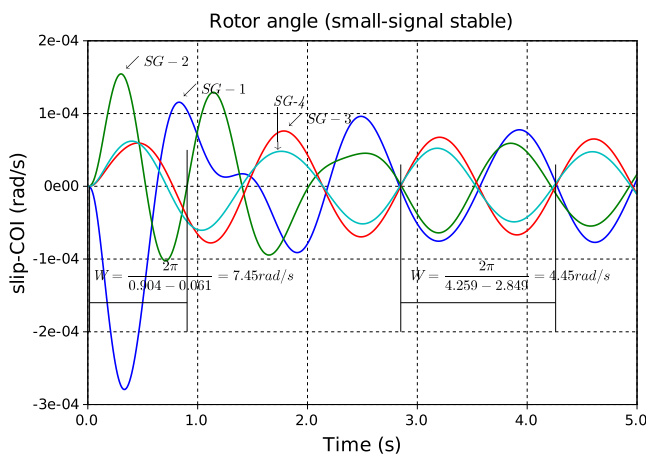


Fig. 9. Slip_COI plots for the perturbation of Vref of m/c-1

The frequencies of oscillations obtained for local-area mode (m/cs 1 and 2) and inter-area mode (m/cs (1,2) and (3,4)) in Table II are verified from Fig. 9.

NOTE: An execution of the following Latex-like commands in Scilab produce arrows and text prints in the figures.

```
//.....Naming the machine swing.....//
xstring(0.828, 1.17 * 10-4, '$\swarrow SG - 1$')
//.....Writing text in figure.....//
xstring(0.02, -1.4 * 10-4,
'$W = \frac{2\pi}{0.904 - 0.061} = 7.45rad/s$');
```

Large disturbance (transient) stability simulations for a 3-phase fault at bus-9 and line-1 tripping are also carried out and the critical clearing time are compared with those of Simulink estimates. The results are given in Table III.

TABLE III
CRITICAL CLEARING TIME COMPARISON

Cases	MATLAB (2014)	Scilab-6.0.0
AVR enabled	0.377 s	0.376 s
AVR disabled	0.303 s	0.299 s

The results are obtained with *Runge-kutta 4(5)* as the solver in Scilab and with ode 4 (Runge-Kutta) in Matlab.

B. 10-machine, 39-bus power system:

The single line diagram of the system is shown in Fig. 10. 1.0 model is employed for generators. Static exciters are enabled on all 10 machines. The system is subjected to disturbance by applying 3-phase fault at bus-34, followed by tripping of line 34-35 in the time-domain simulation. Here, a variable step-size method namely, Sundials/CVODE-ADAMS-NEWTON is used. The delta-COI plots obtained for an unstable case is shown in Fig. 11

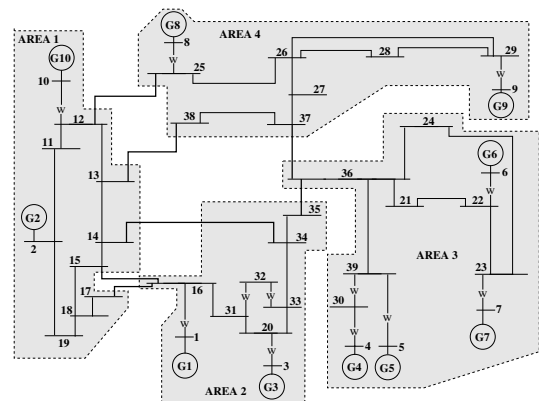


Fig. 10. 10-machine 39-bus Power System.

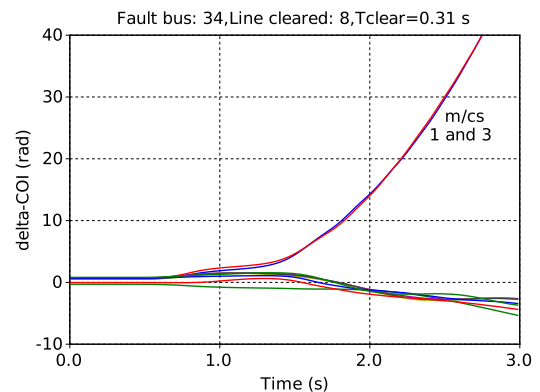


Fig. 11. delta_COI_plots for large-signal unstable case.

C. 50-machine, 145-bus IEEE power system:

This system has 145 buses, 401 lines, 52 transformers, 64 loads and 97 shunts. Out of 50 generators, 44 generators are with classical model and the rest 6 generators are with 1.1 model. These 6 generators are provided with IEEE-type AC4A exciters. To demonstrate the applicability of the developed programme, a three phase fault is applied at time $t=1$ s at bus 7 and is cleared by tripping line 6-7. The loads are modeled as constant impedance type. For a fault clearing time of 0.143 s (T_{cr}), the rotor angle plots are shown in Fig.12

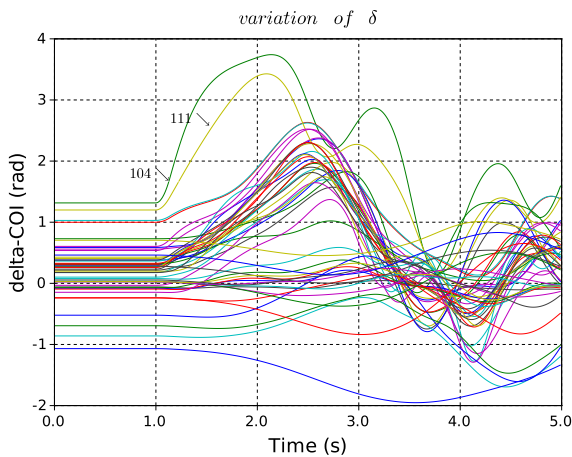


Fig. 12. Variation of rotor angle (50 m/c system).

VI. CONCLUSION

In this paper a multi-machine small-signal stability analysis programme has been developed using an open source tool namely Scilab/Xcos. The programmes have been tested employing 10-bus, 4-machine, 39-bus, 10-machine and 145-bus, 50-machine power systems. From the implementation and case study runs the following observations are made:

- 1) The Scilab scripting of Eigenvalue analysis programme found to work similar to the MATLAB coding. Further, most of the MATLAB-kind of commands are available even in Scilab except for minor syntax differences.
- 2) Though Simulink has the many other capabilities, one major advantage of Xcos is that it provides an open-source alternative to Simulink-like functioning.
- 3) The Xcos library has almost all blocks similar to Simulink. However, they are not as 'polished' as Simulink palettes when vectorized variables are to be handled.
- 4) With regard to fixed step numerical integration methods available for carrying out time-domain simulations, Xcos provides a few options as compared to Simulink.
- 5) Xcos models built on higher version, say, Scilab-6.0.1 can be run using lower version (Scilab-6.0.0) as well, even on older Linux distributions.

- 6) The installation file size for Scilab hardly exceeds 600MB, unlike Matlab (in GBs), independent of the operating system.

The programmes are available for free download at the website: <https://goo.gl/l2jnld> along with manuals.

ACKNOWLEDGEMENT

The authors would like to thank Prof. A.M. Kulkarni and Prof. Kannan Moudgalya of IIT Bombay for their guidance and support.

REFERENCES

- [1] fossee better education [online]. Available: <https://fossee.in>
- [2] Scilab Enterprises. Scilab main web page [online]. Available: <http://scilab.org>. Last seen on 5 May. 2018
- [3] About Octave [online]. Available: <https://www.gnu.org/software/octave/about.html>. Seen on 5 May. 2018
- [4] Sven Erik Mattson, Hilding Elmqvist, "An Overview Of the Modelling language MODELICA", Eurosim'98 Simulation Congress , April 14-15 , 1998 , Helsinki , Finland.
- [5] Federico Milano, "An Open Source Power System Analysis Toolbox", *IEEE Trans. on Power Systems*, vol.20,no.3, August 2005
- [6] R.D. Zimmerman, C.E. Murrillo-Sanchez, and D.Gan (2005) Matpower, version 3.0.0, User's Manual. Power System Engineering Reserach Center, Cornell Univ., Ithaca, NY. [Online] Available: <http://www.pserc.cornell.edu/matpower>
- [7] J.H. Chow and K.W.Cheung, "A toolbox for power system dynamics and control engineering education and research," *IEEE Trans. Power Syst.*, vol.7, n0.4, pp. 1559-1564, Nov.1992.
- [8] A. H. L. Chen, C. O. Nwankpa, H.G. Kwatny, stability toolbox: an Introduction and implement:*North American Power Symp.*, Nov.1996.
- [9] Kusic, G.L, "Computer Aided Power System Analysis", Prentice-Hall, Englewood Cliffs, New Jersey, 1986.
- [10] G. Sybille. (2004, Oct.) SimPowerSystems User published under sublicense from Hydro-Quebec.; Inc.[Online]. Available at: <http://www.mathworks>.
- [11] Vinayaka G Yaji, Abhishek G.S and K.N. Shubhanga, "Matsim-A Tool to Perform Power System Stability Studies", *Centenary Conference, 2011-Electrical Engineering, Indian Institute of Science, bangalore*.
- [12] K.R. Padiyar, *Power System Dynamics - Stability and control*, BS Publications, Hyderabad, India, 2002.
- [13] K.N.Shubhanga and Chandra S.R, *Manual for A Power Flow Programme*. Surathkal: NITK 2009.
- [14] K.N.Shubhanga and Y.Anantholla, *Manual for A Multi-machine Small Signal Stability Programme*. Surathkal: NITK 2009 [online]. Available: <http://www.nptel.ac.in/courses>.
- [15] Pradeep and K.N.Shubhanga, *Manual for A Multi-machine Transient Signal Stability Programme*. Surathkal: NITK 2009 [online]. Available: <http://www.nptel.ac.in/courses>.
- [16] Stott B and O. Alsac, "Fast Decoupled Load Flow Method", *IEEE Trans on Power Apparatus and Systems*, Vol.PAS-93, pp. 859, 1974.
- [17] P. Kundur, *Power System Stability and control*, McGraw-Hill, New York, 1994.
- [18] P.W. Sauer and M.A. Pai, *Power System Dynamics and stability*, Prentice Hall, Upper Saddle River, New Jersey, 1998.
- [19] Power System Test Cases Archive, UWEE, www.ee.washington.edu/research/pstca.