An Open Source
Power System Analysis Toolbox

Prof. Dr. Federico Milano

E-mail: Federico.Milano@uclm.es

Tel.: +34 926 295 219

Departamento de Ingeniería Eléctrica, Electrónica, Automática y Comunicaciones

Universidad de Castilla - La Mancha, España

Who I am

- Federico Milano received from the University of Genoa, Italy, the Electrical Engineering degree and the Ph.D. degree in 1999 and 2003, respectively.
- From 2001 to 2002 he worked at the University of Waterloo, Canada, as a Visiting Scholar.
- He is currently an assistant Professor at the University of Castilla-La Mancha, Ciudad Real, Spain.
- His research interests include voltage stability, electricity markets and computer-based power system analysis and control.

Contents

- Open Source vs. Commercial Power System Softwares
- Introduction to PSAT
- Examples
- Conclusions

Open Source vs. Commercial

Power System Softwares

Universidad de Castilla-La Mancha

Power System Software Outlines

- Software packages for power system analysis can be basically divided into two classes of tools:
  - Commercial softwares.
  - Educational/research-aimed softwares.

Commercial Softwares

- Commercial softwares (the list is not complete!):
  - PSS/E
  - EuroStag
  - Simpow
  - CYME
  - PowerWorld
  - Neplan

Commercial Softwares

- Commercial software packages follows an “all-in-one” philosophy and are typically well-tested and computationally efficient.
- Despite their completeness, these softwares can result cumbersome for educational and research purposes.
- Commercial softwares are “closed”, i.e. do not allow changing the source code or adding new algorithms.

Open-Source Softwares

- For research purposes, the flexibility and the ability of easy prototyping are often more crucial aspects than computational efficiency.
- At this aim, there is a variety of open source research tools, which are typically aimed to a specific aspect of power system analysis.
- An example is UWPFLOW which provides an extremely robust algorithm for continuation power flow analysis.

Compiled vs. Script Languages

- C anf FORTRAN are very fast but requires keen programming skills and are not suitable for fast prototyping.
- Several high level scientific languages, such as Matlab, Mathematica and Modelica, have become more and more popular for both research and educational purposes.
- At this aim, there is a variety of open source research tools, which are typically aimed to a specific aspect of power system analysis.
- Matlab proved to be the best user choice.

Matlab-based Power System Toolboxes

- Matlab-based power system analysis tools (the list is not complete!):
  - Power System Toolbox (PST)
  - MatPower
  - Voltage Stability Toolbox (VST)
  - Power Analysis Toolbox (PAT)
  - Educational Simulation Tool (EST)
  - Power system Analysis Toolbox (PSAT)

Matlab-based Power System Toolboxes

- Comparison of Matlab-based power system analysis softwares:

Package	PF	CPF	OPF	SSA	TD	EMT	GUI	GNE
EST	✓			✓	✓			✓
MatEMTP					✓	✓	✓	✓
MatPower	✓		✓					
PAT	✓			✓	✓			✓
PSAT	✓	✓	✓	✓	✓		✓	✓
PST	✓	✓		✓	✓			
SPS	✓			✓	✓	✓	✓	✓
VST	✓	✓		✓	✓		✓	

Matlab-based Power System Toolboxes

- The features illustrated in the table are:
  - power flow (PF)
  - continuation power flow and/or voltage stability analysis (CPF-VS)
  - optimal power flow (OPF)
  - small signal stability analysis (SSA)
  - time domain simulation (TD)
  - graphical user interface (GUI)
  - graphical network editor (GNE).

Tools Currently not Available in Matlab

- The following tools are still not available as Matlab packages or toolboxes:
  - Fault analysis.
  - Grounding systems.
  - Harmonic analysis.
  - Protection analysis and coordination.
  - Etc.

Matlab vs. GNU/Octave

- An important but often missed issue is that the Matlab environment is a commercial and “closed” product, thus Matlab kernel and libraries cannot be modified nor freely distributed.
- To allow exchanging ideas and effectively improving scientific research, both the toolbox and the platform on which the toolbox runs should be free (Richard Stallman).
- An alternative to Matlab is the free GNU/Octave project.

Introduction to PSAT

Universidad de Castilla-La Mancha

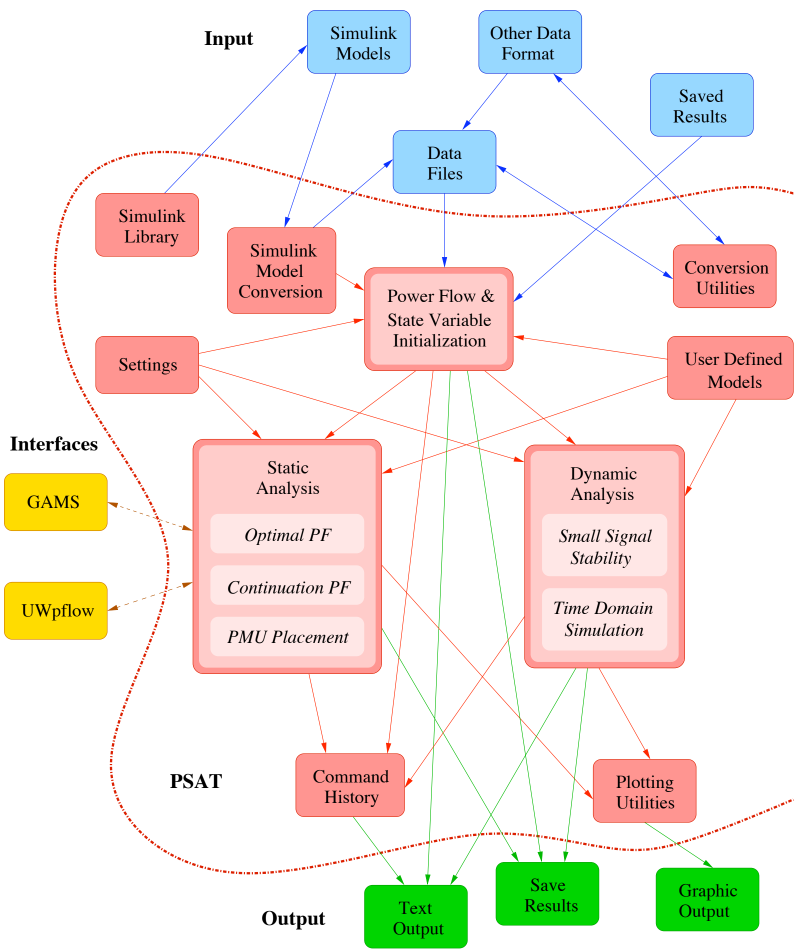
PSAT Features

- PSAT has been thought to be portable and open source.
- PSAT runs on the commonest operating systems
- PSAT can perform several power system analysis:
  1. Continuation Power Flow (CPF);
  2. Optimal Power Flow (OPF);
  3. Small signal stability analysis;
  4. Time domain simulations.

PSAT Features

- PSAT deeply exploits Matlab vectorized computations and sparse matrix functions in order to optimize performances.
- The latest beta version of PSAT also exploits Matlab classes to be more versatile and to ease maintenance and extensions.
- PSAT also contains interfaces to UWPFLOW and GAMS which highly extends PSAT ability to solve CPF and OPF problems, respectively.

Synoptic Scheme



Universidad de Castilla-La Mancha

PSAT Features

- In order to perform accurate and complete power system analyses, PSAT supports a variety of static and dynamic models.
- Dynamic models include non conventional loads, synchronous machines and controls, regulating transformers, FACTS, wind turbines, and fuel cells.

PSAT Features

- Besides mathematical algorithms and models, PSAT includes a variety of additional tools, as follows:

    1. User-friendly graphical user interfaces;
    2. Simulink library for one-line network diagrams;
    3. Data file conversion to and from other formats;
    4. User defined model editor and installer;
    5. Command line usage.

Getting Started

- PSAT is launched by typing at the Matlab prompt:

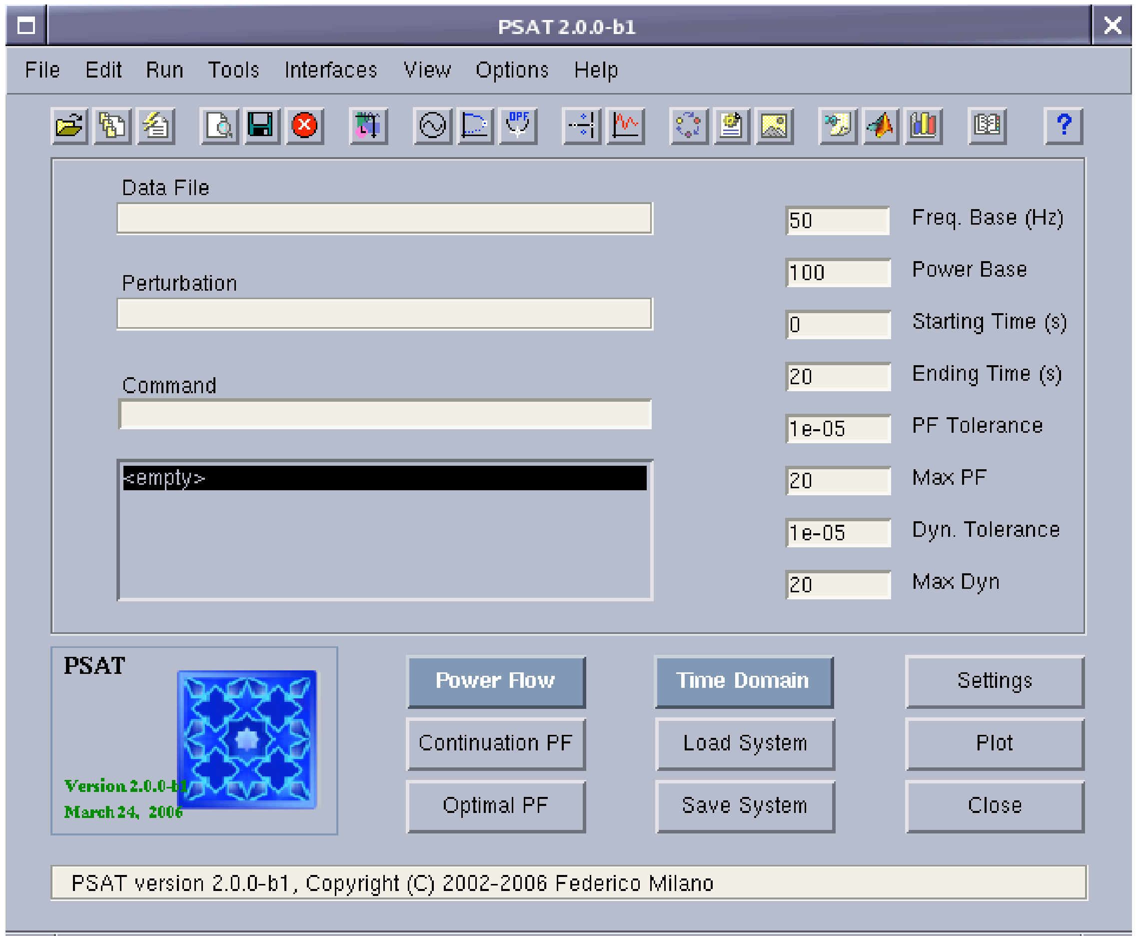
```
>> psat
```

  which will create all structures required by the toolbox and open the main GUI.

- All procedures implemented in PSAT can be launched from this window by means of menus, buttons and/or short cuts.

Getting Started

- Main PSAT GUI:

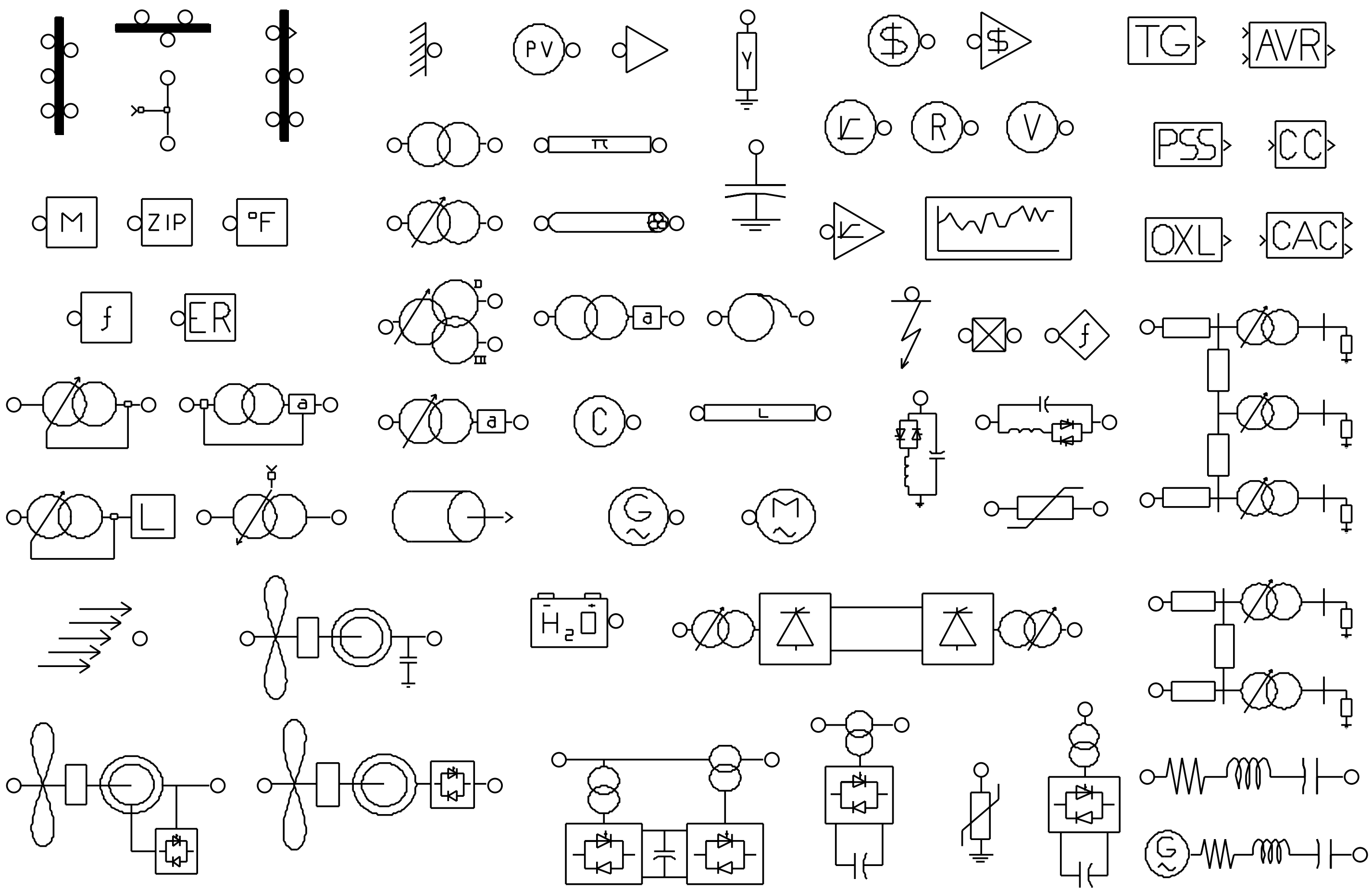


Simulink Library

- PSAT allows drawing electrical schemes by means of pictorial blocks.
- The PSAT computational engine is purely Matlab-based and the Simulink environment is used only as graphical tool.
- A byproduct of this approach is that PSAT can run on GNU/Octave, which is currently not providing a Simulink clone.

Simulink Library

- PSAT-Simulink Library:

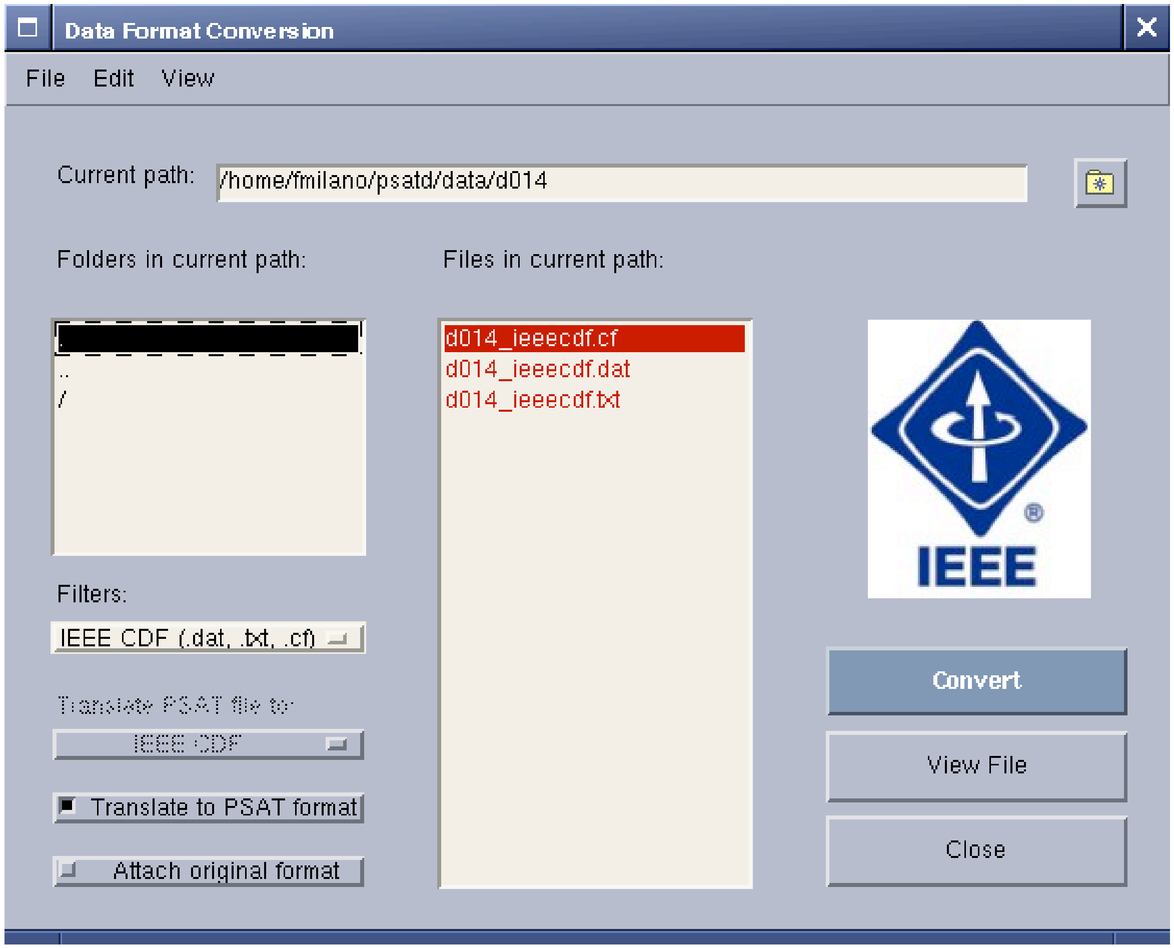


Other Features

- To ensure portability and promote contributions, PSAT is provided with a variety of tools, such as a set of Data Format Conversion (DFC) functions and the capability of defining User Defined Models (UDMs).
- The set of DFC functions allows converting data files to and from formats commonly in use in power system analysis. These include: IEEE, EPRI, PTI, PSAP, PSS/E, CYME, MatPower, PST, etc. formats. On Matlab platforms, an easy-to-use GUI handles the DFC.

Data Format Conversion

- GUI for data format conversion:

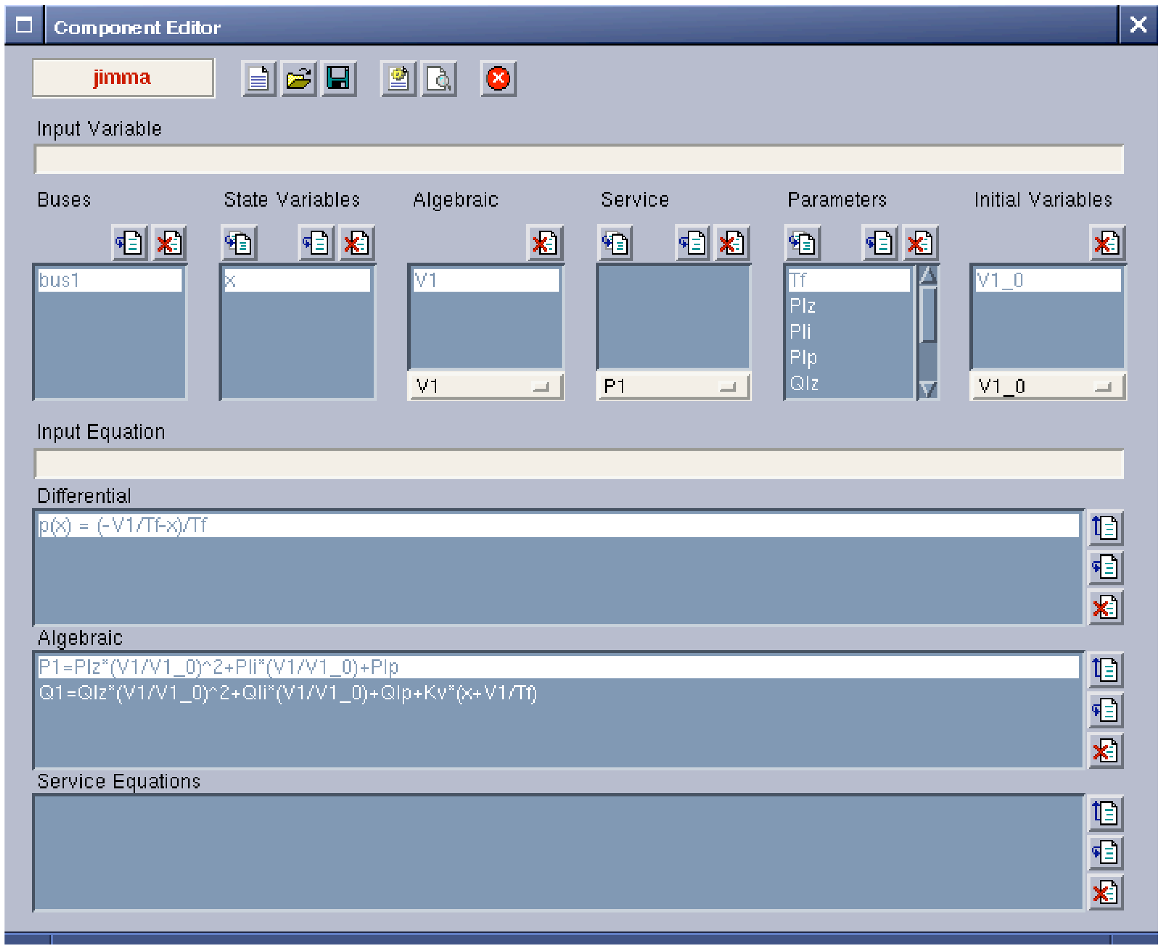


User Defined Models

- The UDM tools allow extending the capabilities of PSAT and help end-users to quickly set up their own models.
- Once the user has introduced the variables and defined the DAE of the new model in the UDM GUI, PSAT automatically compiles equations, computes symbolic expression of Jacobians matrices and writes a Matlab function of the new component.
- Then the user can save the model definition and/or install the model in PSAT.
- If the component is not needed any longer it can be uninstalled using the UDM installer as well.

User Defined Models

- GUI for user defined models:



Universidad de Castilla-La Mancha

Command Line Usage

- PSAT is provided with a command line version. This feature allows using PSAT in the following conditions:

1) If it is not possible or very slow to visualize the graphical environment (e.g. Matlab is running on a remote server).
2) If one wants to write scripting of computations or include calls to PSAT functions within user defined programs.
3) If PSAT runs on the GNU/Octave platform, which currently neither provides GUI tools nor a Simulink-like environment.

Power System Model

- The standard power system model is basically a set of nonlinear differential algebraic equations, as follows:

$$\begin{aligned} \dot{x} &= f(x, y, p) \\ 0 &= g(x, y, p) \end{aligned}$$

where x are the state variables $x \in \mathbb{R}^n$; y are the algebraic variables $y \in \mathbb{R}^m$; p are the independent variables $p \in \mathbb{R}^\ell$; f are the differential equations $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^\ell \mapsto \mathbb{R}^n$; and g are the algebraic equations $g : \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^\ell \mapsto \mathbb{R}^m$.

Power System Model

- PSAT uses these equations in all algorithms, namely power flow, CPF, OPF, small signal stability analysis and time domain simulation.
- The algebraic equations g are obtained as the sum of all active and reactive power injections at buses:

$$g(x, y, p) = \begin{bmatrix} g_p \\ g_q \end{bmatrix} = \begin{bmatrix} g_{pm} \\ g_{qm} \end{bmatrix} - \sum_{c \in \mathcal{C}_m} \begin{bmatrix} g_{pc} \\ g_{qc} \end{bmatrix} \quad \forall m \in \mathcal{M}$$

where g_{pm} and g_{qm} are the power flows in transmission lines, $\mathcal{M}$ is the set of network buses, $\mathcal{C}_m$ and $[g_{pc}^T, g_{qc}^T]^T$ are the set and the power injections of components connected at bus m, respectively.

Component Models

- PSAT is component-oriented, i.e. any component is defined independently of the rest of the program as a set of nonlinear differential-algebraic equations, as follows:

$$
\begin{aligned}
\dot{x}_c &= f_c(x_c, y_c, p_c) \\
P_c &= g_{pc}(x_c, y_c, p_c) \\
Q_c &= g_{qc}(x_c, y_c, p_c)
\end{aligned}
$$

where x_c are the component state variables, y_c the algebraic variables (i.e. V and θ at the buses to which the component is connected) and p_c are independent variables. Then differential equations f are built concatenating f_c of all components.

Component Models

- These equations along with Jacobians matrices are defined in a function which is used for both static and dynamic analyses.
- In addition to this function, a component is defined by means of a structure, which contains data, parameters and the interconnection to the grid.

Component Models: Example

- Let's consider the exponential recovery load (ERL).
- The set of differential-algebraic equations are as follows:

$$
\begin{aligned}
\dot{x}_{c_1} &= -x_{c_1}/T_P + P_0(V/V_0)^{\alpha_s} - P_0(V/V_0)^{\alpha_t} \\
\dot{x}_{c_2} &= -x_{c_2}/T_Q + Q_0(V/V_0)^{\beta_s} - Q_0(V/V_0)^{\beta_t} \\
P_c &= x_{c_1}/T_P + P_0(V/V_0)^{\alpha_t} \\
Q_c &= x_{c_2}/T_Q + Q_0(V/V_0)^{\beta_t}
\end{aligned}
$$

where and P_0, Q_0 and V_0 are initial powers and voltages, respectively, as given by the power flow solution.

- Observe that a constant PQ load must be connected at the same bus as the ERL to determine the values of P_0, Q_0 and V_0.

Component Models: Example

➠ Component Data:

Column	Variable	Description	Unit
1	-	Bus number	int
2	S_n	Power rating	MVA
3	V_n	Active power voltage coefficient	kV
4	f_n	Active power frequency coefficient	Hz
5	T_P	Real power time constant	s
6	T_Q	Reactive power time constant	s
7	α_s	Static real power exponent	-
8	α_t	Dynamic real power exponent	-
9	β_s	Static reactive power exponent	-
10	β_t	Dynamic reactive power exponent	-

Component Models: Example

- Exponential recovery loads are defined in the structure `Erload`, whose fields are as follows:

1. `con`: exponential recovery load data.
2. `bus`: Indexes of buses to which the ERLs are connected.
3. `dat`: Initial powers and voltages (P_0, Q_0 and V_0).
4. `n`: Total number of ERLs.
5. `xp`: Indexes of the state variable x_{c_1}.
6. `xq`: Indexes of the state variable x_{c_2} .

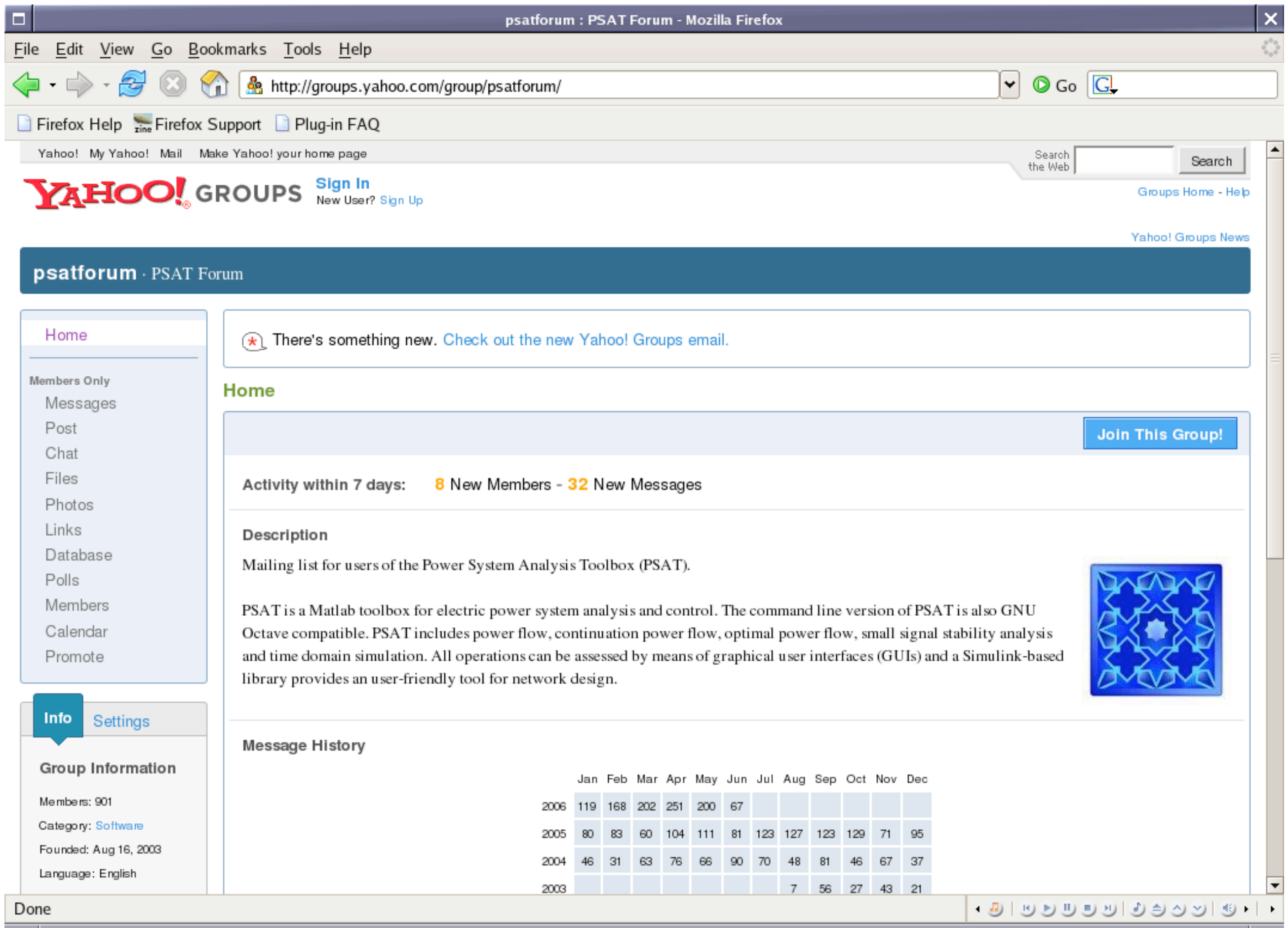
PSAT Forum

- The Forum is a mailing list open to all PSAT users.
- The Forum is available at:

  http://groups.yahoo.com/group/psatforum/

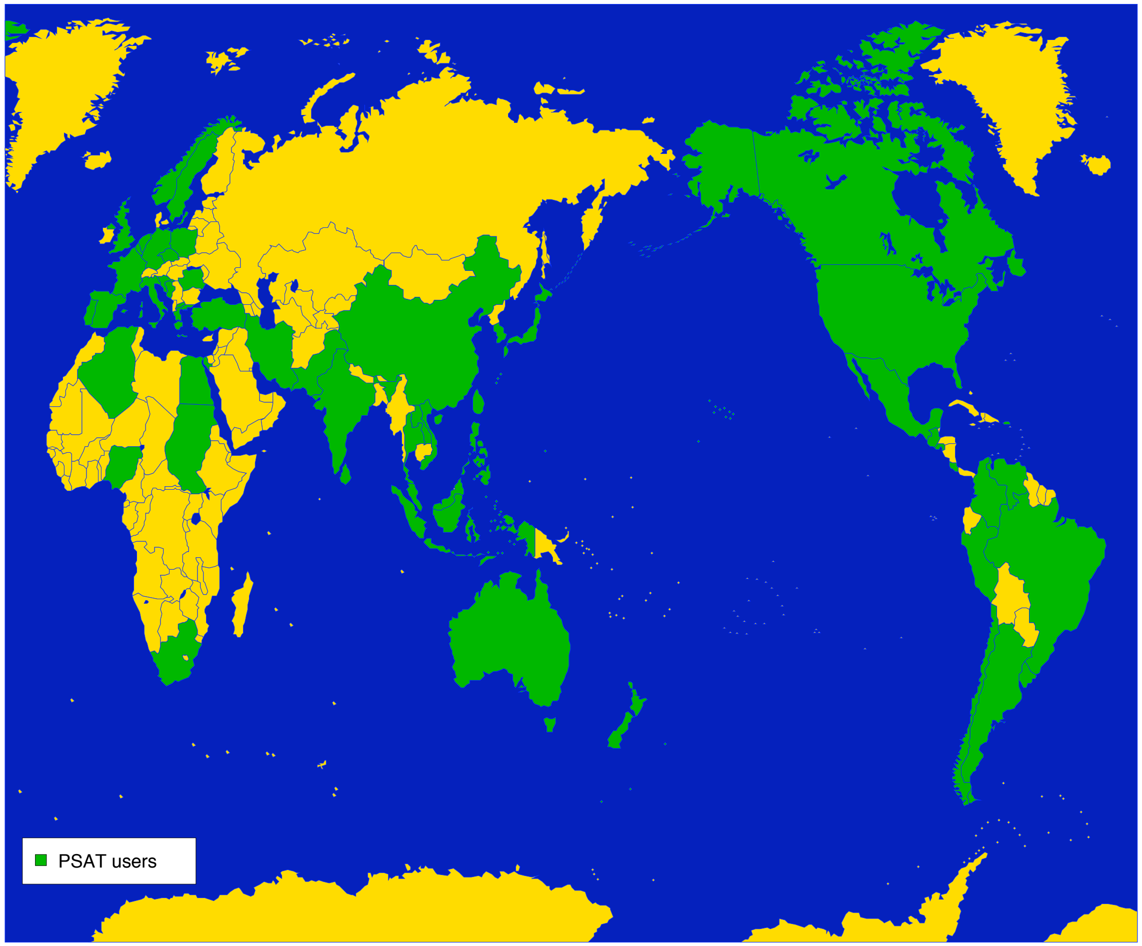
- Currently the Forum counts more than 750 members.

PSAT Forum



Universidad de Castilla-La Mancha

PSAT in the World



Universidad de Castilla-La Mancha

Links

- Where to get PSAT:

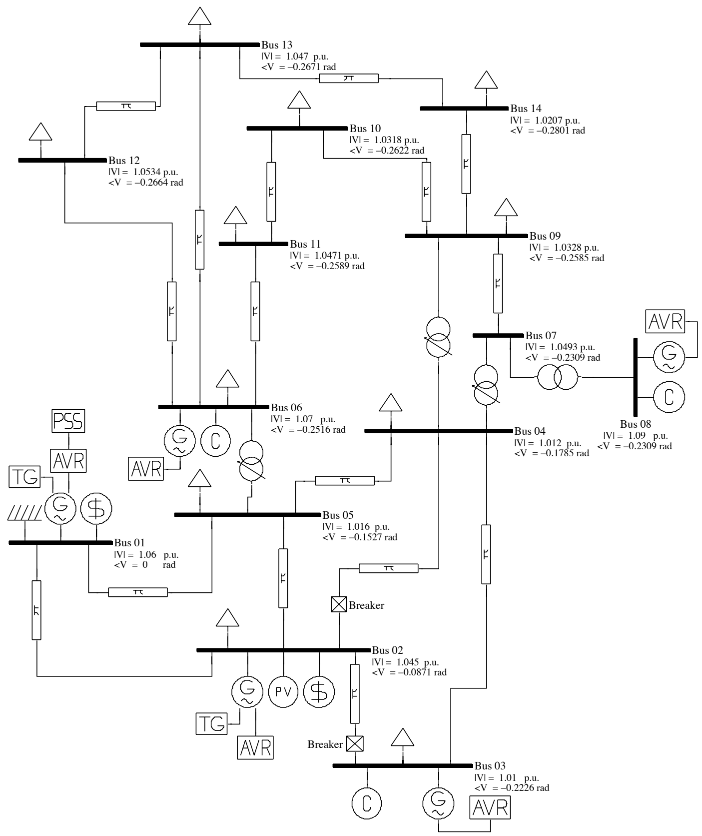
http://www.power.uwaterloo.ca/∼fmilano/

http://groups.yahoo.com/group/psatforum/

PSAT Examples

IEEE 14-bus Test System

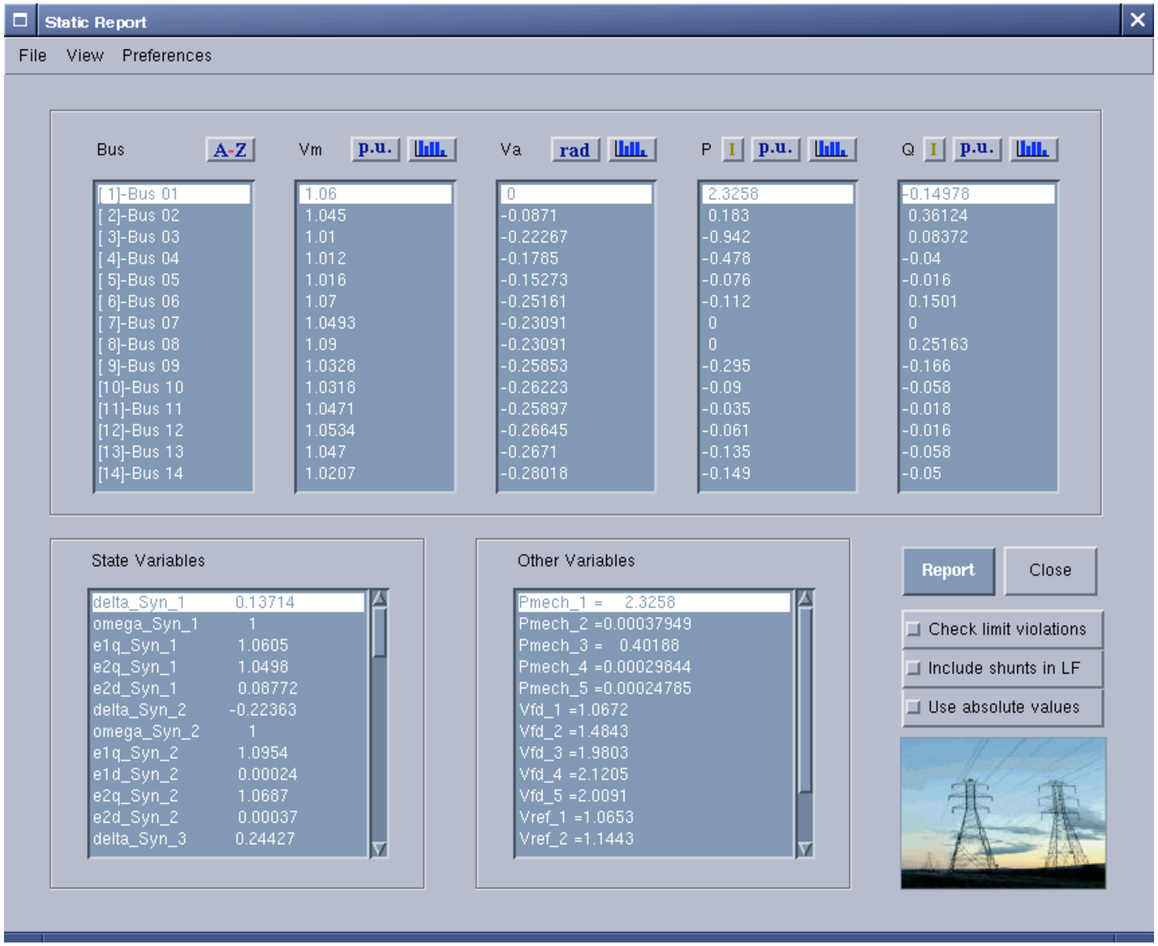
- IEEE 14-bus test system:



Universidad de Castilla-La Mancha

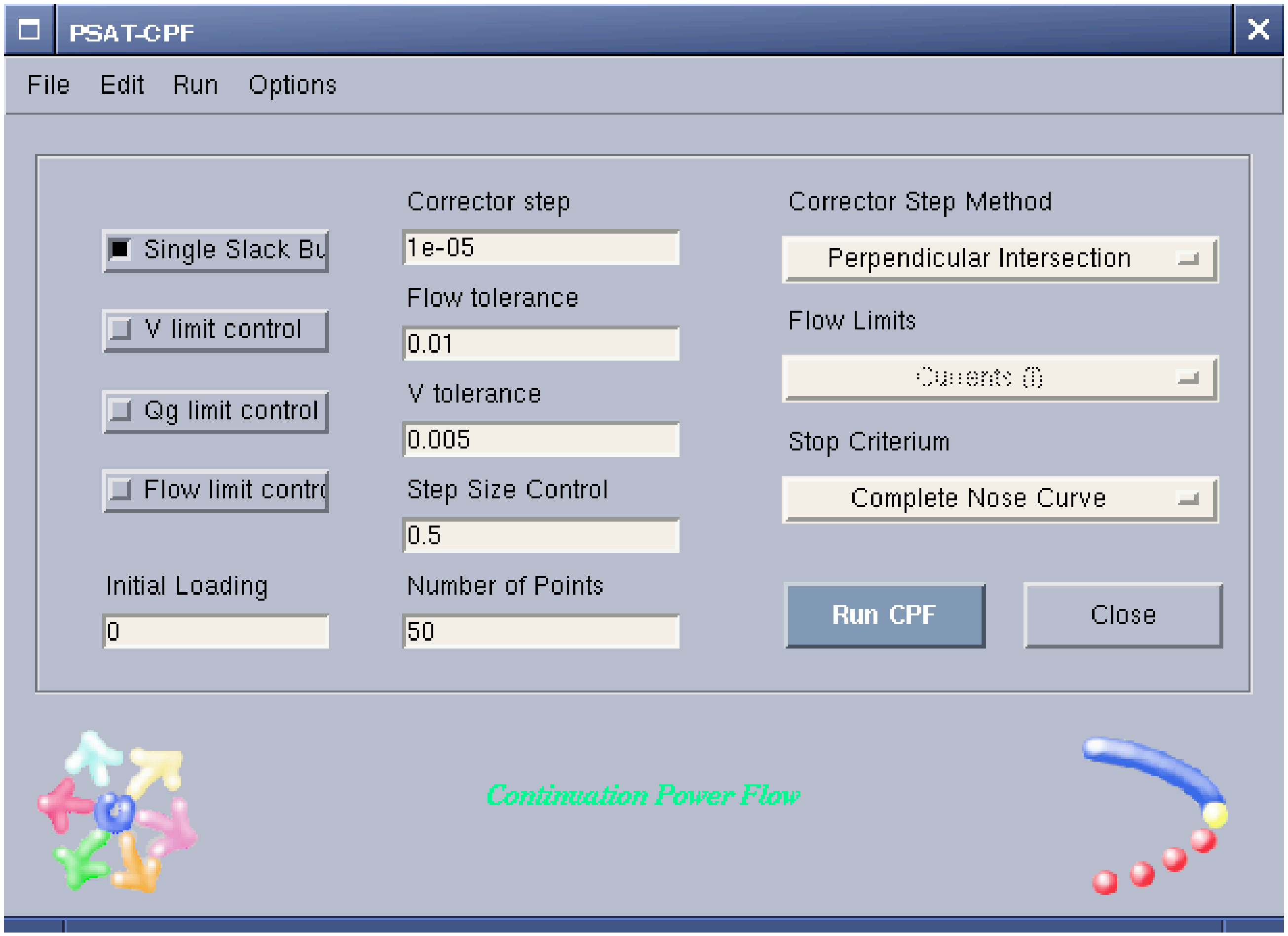
IEEE 14-bus Test System

- Power flow report:



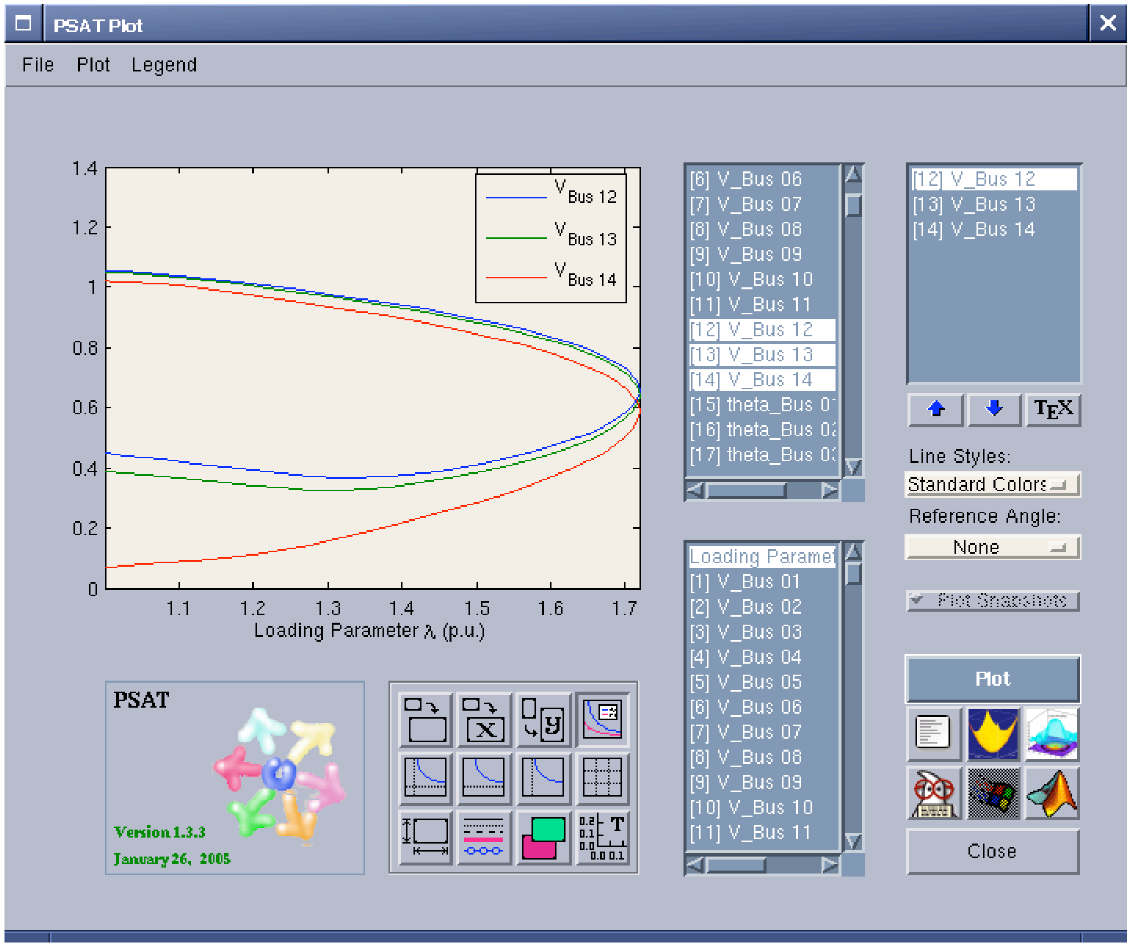
IEEE 14-bus Test System

- Continuation power flow analysis (GUI):



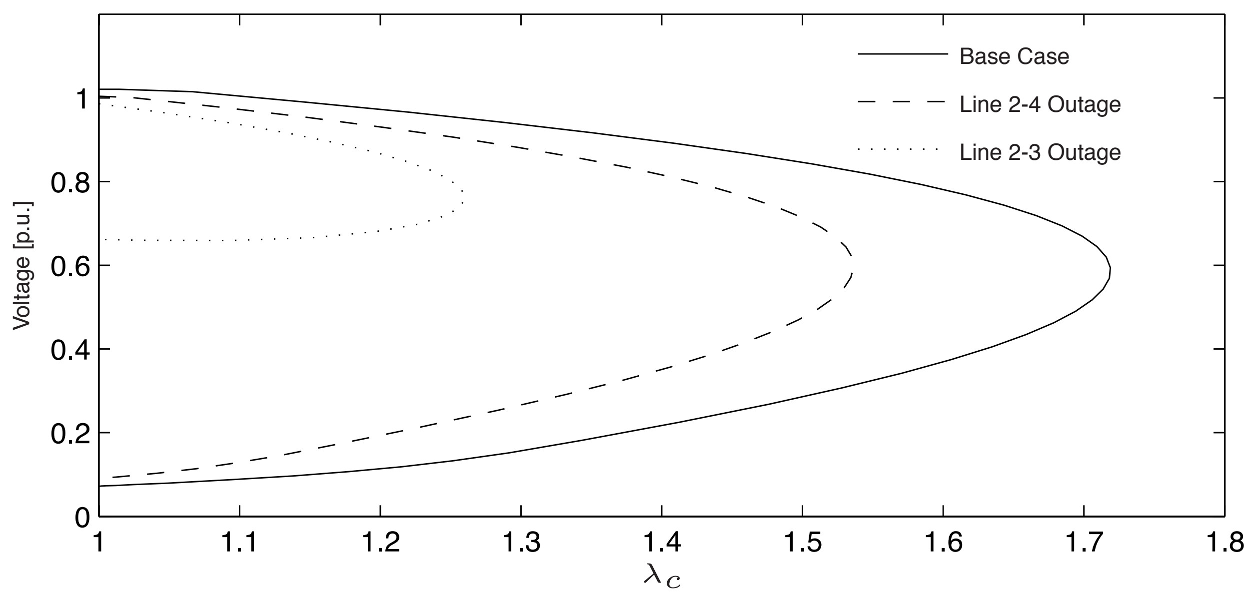
IEEE 14-bus Test System

- Continuation power flow analysis (plots):



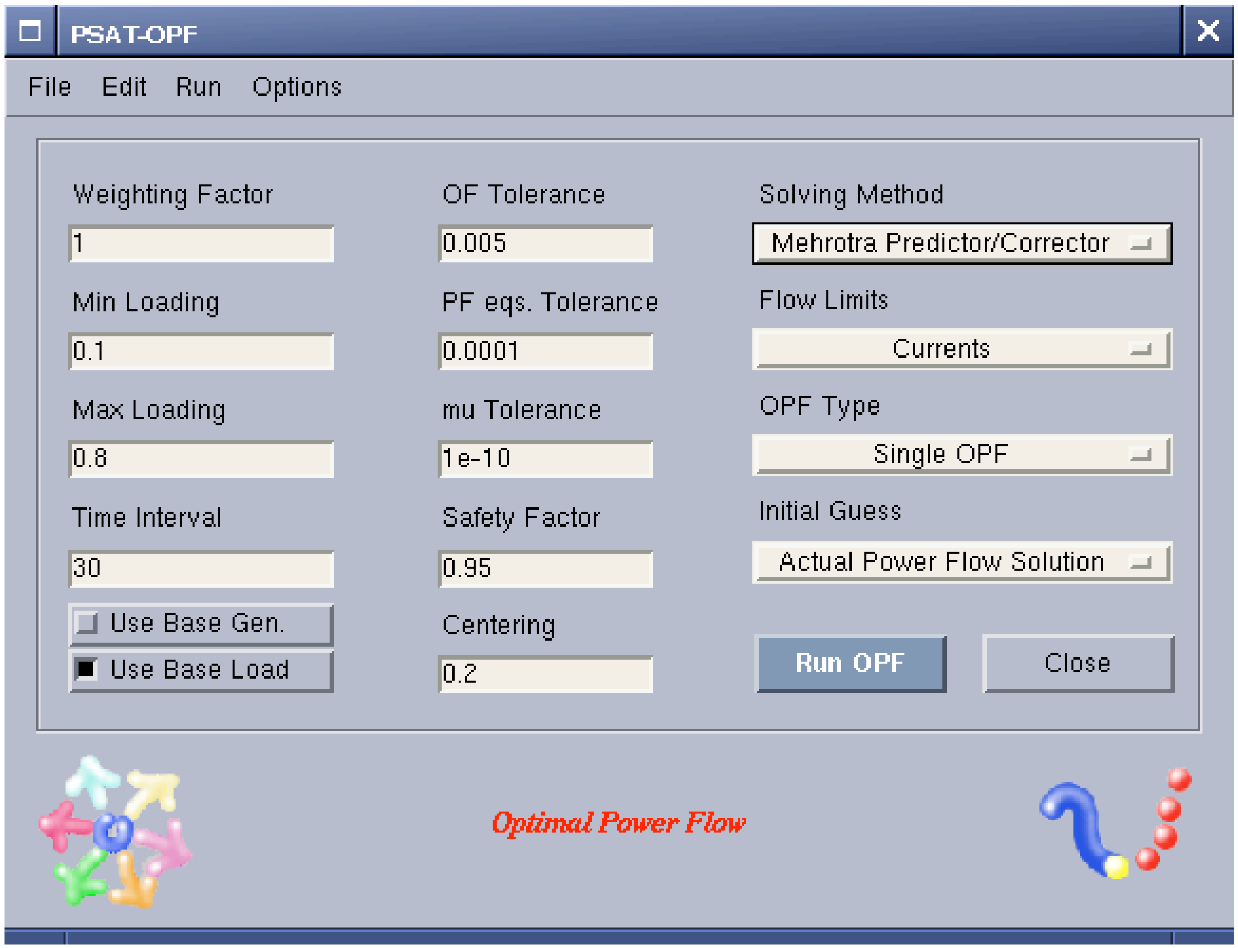
IEEE 14-bus Test System

- Nose curves at bus 14 for different contingencies for the IEEE 14-bus test system:



IEEE 14-bus Test System

- Optimal power flow analysis (GUI):



IEEE 14-bus Test System

- Comparison between OPF and CPF analysis:

Contingency	BCP [MW]	λ^* [p.u.]	MLC [MW]	ALC [MW]
None	259	0.7211	445.8	186.8
Line 2-4 Outage	259	0.5427	399.5	148.6
Line 2-3 Outage	259	0.2852	332.8	73.85

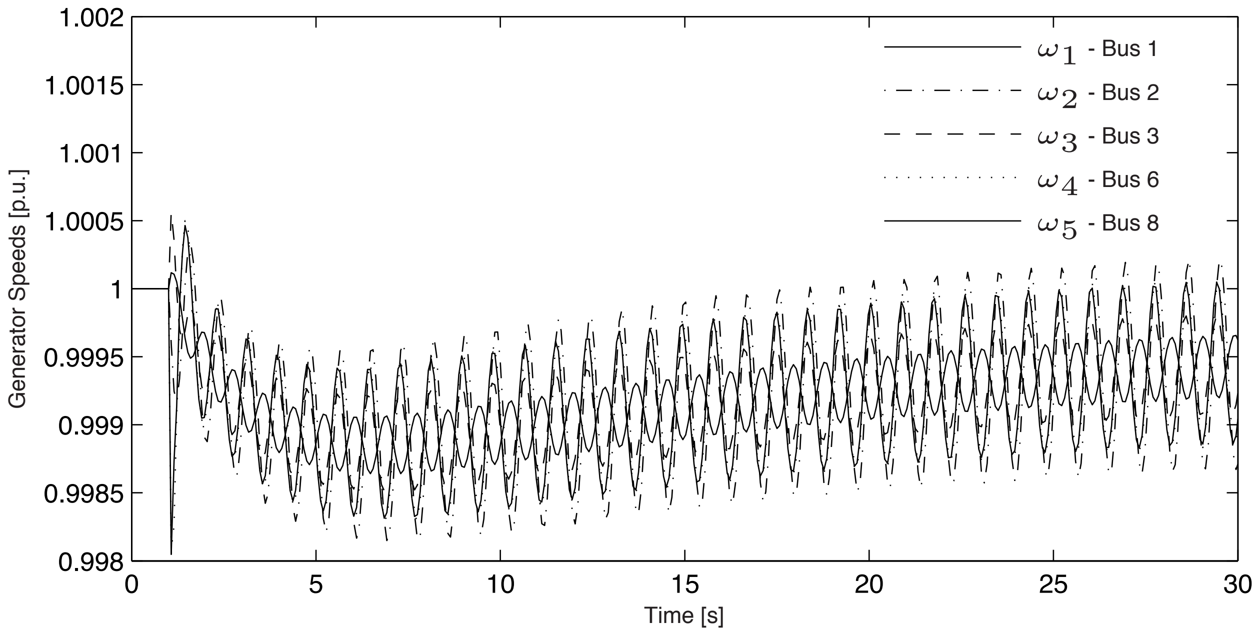
- Because of the definitions of generator and load powers P_G and P_L, one has $\lambda_c = \lambda^* + 1$.

IEEE 14-bus Test System

- Time domain simulation:
  - It has been used a 40% load increase with respect to the base case loading, and no PSS at bus 1. A Hopf bifurcation occurs for the line 2-4 outage resulting in undamped oscillations of generator angles.
  - A similar analysis can be carried on the same system with a 40% load increase but considering the PSS of the generator connected at bus 1. In this case the system is stable.

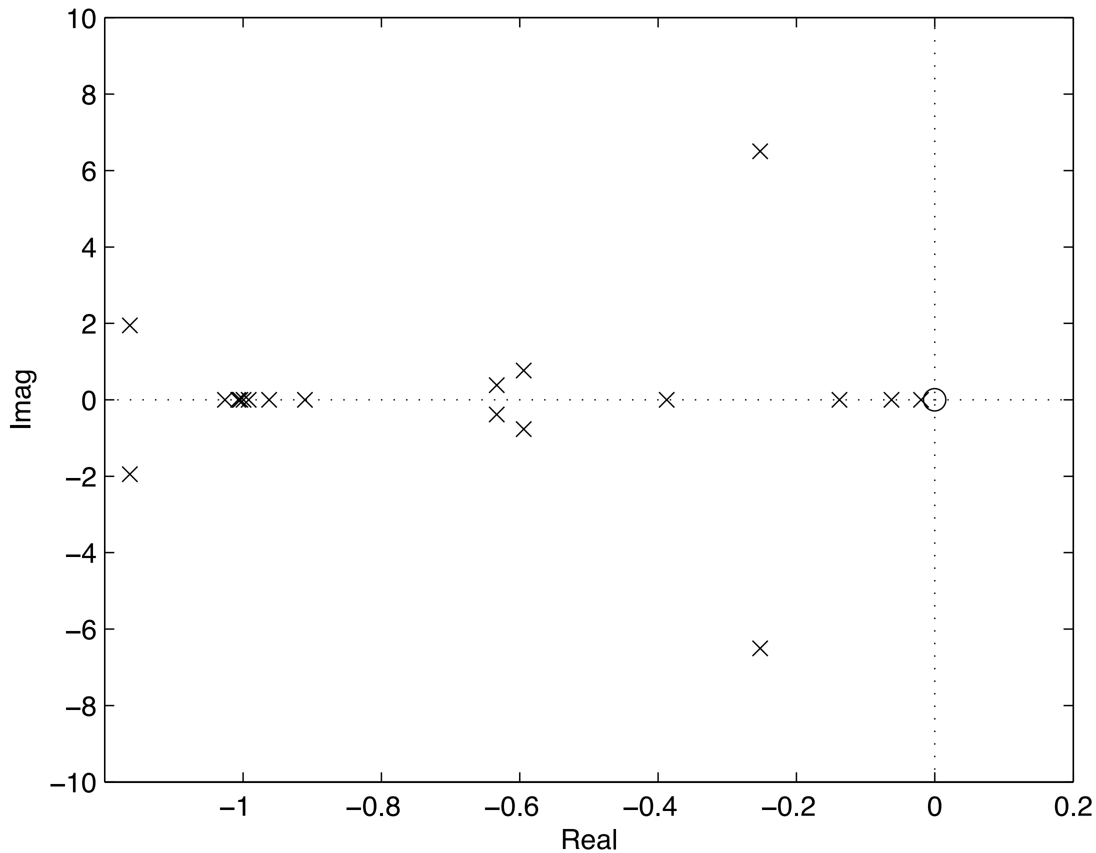
IEEE 14-bus Test System

- Time domain simulation (without PSS):



IEEE 14-bus Test System

- Eigenvalue analysis (with PSS):



Conclusions

What is PSAT for?

- PSAT is ideal for teaching and for explaining basic power system analysis and control concepts.
- PSAT is useful for research purposes because allows easy and fast prototyping of new models and algorithms.
- PSAT is also useful for creating bridges between Matlab and other specialized software packages (e.g. GAMS).

Some Current Advanced Usages of PSAT

- Prototyping of OPF-based market clearing algorithms for the 15000-bus WECC network. Collaboration with NCPA (Northern California Power Agency) and University of Waterloo, Canada.
- Study of the impact of increasing wind generation in the Castilla-La Mancha subtransmission network. Contract with the local government of Castilla-La Mancha.

Thanks for your attention!

Questions?