COMPARTMENTALISED VALIDATION OF MODELS FOR COMPLEX NONLINEAR DYNAMIC SIMULATORS USING A TRUSTED THIRD PARTY GENERIC SIMULATOR.

G. H. BENSON, AMIEE
University of Bath, UK
email: G.H.Benson@bath.ac.uk

R. W. DUNN, MIEEE
University of Bath, UK
email: R.W.Dunn@bath.ac.uk

ABSTRACT

Direct comparisons between different simulators can be misleading, especially when the simulators are modelling complex high-order systems. Any small section being examined can be affected by differences emanating from completely unrelated sections of the model.

A method for isolating small sections of complex simulations for the purpose of testing is presented, and its use in the context of simulator validation is described.

This technique involves the creation and validation of an identical replica of the section of interest without approximation and without the need to replicate the entire model.

Keywords: power system simulation, model validation

INTRODUCTION

We live in a world in which the stable and secure supply of electricity is of paramount importance. The demand for electrical power is constantly increasing, but the expansion and reinforcement of the transmission system is often restricted by external influences. Environmental concerns can hinder or prevent projects to upgrade the network, especially in cases that involve the construction of new transmission lines.

This growing demand for power must be met without compromising the quality of the supply – the voltage and frequency must lie within stringent limits at every point of supply. Since network improvements are so often ruled out, the operators are forced to drive the power system ever closer to its limits.

There are other factors which make operating at the limit desirable; in the UK system, for example, there are several ‘weak links’ – critical interconnections which cannot be reinforced – along which cheaper electricity can be sourced. By reducing uncertainty, by knowing more accurately where the stability limits of the system lie, these weak interconnections can be used to the maximum possible extent. The resulting gains in efficiency are beneficial to all.

Time domain simulation

In order to deal with the above scenario, some method must be developed to analyse the system. Realistic models of the generation sets contain many nonlinearities, due to their underlying nature (e.g., magnetic saturation in the generator) and due to the design of their controllers (e.g., limits and hysteresis). In addition, modern power systems are heavily interconnected, so that the supply can be maintained in the event of loss of plant. These two factors interact to form a highly complex, nonlinear problem space which resists analysis by the more traditional exact analytical approaches. These may be desirable due to their low computational requirements, but in order to apply them it is necessary to make several gross approximations of the problem space.

Of course, no method of analysis encompasses every single factor; one aims to consider only those factors which are important for the required level of accuracy and discard the rest. However, the approximations that must be applied to allow the use of these so-called ‘exact’ methods tend to obscure the rich complexity of the system under investigation. It is widely regarded that time domain simulation is the only way to model and analyse the system at any more than a superficial level.

The complexity of modern power systems cannot be understated. The power system in England and Wales, for example, contains in excess of 200 generators, 1100 busbars (the nodes in the network) and 1800 transmission lines. Each generator is modelled using between 7 and 30 differential equations coupled with nonlinear algebraic equations. This conspires to make power system simulation a very computationally demanding task.

The power system simulator PSSENG

The University of Bath has been developing power system simulation software since 1986. Initially the software was designed for custom multiprocessor hardware and later adapted to run on standard PCs since these

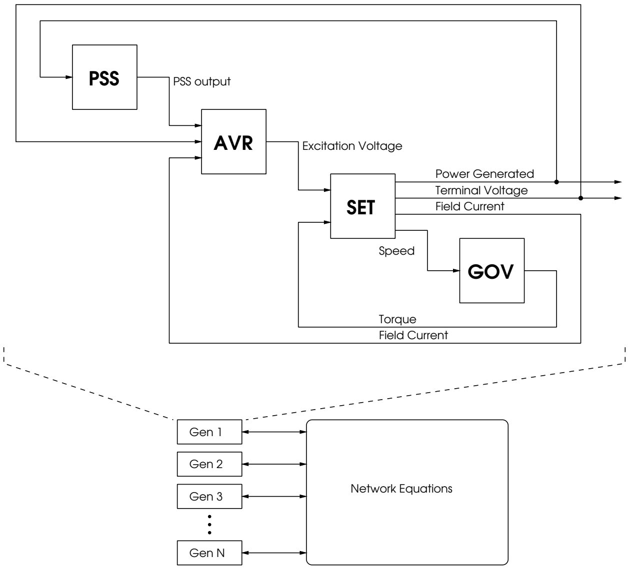

Figure 1: The overall hierarchy within PSSENG.

became available with the power of the previous generation's supercomputers. The current incarnation of this simulator is known as the Power System Simulation Engine (PSSENG), and because of its heritage has extremely modest hardware requirements. The full England and Wales system, as outlined above, can be simulated at more than 3 times faster than real time on a single 300MHz Pentium II PC, and at over 7 times faster on a single 650MHz Athlon PC.

Much work has been carried out at the University of Bath, using PSSENG, in both the fields of optimising power flows for better use of the network, and in assessing the stability of systems both for planning [1] and in an on-line environment [2, 3]. All of this work requires confidence that PSSENG is indeed operating as it should.

Overview of PSSENG

To simplify the problem of simulating power systems, it is traditional to separate the models of the transmission system and the generators. Each generator is then treated as an individual system, whose inputs and outputs link it to the network equations and from there on to the other generators.

A simulation, therefore, consists of N generators and the network of transmission lines that connect them together. The latter is in principle the easier of the two to model, since it is described by simple linear algebra. The parameters which describe the network are formed into an *admittance matrix* $\mathbf{Y}$ and the currents may be found by solving the vector equation $\underline{V} = \mathbf{Y}^{-1}\underline{I}$, where $\underline{V}$ is the vector of voltages and $\underline{I}$ is the vector of currents. Although simple to describe, in practice this is a non-trivial equation to solve, since the dimension of the matrix $\mathbf{Y}$ is the number of busbars in the network: 1100 for the England and Wales system, as described above.

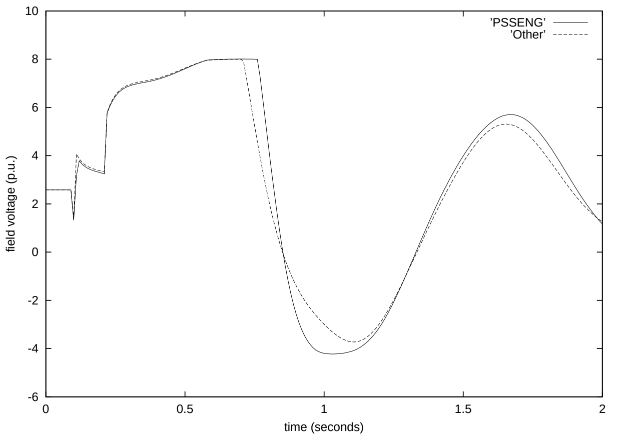

Figure 2: Example comparison with an anomaly at about 0.7s.

The generators are more complex to model since they are high-order systems, often with many nonlinearities. The core of a generator model is the *machine set*: in PSSENG, this is modelled using a 6^{th} order *voltage behind subtransient reactance* model, based on work by Berry [4]. Detailed knowledge of this model is not important for the purposes of this paper.

Each machine set is accompanied by a number of items of control equipment, such as a speed governor to regulate the frequency, an automatic voltage regulator (AVR) to regulate the terminal voltage and a power system stabiliser (PSS) to suppress oscillations within the transmission network. Figure 1 shows a schematic representation of this overall hierarchy. Each block – PSS, AVR, machine set and governor – can be further subdivided into its component parts until one is left with only integrators, gains and nonlinearities.

Figure 3 shows an example of an AVR model that illustrates the level of detail involved.

The problems of validating models

Whilst they are simulated as a single system, the sections of the generator model – PSS, AVR, machine and governor – are specified individually, to allow a generator model to be built from any combination of pieces. When a new model of a section is implemented in the simulator, it is desirable to compare the results of simulations from PSSENG with the results of other simulators.

Now, no two simulators are alike; different schools of thought will make different design decisions and different approximations. It should be no surprise, therefore, that the results are subtly different when one attempts to validate a model in this way. Usually the differences are inconsequential to the final result, but occasionally a result may vary in a more significant

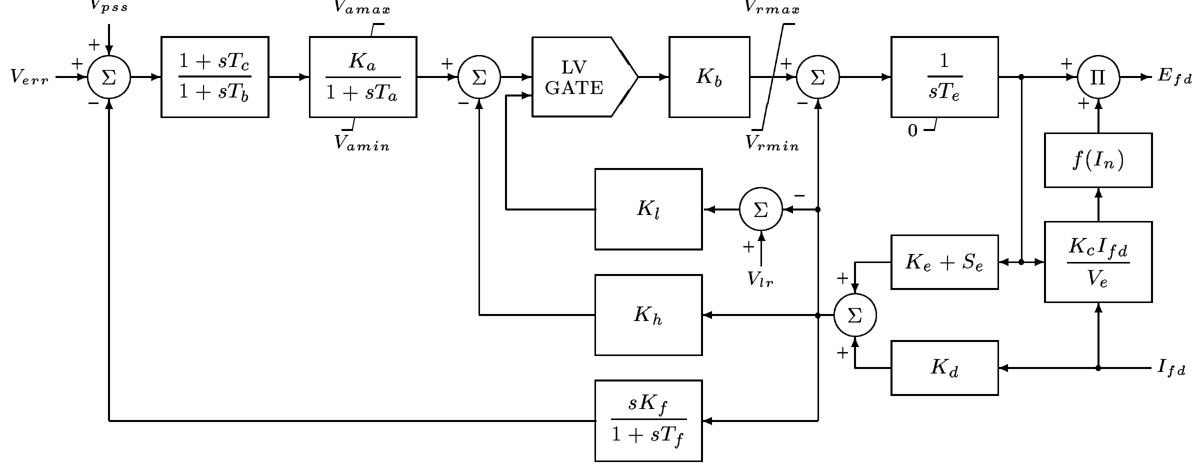

Figure 3: A typical AVR model: the IEEE AC2 standard AVR model [5].

manner. Figure 2 shows such an example – the curves are an excellent match until about 0.7 seconds, when they diverge markedly.

It is not obvious whether the differences emanated from the new model under test, from elsewhere in the generator model, or from the network or another generator in the simulation. The difficulty in tracing the source of the error is exacerbated by the fact that power system simulators often do not output the internal states from the generator model: usually only the values shown in figure 1 are actually output, since any model-specific internal values are generally of no interest to the end users of the simulators.

Some simulators, including PSSENG, allow varying degrees of access to the internal states of the model, but unfortunately this luxury did not extend to any of the third-party simulators involved in the tests. It became apparent that a method was required to infer reliable values for the internal states from the third-party simulators, without having to emulate the entire power system model.

This work

A technique for simulator comparisons was devised using a third-party generic simulation package. This paper describes the method, and is not tied to any specific package; in this case MATLAB was used, with its sister package Simulink, both produced by The MathWorks inc.

THE TECHNIQUE

When generating curves for comparison between simulators, it is useful to generate data using a representation of the system known as a single machine, infinite bus (SMIB) system, shown in figure 4. The SMIB system contains one generator set, the one under test, and an 'infinite bus', which is treated as a machine so large that the generator under test can have no effect on either its voltage or frequency.

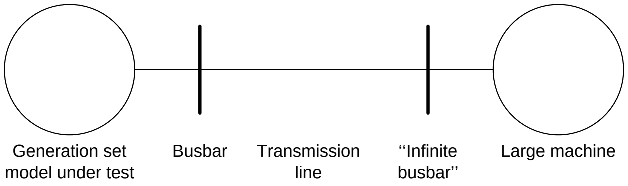

Figure 4: The SMIB system.

The SMIB system was originally created as a mathematical tool to facilitate the analysis and design of a machine and its controller, but unfortunately it is often misused as an approximation to a real power system. In this case, however, we are simply comparing simulators as opposed to attempting to model a real system, and the SMIB system has several advantages: there are fewer possible sources of difference, and furthermore the machine is less well damped than it would be if connected to a more realistic system model. This reduced damping serves to stress the simulators more, and thus exaggerate any differences.

When comparing results, it is to be noted that the subtle differences are found in *all* of the measured signals. Thus, the section of model that is being validated is producing different output signals from different input signals. Clearly, although differences between the simulators are evident, it may be that the section under test is performing identically in both simulators, and the only differences are those that have propagated through from other sections.

Building the replica

The technique for validation hinges around the fact that the system being simulated is *deterministic* – given the same initial conditions and the same input sig-

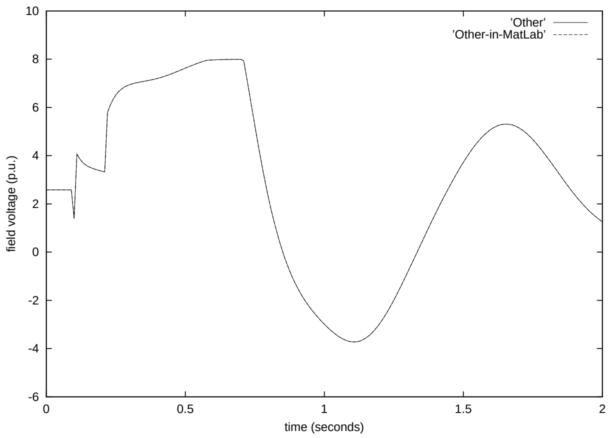

Figure 5: Comparison of reference and replica.

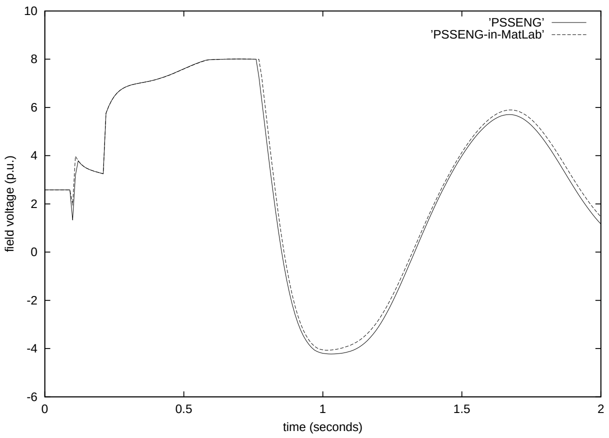

Figure 6: Comparison of replica and PSSENG.

nals, the time histories of the outputs should always be the same. For example, if a new AVR model was being tested, one could record the time histories of the PSS output V_{PSS}, the busbar voltage error V_{ERR} and the field current I_{FD}, and the initial conditions within the model. If these values were fed into the simulator again, the outputs of the model – in this case, the excitation voltage E_{FD} – should be *identical.*

That explanation of the determinism assumes one thing: the output signals will be identical if the initial conditions and input signals are applied to an *identical model.* Therefore, if an identical model is created in some other form, those same initial conditions and input signals should create the same output signals.

In this case, an identical model was created using MATLAB and Simulink, but the technique would be equally valid in any simulation package or even – with appropriate interfacing – an electrical equivalent circuit. Thus, by replicating only the small section of the power system model under test, a replica system has been created which is equivalent in every way.

Validating the replica

Once the replica has been created in the format of choice, it must be compared to the simulator which is being used as the reference. Input signals and initial conditions from the reference simulator are fed into the replica, and the outputs compared. Unlike the comparison between the two power system simulators, the model in the reference simulator and the replica have the same input signals, so the same output signal is generated in each case (assuming the replica is correct). An example reference-replica comparison is shown in figure 5 – the difference is several orders of magnitude smaller that that in figure 2, so small in fact that it is not discernible to the eye. With a pair of curves like that of figure 5, it is highly unlikely that the replica is incorrect, although one must not be overconfident – there may be features such as limits which are not hit in that particular run. A small amount of testing is required to confirm that all is as it should be.

Using the replica

Once the replica has been validated, one is in possession of an equivalent system to the reference simulator. Recall that the reference simulator is a third party package, and only a limited number of values can be monitored within the model itself. The replica, which we know to be equivalent, is under our control, and as many signals can be measured as is desired. A further advantage is that any ambiguities about the exact implementation of the model should have been cleared up by this point, which speeds up the process of model implementation.

We know that the replica is the equivalent of the reference, so we can now use it to validate the simulator that we are testing. If the input signals and initial conditions are recorded from the simulator under test and then fed into the replica, then the output signals from the test simulator and the replica should be the same. Figure 6 shows such a comparison, from which we can draw the following conclusions:

1. Most of the difference shown in figure 2 comes from elsewhere in the power system model, and thus no amount of searching within this section of the model will reveal its source.

2. There are slight differences at around 0.2 seconds, but further examination revealed these to be artifacts of the graph plotting method and not the simulator itself.

3. At around 0.8 seconds, the curves diverge. Essentially, some signal within PSSENG came out of limit one timestep earlier than its equivalent in the replica.

4. No conclusions can be drawn after this first real difference, since it has affected all future values.

In this particular case, if one had looked only at figure 2, one would conclude that the section of the model was incorrectly implemented. By using this new technique, one can see that this section is not the dominant source of the difference between the two simulators. The difference, therefore, must emanate from elsewhere in the system and can be methodically tracked down by using the same technique on different sections of the generator model. Finally, if the difference introduced by this section is judged to be significant, its cause can be located using the extra signals provided by the replica.

CONCLUSION

The obvious advantage of this technique is that the source of any differences between two simulations may be tracked down without requiring exhaustive amounts of data which may not be available.

A more subtle advantage is this: from the author's experience, small differences may build up in many places between two simulations. If the inputs to a section of a model have been through several sections prior to the section under test, then these many small differences can form a large difference to the outputs of the section. The differences from previous sections may swamp any differences which have been introduced by the section under test, and these could go unnoticed. This technique strips away differences from previous sections, exposing any new differences to the tester. Differences between simulators can thus be isolated one by one and reduced or removed wherever possible.

Much of simulation is a compromise between accuracy and speed. This technique can not only be used to compare simulators from different sources: it can be used to compare, for example, different methods of integration or different timestep lengths on the same simulator. With detailed analysis, many sources of error can be removed, and this may allow the simulation to either have the same degree of accuracy with reduced execution time, or have a higher level of accuracy for the same execution time.

The analysis can also be used to pinpoint the limits of accuracy within the simulator, for example the longest possible timestep length for a given controller before numeric instability sets in. Work is now in progress to find a method of quantifying the differences in a meaningful way, to enable automated analysis of generator models prior to their use in the simulator.

ACKNOWLEDGEMENTS

The authors are pleased to acknowledge the continuing support they have received from The National Grid Company for this and other research studies.

REFERENCES

[1] B.A. Nicholson, R.W. Dunn, K.W. Chan, and A.R. Daniels. A new technique for applying genetic algorithms to power system security optimisation problems. *Proceedings of IEEE Energy Management and Power Delivery*, supp:22–27, March 1998.

[2] A.R. Edwards, K.W. Chan, R.W. Dunn, and A.R. Daniels. Interactive online dynamic security assessment of large complex power systems part 2: Contingency screening. *Proceedings of IEEE Power Engineering Society*, Winter 1996.

[3] M.E. Bradley, A.O. Ekwue, F. Li, K.W. Chan, R.W. Dunn, and A.R. Daniels. On-line stability analysis for an operational tripping scheme monitor. *Proceedings of IEE Power System Control and Management*, IEE-421:71–75, April 1996.

[4] T. Berry, L.A. Dale, A.R. Daniels, and R.W. Dunn. Real-time modelling of multimachine power systems. *IEE Proceedings*, 140:241–248, 1993.

[5] IEEE Committee Report. Excitation system models for power system stability studies. *IEEE Transactions on Power Apparatus and Systems*, PAS-100(2):494–509, February 1981.