4

Real-Time Simulation Technologies in Engineering

Christian Dufour and Jean Bélanger

4.1 Introduction

Real-time simulators have always been used to design and test control and protection equipment performance before their installation on actual power systems. Protection and control equipment was usually interfaced with analogue benches emulating the real power systems in order to perform tests under realistic steady-state and faulty operating conditions. Actual control systems as well as analogue benches were, by definition, operating in real time, meaning that control and protection systems under test were exchanging signals at the same speed as when the tested equipment was installed on the actual power systems. The same technique was and is still applied in all industries including automotive, aircraft and aerospace.

Over the past two decades, commercially available computers have become both increasingly powerful and more affordable. This, in turn, has led to the emergence of highly sophisticated simulation software tools enabling high-fidelity real-time simulation of dynamic systems, as well as of fast electromagnetic transients (EMT) expected in electrical and power electronic systems. The consequence is that several expensive analogue test benches and simulators have been replaced by more affordable, very flexible and accurate, fully digital real-time simulators (RTS). The RTS can then be connected in closed-loop with control and protection systems under design and test, as was the case with former analogue simulators. This is called hardware-in-the-loop (HIL) simulation and test. Of course, the RTS must have the bandwidth and the very small latency required to simulate the high-frequency components of the phenomena that can affect the performance of the control and protection systems under test. The RTS must then have the ability to simulate all phenomena within a specified time-step, from μs to milliseconds, depending on the application, as discussed in following sections. But in all cases, the RTS must maintain its real-time performance, that is, all signals must be updated *exactly* at the specified time-step. Otherwise, an overrun or a failure to update its outputs and inputs at the specified time-step will cause unacceptable signal distortion affecting the performance of the equipment under test. Consequently, at the terminals of systems under test, we should not be able to see any difference between the signals generated by the simulator and the real plant systems within a specified bandwidth.

Transient Analysis of Power Systems: Solution Techniques, Tools and Applications, First Edition.
Edited by Juan A. Martinez-Velasco.

Automatic code generation tools are used in conjunction with the current generation of a real-time simulators for implementation in industrial controllers. These simulation software tools, including off-line and real-time tools as well as automatic code generators, form the basis of the 'model-based design' paradigm; a control design methodology that is centred on the use of reference system models and simulation at each design step. In the model-based design (MBD) approach, described in greater detail in the next section, initial modelling and requirements, early controller prototypes, production code generation, production controller testing and integration are all derived from reference models. The approach aims to accelerate the design cycle and reduce total design cost through the early detection of design flaws and other problems. In the automotive industry in particular the MBD method is a de facto standard. MBD is also quickly gaining acceptance and being adopted by design engineers in many different industries.

Real-time controller prototypes, whose development is based on automatic code generation, are being used in many engineering fields and applications, such as aircraft flight control design and validation, industrial motor drive design, complex robotic controller design and power grids. In several cases, the same controller code used for the prototyping phase is also used for the final hardware implementation.

The controller developed for these applications can be tested with scaled-down analogue benches, as was common 15–20 years ago. However, these applications can now benefit from the use of fully digital real-time simulators in a number of ways:

- Real-time simulation produces a set of requirements and specifications that can be used by all teams/subcontractors involved in a project.
- It enables testing of prototypes and actual controllers at or beyond their normal operating limits without the risks involved in testing real devices connected to the actual systems being controlled, especially when high power levels are present. It is easier and less risky to test fault responses on a simulated plant model.
- The simulation acceleration factor obtained by the use of compiled code (instead of the interpreted code used by most simulation tools), as well as by specialized parallel electrical solvers and hardware, enables the realization of rapid batch simulations. More tests can be done in less time.

Fast and real-time simulation can be used for statistical studies using the Monte Carlo simulation method for power grid applications such as finding the statistical distribution of amplitude of overvoltages and overcurrents caused by a series of random events. This analysis can be performed with prototypes or actual fast controller and protection equipment connected to the simulator (HIL). Data fitting/curve matching in the case of aircraft dynamic model parameter identification is another example of the use of fast simulation.

4.2 Model-Based Design and Real-Time Simulation

Model-based design is an engineering process that addresses problems associated with the design of complex systems, and it is based on the extensive use of models and simulation at each design phase. MBD is a methodology based on a workflow known as the 'V' diagram, as illustrated in Figure 4.1. It allows multiple engineers involved in a design and modelling project to use models to communicate knowledge of the system under development in an efficient and organized manner.

The left side of the V-cycle leads to the development of a production-type controller. The right side consists of steps to deploy this controller gradually until the final release.

MBD offers many advantages. By using models, a common design environment is available to every engineer involved in creating a system from beginning to end. Indeed, the use of a common set of tools facilitates communication and data exchange. Reusing older designs is also easier because the design environment can remain homogeneous through different projects. In addition to MBD, graphical

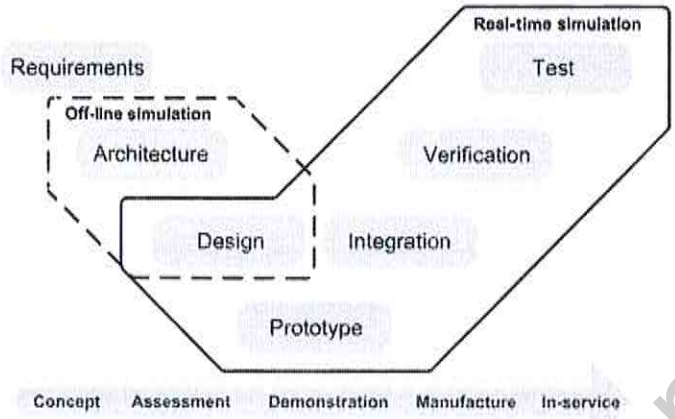

Figure 4.1 Model-based design workflow.

modelling tools, such as the SimPowerSystems toolbox for Simulink from The MathWorks, simplify the design task by reducing the complexity of models through the use of a hierarchical approach.

Most commercial simulation tools provide an automatic code generator (ACG) that facilitates the transition from controller model to controller implementation. The added value of real-time simulation in MBD emerges from the use of an ACG, which translates graphical simulation models into C code. By using an ACG with a real-time simulator, a hardware prototype of the controller – this is called rapid control prototyping (RCP) – can be implemented from a model with minimal effort. The prototype can then be used to accelerate integration and verification testing, something that cannot be done using off-line simulation. The same holds true for HIL testing. By using an HIL test bench, test engineers become part of the design workflow early in the process, sometimes before an actual plant becomes available. For example, by using the HIL methodology, automotive test engineers can start testing of a car or power system controller before a physical test bench is available. Combining RCP and HIL, while using the MBD approach, has many advantages:

- Design issues can be discovered earlier in the process, enabling required trade-offs to be determined and applied, thereby reducing development costs.
- The development cycle duration is reduced, due to parallelization in the workflow.
- Testing costs can be reduced in the mid to long term since HIL test setups often cost less than physical setups, and the real-time simulator can typically be used for multiple applications and projects.
- Test results are more repeatable because real-time simulator dynamics do not change through time the way physical systems do.
- It can replace risky or expensive tests that use physical test benches.

4.3 General Considerations about Real-Time Simulation

4.3.1 The Constraint of Real-Time

The most important aspect of real-time simulation technologies is that they need to make their computations fast enough to keep up with real-world time. Moreover, this constraint must be respected at all times and, specifically in terms of solvers, at all time-steps. This means that the solver used to compute the equations of the phenomenon under study must be optimized with this in mind.

Typically, the differential-algebraic equations (DAE) of the simulated system will be discretized and computed along a sequence of equal time-spaced points. Then, all these points must be computed and completed within the specified time-step. If a time-step is not completed in time, we get what is called an 'overrun', meaning that the simulated process is late compared to real time. Such overruns create distortion of the waveforms injected into the controller under test, which is not acceptable.

4.3.2 *Stiffness Issues*

One other problem that occurs in simulating electric circuits is that their equations often exhibit stiffness. Stiffness occurs when one simulates a DAE with a spread in the eigenvalues or natural frequencies – let's say high, medium and low – but with a special interest in the medium or 'mid-range' bandwidth phenomena. Non-stiff solvers (e.g. explicit Runge–Kutta) will be unstable because of the higher-frequency components of the DAE. In other words, to simulate these systems with a non-stiff solver, you will be forced to adapt the time-step to the higher frequency component, and the simulation will become extremely long for the frequency of interest. But a certain category of solvers called 'stiff solvers' are able to 'cut through' the high-frequency components and be less influenced by frequency components that are higher than the sampling frequency. Mathematically speaking, the solver stiffness property refers to A-stability or L-stability.

For power systems switching transient studies, people are typically interested in components below 2 kHz. But the equations may have eigenvalues in the MHz range that cannot easily be eliminated. For example, modelling switches with small and high resistance values to simulate the on and off states have a tendency to create stiff systems. This is why most power system simulator solvers are based on the A-stable order-2 trapezoidal rule of integration, as in RTDS [1] or Hypersim [2]. However, in most cases users must still use snubber resistors and capacitors across the switches to improve numerical stability.

The order-5 L-stable discretization rules of ARTEMiS [3] are, however, more stable for many situations. Because of its higher order, ARTEMiS order-5 is usually more accurate than an order-2 solver for the same time-step value. Such features enable the use of larger time-step values, which facilitate real-time simulation of difficult cases. Furthermore, while the trapezoidal integration rule can be applied with the classic nodal technique used in most simulation tools, higher-order solvers like ARTEMiS order-5 can only be used with a state-space formulation as in the state-space-nodal (SSN) solver algorithm of ARTEMiS.

4.3.3 *Simulator Bandwidth Considerations*

Another important aspect of a real-time simulator is its bandwidth; the definition of the simulation (the selected time-step) must be compatible with the speed or bandwidth of the phenomena that we want to simulate. For example, simulating electric circuits requires a much smaller time-step than mechanical ones because the former is much faster by nature. This is one reason why electrical systems are difficult to simulate: their bandwidth is high, imposing smaller time-steps and thus more powerful computers are needed to simulate them.

Because the simulation is a sampled process, the appropriate bandwidth of a simulator refers to the Shannon theorem, which requires a minimum of two samples within the period of the largest frequency of interest. But in practice, the simulation sampling frequency (inverse of the time-step) will be selected between 5 and 10 times the frequency of the phenomenon that the user wants to study.

4.3.4 *Simulation Bandwidth vs. Applications*

Figure 4.2 outlines typical time-step and computing power requirements for a variety of applications. The left side of the chart illustrates mechanical systems with slow dynamics that generally require a simulation time-step between 1 and 10 ms, using the rule of thumb that the simulation step should be

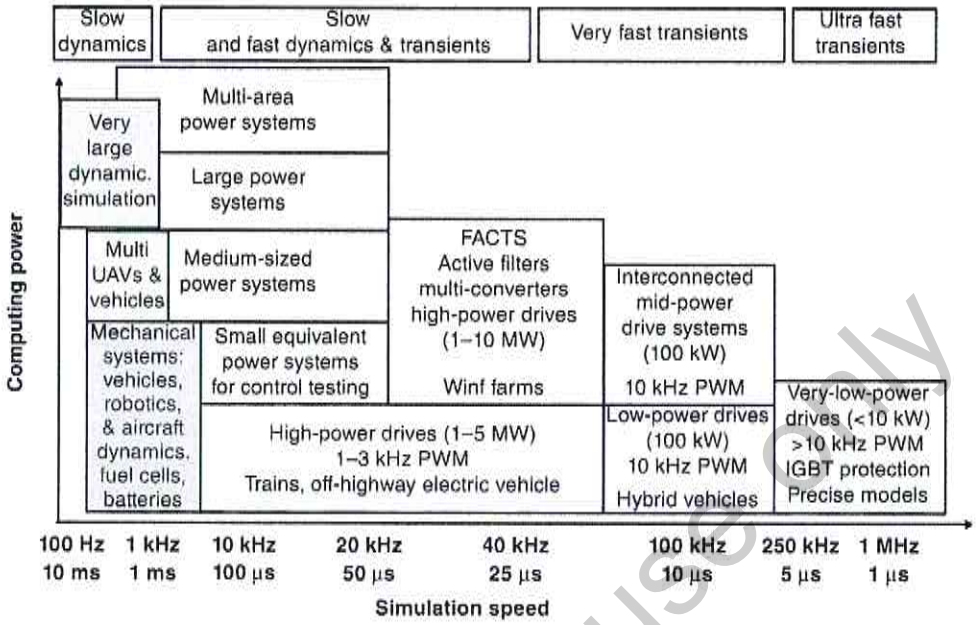

Figure 4.2 Simulation bandwidth by application.

smaller than 5–10% of the smallest time constant of the system. A smaller time-step may be required to maintain numerical stability in stiff systems. When friction phenomena are present, simulation time-steps as low as 100–500 µs may be required.

It is a common practice with EMT simulators to use a simulation time-step of 30–50 µs to provide acceptable results for transients up to 2 kHz. Because greater precision can be achieved with smaller time-steps, simulation of EMT phenomena with frequency content up to 10 kHz typically requires a simulation time-step of approximately 10 µs.

Accurate simulation of fast-switching power electronic devices requires the use of very small time-steps to solve system equations. Off-line simulation is widely used, but is time-consuming if no precision compromise is made on models (i.e. the use of average models). Power electronic converters with a higher PWM carrier frequency in the range of 10 kHz, such as those used in low-power converters, require time-steps of less than 250 nanoseconds without interpolation, or 10 µs with an interpolation technique. AC circuits with higher resonance frequency and very short lines, as expected in low-voltage distribution circuits and electric rail power feeding systems, may require time-steps below 20 µs. Tests that use practical system configurations and parameters are necessary to determine minimum time-step size and computing power required to achieve the desired time-step.

4.3.5 *Achieving Very Low Latency for HIL Application*

One of the main applications of real-time simulators is to develop and test protection and control systems by interconnecting the equipment in closed-loop with the simulator. This is called Hardware-In-the-Loop (HIL) or Controller-in-the-Loop (CIL). In such application, the total system must react like the real system, implying that the simulator delays are small compared to the delays and time constant of the actual controller, which is usually the case for typical thyristor-based power electronic controllers when the simulator time-step values are 25 to 50 µs.

However, the use of fast voltage source converters with PWM frequency larger than 10 kHz as well as modular multilevel converters (MMC) with a very large number of levels may require time-step values below 1 to 2 μs to achieve a total latency below 2–4 μs. The latency is defined as the time elapsed between the IGBT firing pulses sent by the controller under tests and the reception, by the controller, of voltage and current signals sent back by the simulator. Reaching such a low latency requires the use of field programmable gate array (FPGA) chips that are available in most modern simulators.

4.3.6 *Effective Parallel Processing for Fast EMT Simulation*

Conventional off-line simulation tools such as EMTP, EMTP-RV, ATP and PSCAD typically use only one processor to compute the total system, which means that the time to compute one time-step can exceed the wall-clock time when the size of the simulated system increases. However, real-time constraints also require the use of highly optimized solvers, taking advantage of parallel processing to reach the specified time-step, even if the complexity and size of the simulated system increases.

An important improvement in simulation speed was achieved in the 1990s by using the transmission lines modelled with distributed parameters to make the admittance matrix block-diagonal, creating subsystems that can be solved independently of each other. This property allows the network to be divided into subsystems with smaller admittance matrices which can be solved in much shorter time. The resulting simulation model structure is illustrated in Figure 4.3 where the line equations serve as links between the decoupled set of equations.

However, the effective implementation of this technique implies that the processing time for data exchange between processors is much smaller than the time used to simulate each subsystem. Fifteen years ago, only high-end commercial or very expensive custom-made supercomputers were able to implement such parallel simulation effectively with time-steps as low as 50 μs. The main impediment was the difficulty in implementing a low-latency interprocessor communication capable of transferring data with a delay below 50 μs without overloading each processor. In the early 1990s, organizations such IREQ (Hydro-Quebec's R&D centre), EDF (ARENE simulator [4]) and Manitoba HVDC R&D centres (and RTDS Technologies Inc.) developed their own custom-made supercomputers to implement real-time digital parallel simulators. Such supercomputers were expensive to develop and maintain.

By contrast, ubiquitous PC and laptop computers are equipped with processors that have up to eight cores. Standard PC servers can be equipped with up to 32 processor cores, tightly interconnected by a fast shared memory. They are able to run several threads in parallel with fast interprocessor communication, provided that the software uses these features.

Traditional EMT off-line simulation software packages, such as EMTP and PSCAD, are being modified by their respective development teams to take advantage of parallel processing. The EMT simulation

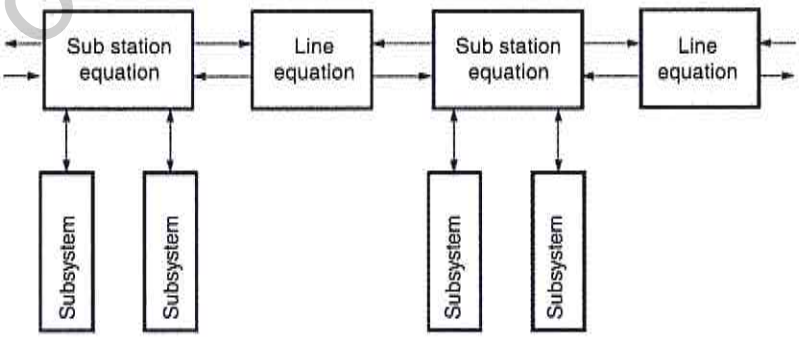

Figure 4.3 Program execution structure for EMT parallel simulation.

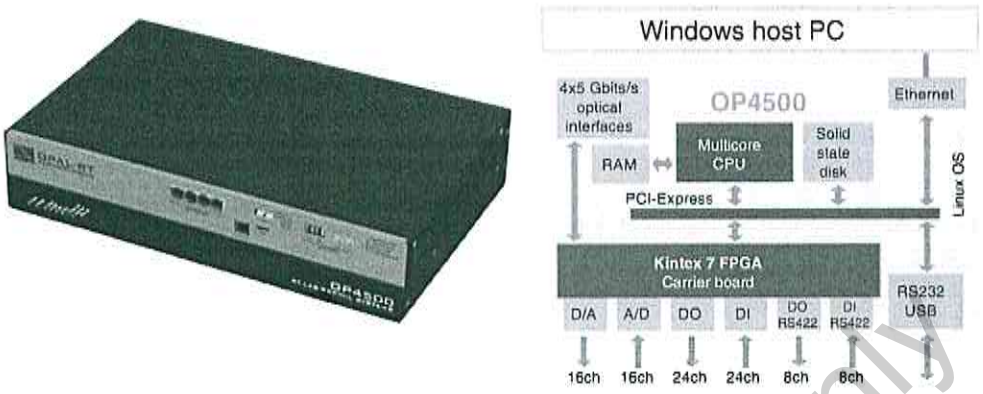

Figure 4.4 A multiprocessor desktop computer for fast simulation of complex systems.

software packages, Hypersim, eMEGAsim and RTDS, have been specifically developed, from the beginning, to take advantage of parallel and distributed processing, with great improvements in simulation speed. To perform this, Hypersim and RTDS partition the network at the connection points to transmission lines, creating subsystems, as illustrated in Figure 4.3.

Hypersim and eMEGAsim generate C code for each subsystem and compile them individually. Linkage is performed, depending on the commercial hardware used for the simulation, so that different executable modules are generated to be assigned to different processor cores. In this way, the simulation speed is increased according to the number of assigned processor cores.

Over the past 20 years, RTDS Inc. has developed ever-more-powerful custom computer boards optimized for real-time simulation of power grids. At the same time, the same Hypersim EMT parallel code has been ported from a custom supercomputer to the SGI supercomputer and now to standard multicore PCs.

Figure 4.4 illustrates an example of a small but powerful four-core simulator that can simulate a power grid with about 220 electrical nodes (single-phase) at 50 μs, using a standard low-cost Intel i7 four-core processor such as can be found in typical PCs. This small desktop simulator is also equipped with a powerful Kintex-7 FPGA to simulate fast power-electronic converter systems and controllers with time-step below 1 μs [5]. This setup can run a small system model in real time, for testing purposes, with HIL.

Although Hypersim is adequate for regular PCs, an interesting alternative for large, complex simulations is to use server-grade multiprocessor computers, which are now available at costs comparable to normal workstations. In this case, a 32-core Intel-based server can be used to simulate grids with more than 2000 electrical nodes with hundreds of generators, transformers and loads and several HVDC transmission systems with a time-step of 50 μs. This capability is very useful for integrating tests of power grids equipped with several HVDC, FACTS and distributed generation.

For very large grids, Hypersim can also take advantage of large SGI supercomputers with hundreds of processors cores. The ability to run the same simulation software on laptops or on commercial servers, supercomputers and on the cloud is certainly an advantage for large utilities, R&D centres and universities.

It must also be noted that using parallel processing for EMT simulation is essential not only for real-time HIL simulation but also to accelerate simulation studies performed in off-line mode. A faster simulation speed improves user interaction and enables the analysis of more contingencies in less time. For such applications, Hypersim can take advantage of commercial or in-house cloud computing.

Of course, using hundreds of processor cores to simulate very large grids requires the use of efficient and automatic processor allocation software to facilitate the use of such powerful parallel computers. Hypersim provides this essential feature with the ability to calculate all initial conditions.

4.3.7 FPGA-Based Multirate Simulators

The simulation of complex voltage source converters used in transmission and distribution systems such a modular multilevel converter (MMC) – as well as fast converters found in distributed generation systems such as PVs, wind farms and microgrids – often requires simulation time-steps below 1 μs. Such a small time-step cannot be achieved by today's standard computers. Specialized processor technologies such as FPGA processors must be used for simulation time-steps down to 200 ns, and to interface simulation results with external equipment using fast I/O converters. One such simulator architecture is that of the OPAL-RT OP4500 simulator in Figure 4.4. In this case the slower subsystems are simulated with a time-step of 10–50 μs on standard Intel i7 processors, while the fast power-electronic systems are simulated with a time-step below 1 μs. A general nodal solver, called eHS (electrical **H**ardware **S**olver) has been implemented directly on FPGA chips to facilitate the simulation of complex power electronic systems [5, 6]. Users can simply draw the circuit diagram, and the FPGA will automatically simulate it without any need to learn complex FPGA programming.

A powerful multi-FPGA simulator has also been delivered for the real-time simulation of MMC HVDC power grids using a total of 7500 MMC cell models and five converter terminals [7]. Such FPGA simulators are interfaced with eMEGAsim and Hypersim multicore simulators to allow the simulation of very large power grids, integrating fast power-electronic systems for transmission, distribution and microgrid systems.

4.3.8 Advanced Parallel Solvers without Artificial Delays or Stublines: Application to Active Distribution Networks

The parallel solver implementation described above is based on the availability of transmission lines or cables to separate the system into small independent subsystems, which can be simulated within the specified time-step, using only one processor. If the computational time for one of these subsystems becomes too large because it has too many nodes, then the common practice is to add artificial delays to enable parallel processing to reduce the computational time.

An artificial delay is usually implemented with a stub line, which is a line with its length adjusted to obtain a propagation time of one time-step. Large capacitors and inductors can also be used to split a large system into several smaller subsystems to take advantage of parallel processing. However, adding artificial delays also adds parasitic series inductances and parasitic shunt capacitances. Users must then compare results obtained without the addition of artificial delays using off-line simulation with results obtained using the artificial delays to reach the specified time-step. These parasitic L and C can be tolerated in many applications, depending on the values of these elements with respect to the other circuit impedances. This technique is used successfully by users of RTDS, eMEGAsim and Hypersim for many cases.

However, in some cases, such as large distribution circuits simulated with several pi circuits, the addition of artificial delays may be problematic because parasitic L and C will be large compared to actual component values and therefore *the standard decoupling methods cannot be used in distribution systems*. Simulating large converter systems interconnected with a large number of AC and DC switchable filters may also be problematic to reach time-step values below 25 to 50 μs even with the fastest available processors. Steady-state and transient values may be affected by the addition of artificial delays. Consequently, the use of circuit solvers that can simulate large circuits in parallel without adding any parasitic L and C can be very useful to increase simulation speed and accuracy. This is the key feature of the state-space-nodal algorithm described in detail in [3, 8].

SSN is a nodal admittance-based solver that allows us to minimize the number of nodes and thus the size of the nodal admittance matrix. This, in turn, can help increase the speed of simulation because the LU solution of the admittance equation requires a number of operations that is proportional to N^3, the cube of the size N of the nodal matrix. It does so by letting users choose the node locations and the corresponding groups of elements.

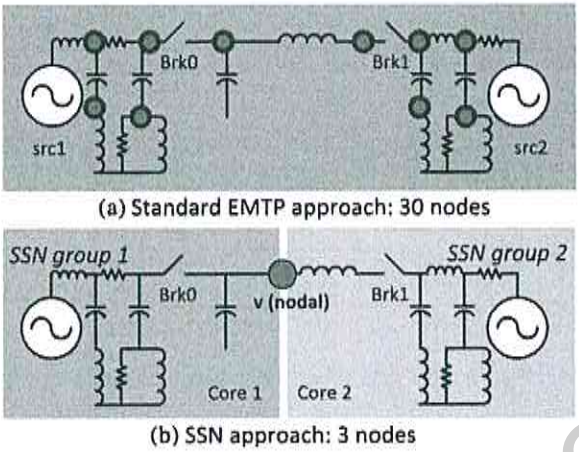

Figure 4.5 Comparison of node number for standard nodal admittance method and SSN.

As a simple example, we show the single-line circuit of Figure 4.5. In this circuit, the classic method to build the nodal admittance matrix, using standard RLC branches, would result in 30 nodes. By comparison, the SSN method can result in only three nodes. The reduction of the size of the nodal admittance matrix comes with an increased complexity in the resulting SSN group equations (a group is a multiterminal generalization of a branch).

When the SSN group equations are large enough, they can also be computed in parallel on different processor cores, without delays, thus accelerating the simulation even more. The SSN algorithm is more complex to compute than the simple nodal technique but actual results show a performance gain when using up to four processors, which can be very useful for the simulation of difficult cases where too many R, L and C components are tightly interconnected without lines or cables. This is depicted in Figure 4.7 (in the next section about the iterative MOV feature) where we can observe that the SSN group history source calculation and update stages are threaded on different cores without delays. Note that a similar delay-free parallelization could, in theory, be done with the standard EMTP approach. However, in practice, if the branch size is too small and their number is too high, the parallelization gains would be nullified by interprocessor communication overheads and other phenomena such as CPU cache trashing.

Using the SSN approach and the principles shown in Figure 4.5, it is possible to simulate in real time a radial distribution system with more than 700 nodes (with 980 LC states from short pi-lines) at a time-step under 65 µs, *without algorithmic delay or stub lines*, using four cores of a standard Xeon multicore server running RT-LAB. This performance is possible because the SSN reduces the network to a system with only six nodes and six multiterminal branches (i.e. SSN groups), and uses a threaded process to compute the SSN group equation in parallel, without any delay. A similar distribution system (Figure 4.6) with an on-load tap changer (OLTC) at the feeder point and more than 650 equivalent EMTP nodes can also be simulated in real time with the SSN solver. Using SSN and an 18 SSN nodes separation of the network (the six large dots in the figure), this model can be simulated in real-time at a time-step of 85 µs on a 3.3 GHz i7 Intel PC, without any delays or stub lines. In [9], this network was simulated without the SSN solver, and some delays were added to parallelize the simulation.

4.3.9 The Need for Iterations in Real-Time

When simulating, in real time, highly nonlinear devices such as surge arresters and metal oxide varistors (MOV), it is often necessary to use iterations in the standard nodal admittance solver. Hypersim [10] and

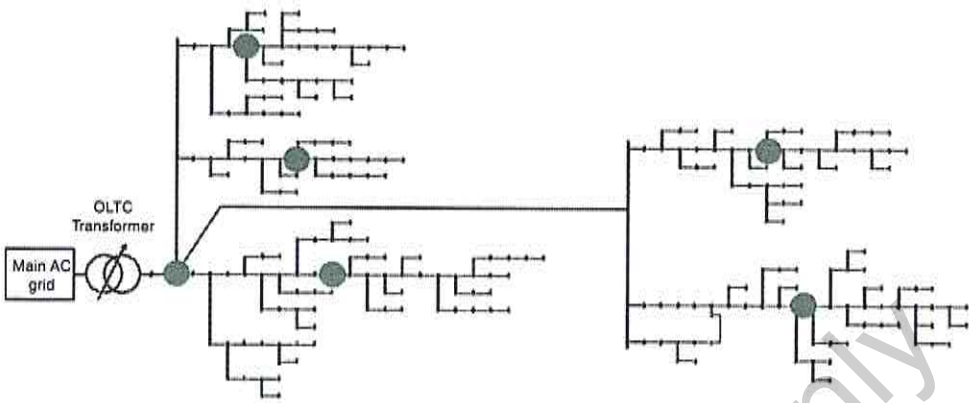

Figure 4.6 Large distribution network with more than 650 'EMTP-type' nodes.

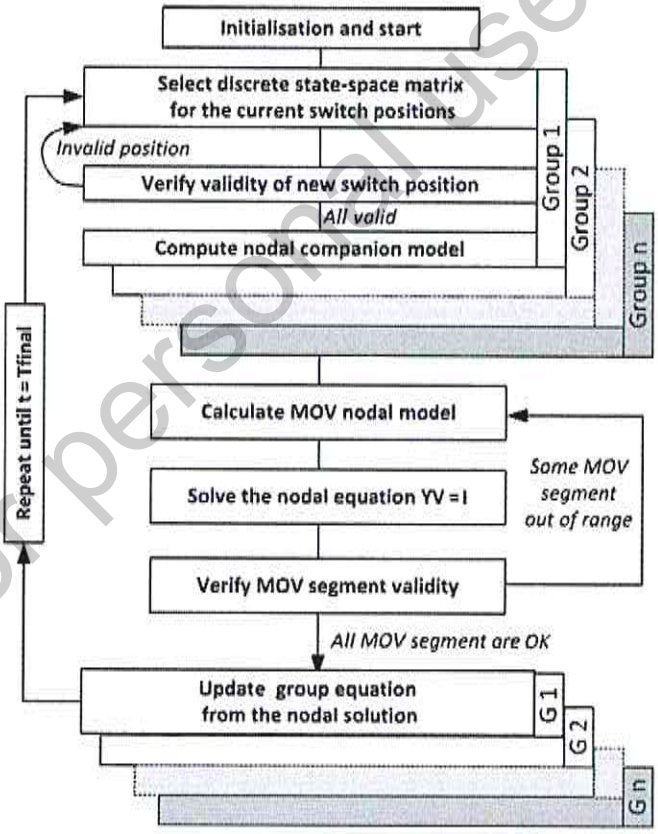

Figure 4.7 SSN algorithm with iterative MOV.

ARTEMiS [11] implement this type of iterative algorithm. The previous reference clearly shows realistic cases in which iterations are required to obtain accurate results.

Making iterations in a real-time simulator can be done efficiently because MOV only modifies the nodal equations of the solver, without history source computations, as shown in Figure 4.7.

Similar iterative principles are currently being applied to switches and also, for Hypersim only, saturable transformers.

4.4 Phasor-Mode Real-Time Simulation

Real-time simulation for EMT studies in power systems is well established. As discussed in previous sections, the fundamentals of most power-grid real-time simulations are based on the mathematics of EMT-type analysis. This includes the hardware/power/model-in-the-loop simulations. However, power system experts have long since realized that besides the real-time EMT domain simulation, they also need tools for off-line real-time phasor-domain simulation. These types of tools can be used to test the functionality of hardware such as global control and special protection devices in large-scale power systems. Furthermore, it can be utilized for training purposes in academic laboratories, or as an operator training tool in energy management centres. Some organizations are contemplating the use of faster-than-real-time phasor-type simulators for on-line prediction of system instability and implementing corrective actions to prevent system collapse or to minimize damage.

The ePHASORsim from OPAL-RT Technologies Inc. is a simulation environment for both transmission and distribution power systems and for balanced and unbalanced systems. The mathematics of ePHASORsim is based on a set of differential-algebraic equations [12]:

$$\dot{x}(t) = f(x, V) \tag{4.1}$$

$$YV = I(x, V) \tag{4.2}$$

$$x(t_0) = x_0, \tag{4.3}$$

where x is the vector of state variables, V and I are the vector of bus voltages and currents, Y is the nodal admittance matrix of the network, and x_0 is the initial values of state variables. The solver core is built as a Matlab/Simulink S-function, and its library of models is coded in C++, which can also be used in standalone mode. The built-in library includes major and most common components that are used for this type of simulation in power systems, such as synchronous machines and their controllers, different type of loads and transformers with on-load tap changers. An interface with Modelica FMI enables users to implement their own models. User models can also be implemented in C++ or with Simulink.

As of 2014, ePHASORsim can simulate systems with up to 10 000 nodes with a time-step of 10 ms with only one Intel processor core [13]. The systems can include about 3000 synchronous machines (six states), 5000 controllers and 20 000 other components. Faults, line switching and OLTC can also be simulated.

Similar research is being done, for example by PNNL in the USA [14] and PEGASE in Europe [15], to develop real-time and faster-than-real-time parallel transient stability simulators.

4.5 Modern Real-Time Simulator Requirements

Modern real-time simulators are typically built around multicore multiprocessor PCs with extensive I/O capabilities based on FPGA technology. Some simulators use custom-made computer boards.

A central processing unit (CPU) is a highly serialized arithmetic processing unit with a very flexible code flow architecture that enables it to implement very complex algorithms such as advanced ODE solvers. In modern PCs, it is commonly made with several computing cores and interfaced with other I/O components through a bus structure, such as PCIe in PCs.

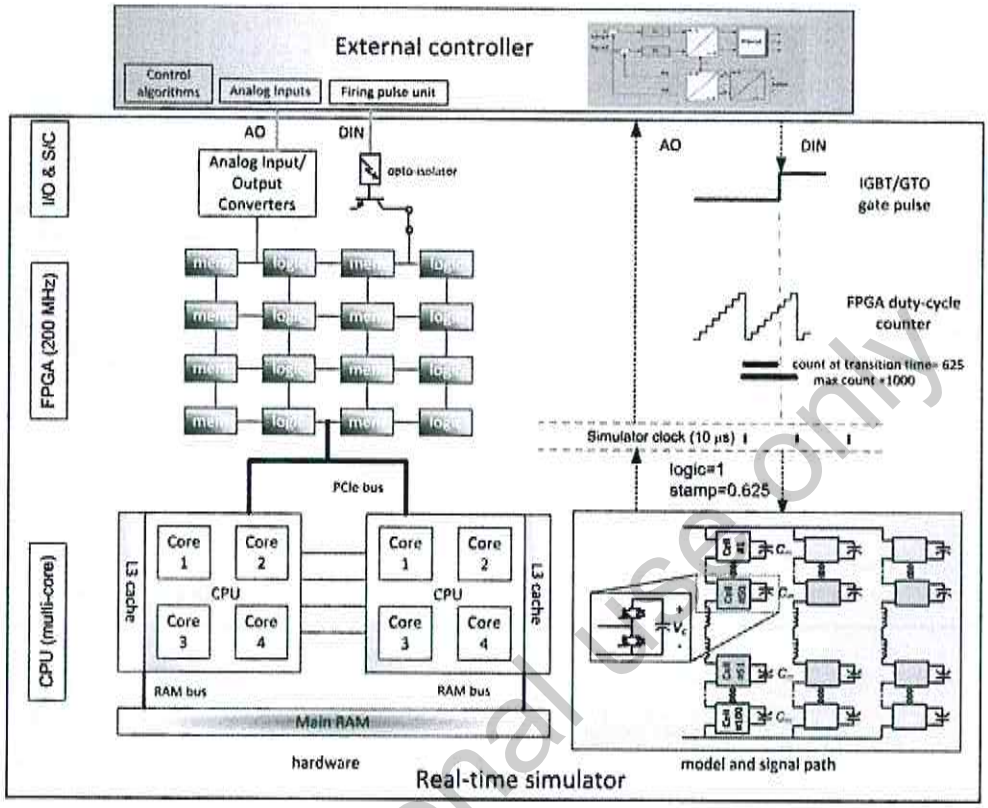

Figure 4.8 Simulator hardware and model paths.

A field-programmable gate array is a massively parallel structure of basic logic and memory elements that can be assembled into computing units from a user specification. Typical FPGA chips are mounted on electronic boards that, in addition to RAM memory and SSD hard disk, provide a direct and rapid interface with I/O points. It can also implement models and solvers of moderate complexity, especially for power electronics [5], as well as fast control systems and signal processing.

In Figure 4.8, we see the simulator hardware composed of CPUs and FPGA, along with the model path for MMC circuits. It shows that the I/O signals driving the IGBTs of the MMC cells are first captured by some timing on the FPGA card. This allows the precise timing of the IGBT gate pulse to be recorded. This timing, a time stamp, is then sent to the complete MMC model on the CPUs, where the time stamp is used to correctly compute the model voltages and currents.

In other cases, the MMC models can be simulated directly in the FPGA chips [7] using a very small time-step value, below 1 µs, to increase overall accuracy and to decrease the total latency.

4.5.1 Simulator I/O Requirements

I/O requirements are increasing for real-time simulation. Today's power electronics systems are getting more complex as new topologies, such as the MMC, gain acceptance. In order to perform HIL simulation of power electronic devices, real-time simulators must have various I/O types such as: (1) time-stamped

digital input, capable of sampling switching device gate signals with a resolution better than 100 nanoseconds (these types of inputs are sometimes referred as PWM inputs); (2) analogue output (1 μs sampling time), to emulate the device current, voltage and sensors as viewed by the controller under test; (3) digital input and output to for routing various signals between the controller under test and the real-time emulated power electronic devices.

The FPGA also provides the capacity to easily code and emulate various sensors, such as motor resolvers, quadrature encoders, RVDT and fault sensors.

Modern power electronic devices and controllers can also communicate through high-level communication links, such as Ethernet-based IEC-61850, DNP3 protocols for power system relays and substations, C37.118, the IEEE standard for using synchro-phasors in power systems, CAN protocol for automotive or the TCN protocol for trains. The simulator must be able to physically interface with such protocols with an appropriate driver. For this purpose, on PC-based real-time simulators, the simulator can be equipped with a PCI/PCIe interface board and/or a simple Ethernet board for IEC-61850. Alternately, the user can also program the FPGA board to interface to the desired protocol.

The simulator must also provide proper signal conditioning for all I/Os such as filtering and isolation. Additionally, the simulator should provide easy access to I/O interfaces between the simulator and the device under test. Typical real-time simulators provide such probing points on the front panel of the simulator, as does the OPAL-RT OP5600 simulator, Figure 4.9, or with a more complete patch panel. Figure 4.9 shows the probing connection points on the front of the simulator. The controller under test is interfaced with the simulator through the I/O interface on the back of the simulator.

The ability to provide a very large number of I/O points is also very important in certain applications. The MMC is a good example of a device with a very large number of I/Os.

As an example, a dual three-phase 100 cells/arms MMC simulator was recently commissioned by OPAL-RT with a total of 2788 I/O points with three digital signals per cells (1800), one analogue output per capacitor (600) and various other voltage and current sensors and well as breaker signals. The time to manage all these I/O channels and to compute the models was 25 μs using parallel processors and several FPGA chips.

A large MMC system, with up to five converters with 2400 cells each (120 000 cells in total) to simulate a DC grid, was also delivered to the China Institute, using 10 FPGA systems – OP7020 Virtex-7 FPGA racks that basically extend the number of I/Os for the OP5600 simulator – to simulate all the MMCs in less than 500 ns and to interface thousands of signals to customer controllers using optical fibres.

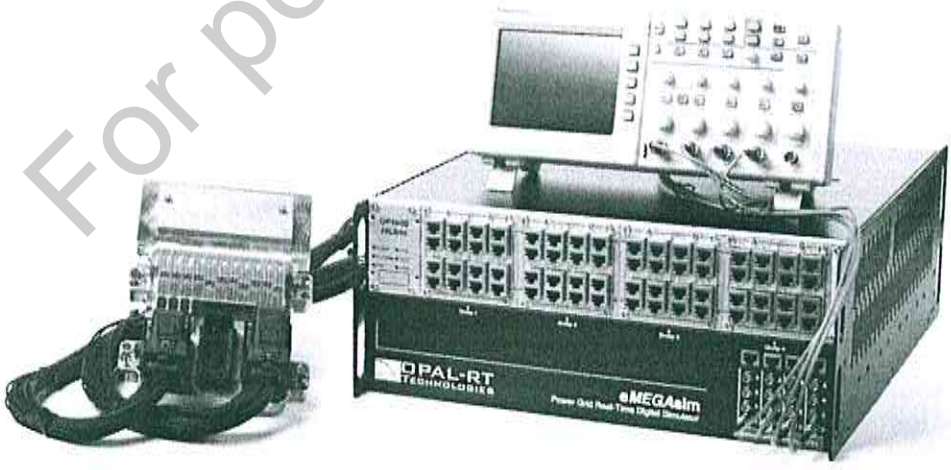

Figure 4.9 OP5600 real-time simulator (right) with a controller under test (left).

4.6 Rapid Control Prototyping and Hardware-in-the-Loop Testing

Various applications of real-time simulation are detailed in this section. These applications can be broken into five important categories:

- rapid control prototyping (RCP)
- hardware-in-the-loop testing (HIL) or controller-in-the-loop (CIL)
- software-in-the-loop (SIL)
- power-hardware-in-the-loop (PHIL)
- rapid simulation (RS)

Consider a controlled process that is composed of a plant with a controller acting upon it. In RCP applications, an engineer will use a real-time simulator to quickly implement a controller and connect it to the real plant or to a simulated plant. HIL (or CIL) acts in an opposite manner: its main purpose is to test actual controllers connected to a simulated plant.

Software-in-the-loop can be done when controller object code can be embedded in the simulator to analyse the global system performance and to perform tests prior to use of the actual controller hardware (HIL or CIL).

Power-hardware-in-the-loop (PHIL) consists of using real power components in the loop with the simulator. An example is a high-power motor emulator used to test actual inverter systems. The motor emulator uses a real-time simulator controlling a power amplifier to output the simulated motor current with the same amplitude as the actual motors. PHIL is also used for battery emulator and microgrid laboratories [16–19].

Rapid simulation take advantage of parallel processing to accelerate simulation in massive batch run tests, such as aircraft parameter identification using aircraft flight data or in Monte Carlo simulation used in power system analysis. RS is very useful for simulating large MMC HVDC grids using detailed converter models to reduce simulation time by several orders of magnitude depending on system complexity and size.

4.7 Power Grid Real-Time Simulation Applications

4.7.1 Statistical Protection System Study

Protection and insulation coordination techniques make use of statistical (Monte Carlo) studies to deal with inherent random events, such as the electrical angle at which a breaker closes, or the point-on-wave at which a fault is applied. For protection coordination studies, multiple fault scenarios are required to determine appropriate protective relay settings and correct equipment sizing. By testing multiple fault occurrences, the measured quantities are identified, recorded and stored in a database for later retrieval, analysis and study.

Protection relay development and testing is one of the most traditional uses of real-time simulators. Actual protection relay equipment is interfaced to the simulator using voltage and current amplifiers in such a way that the relays receive the same signals as they would if connected to actual power systems. Protection relay breaker tripping signals are connected to the simulated breaker through logical inputs. Faults can then be applied to the simulated network to test relay performance (i.e. its operating speed and accuracy) as well as to test the relay security (i.e. to determine the number of erroneous operations).

Figure 4.10 presents a typical setup for testing modern relays interfaced with the IEC 61850 Ethernet protocol replacing hard-wired connections.

Data measured by the PT and CT is sent by the simulator using Ethernet protocol IEC 61850-9-2 sampled values to the MiCOM P444 relay. The relay backup overcurrent function detects the fault and the relay sends an IEC 61850-8-1 GOOSE message back to the simulator in order to control the circuit breakers at each end of the transmission line.

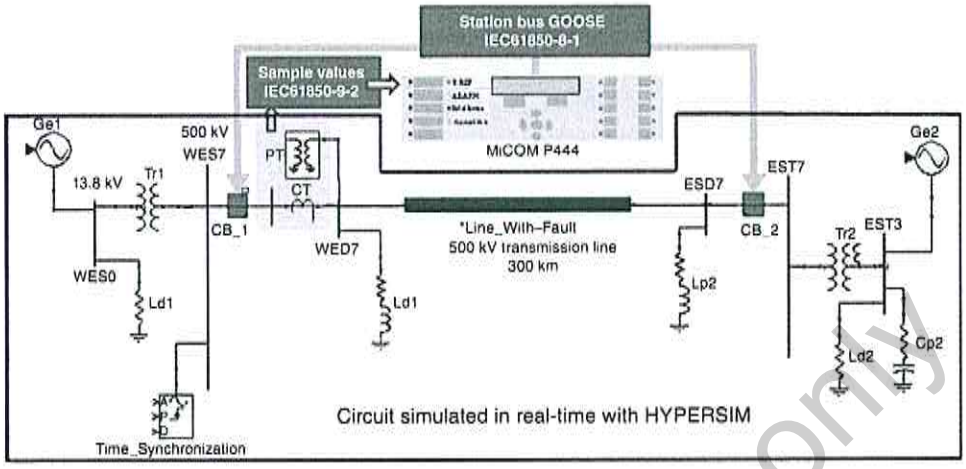

Figure 4.10 Typical protection relay testing setup with IEC61850 and HYPERSIM.

Many automated tests must be executed to evaluate the relay performance as a function of the fault impedance, duration and incidence angle.

Figure 4.11 presents such statistical results, executed with TestView, the Hypersim test automation software, by modifying all these parameters so that a fault is provoked at 50 km from the west bus on the 500 kV transmission line. In this test, the incidence angle varied between 0° and 359° and 332 tests were run in order to characterize the reaction time of the relay. The average value was 21.25 ms and the standard deviation was 1.08 ms. Several tests must also be performed to test the capability of the relay to not operate for transient voltages and current values outside its specified operating range. Of course, more complex tests can be performed using several relays installed on several lines and transformers, to evaluate the risk of cascaded relay misoperations caused by the effect of previous relay operation. Such tests are often performed to test new relay design or new network topologies. The integration of new HVDC and FACTS technologies, as well as distributed generations and microgrids, may require more sophisticated tests.

In several cases, transients overvoltage and overcurrent waveforms can be recorded and played back later to test the relays in open loop mode. This testing method is very popular when it is not necessary to evaluate the effect of relay operation on the performance of other relays. Waveform recording can be performed by standard off-line simulation software, such as EMTP-RV and PSCAD. Real-time simulators are also used in faster-than-real-time mode, when the grid is very large, to reduce the time to record thousands of waveforms.

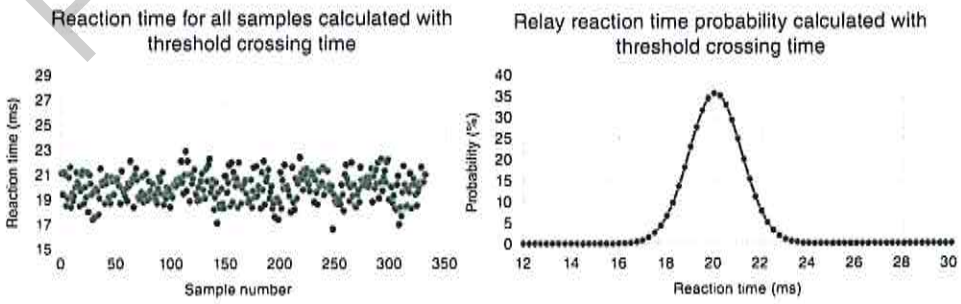

Figure 4.11 Protection relay testing results showing the probabilistic reaction time.

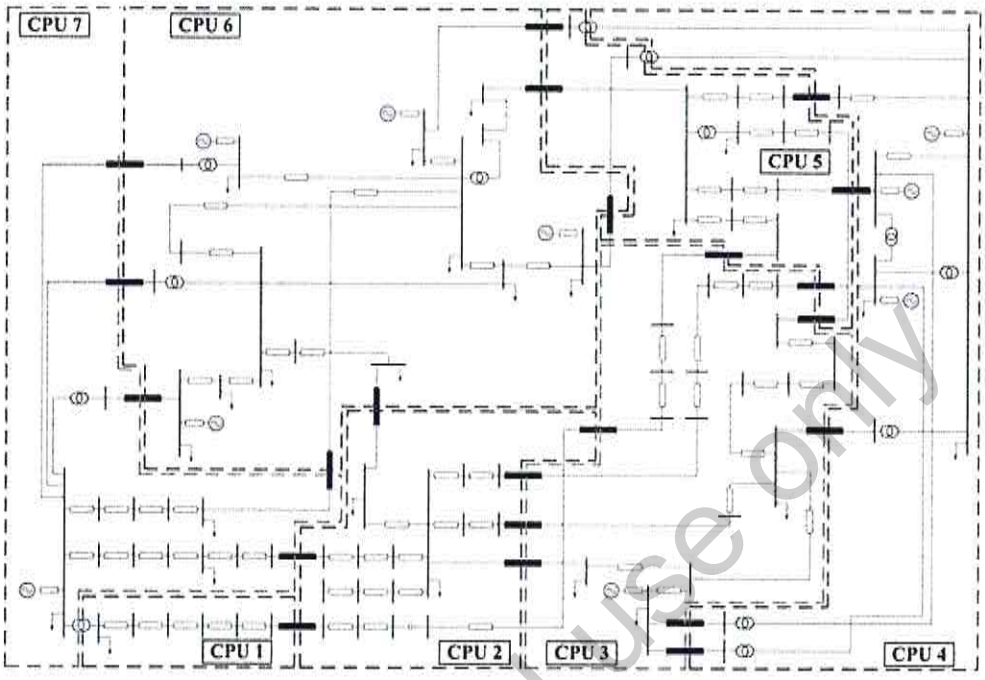

Figure 4.12 Schematic diagram of the network model – Courtesy of Entergy.

4.7.2 *Monte Carlo Tests for Power Grid Switching Surge System Studies*

The Monte Carlo technique has been used in the past for practical studies, such as the evaluation of the probable overvoltage at a substation [20], using traditional off-line simulation software such as ATP, PSCAD or EMTP-RV, as well as real-time simulators.

The large power system model depicted in Figure 4.12 illustrates a network with a very large number of busses and short lines. It was first built and tested in the EMTP-RV environment and then converted to the SimPowerSystems/Simulink environment as a distributed model, ready to use with the RT-LAB real-time simulator to perform statistical tests in less time but also to analyse the impact of SVC and future HVDC systems using actual control equipment.

The 60 Hz, 138/230 kV HVAC power system model is an 86-bus electrical network. Its large number of transmission lines supply power to a total of 23 loads, and nine ideal voltage sources with lumped equivalent impedance represent the generators. Full machine dynamics can easily be added, as these models are available in SimPowerSystems.

Figure 4.12 also describes the simulated power system along with CPU task separation for the eight-core target used in this test. Just a decade ago, simulation of such a model would not have been possible. A model of such complexity would have required a supercomputer composed of proprietary hardware. Today, this model can be simulated in real time at a time-step under 50 µs on commercial-off-the-shelf (COTS) PC computer hardware equipped with the RT-LAB real-time simulation platform. This particular 86-bus model can now be run on four 3 GHz Pentium cores under the Linux real-time OS.

The Monte Carlo test is designed using the Python script language, and uses the RT-LAB API to control simulation runs, data acquisition and post-treatment [21]. As can be observed in Figure 4.13(a), at the point where two different fault durations are applied (one cycle versus three cycles), the overvoltage measured during certain contingencies is dependent on fault timing. This comparison shows the necessity

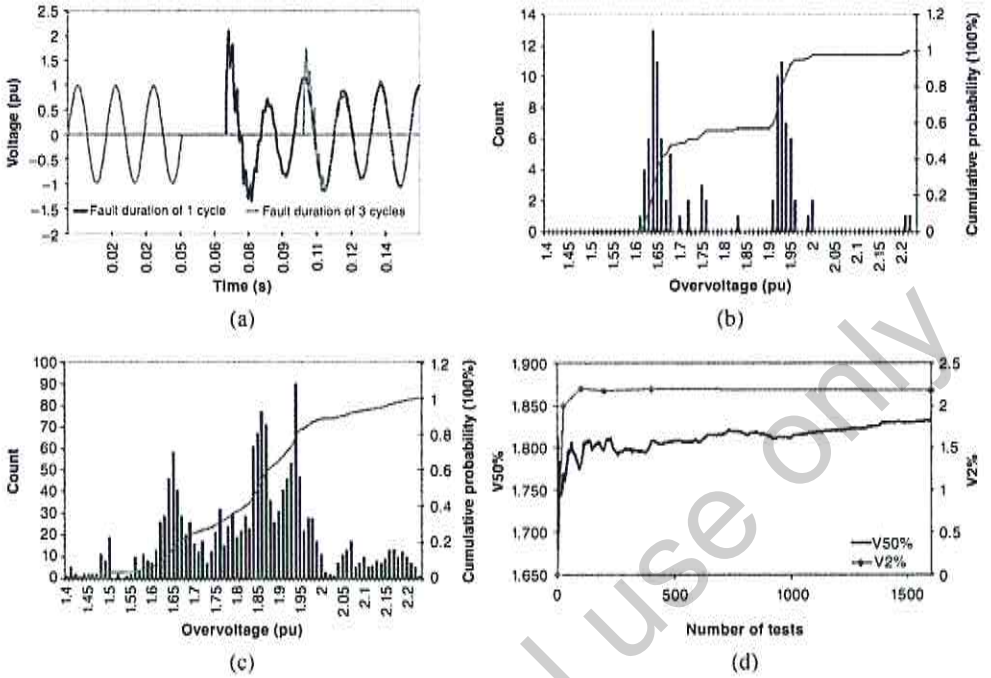

Figure 4.13 Monte-Carlo Study results. (a) Phase-a voltage for one-cycle and a three-cycle fault duration (Tstart = 50 ms). (b) and (c) Overvoltage statistic distribution for 100 and 1600 tests. (d) Mean value and standard deviation evolution with respect to number of samples.

of using statistical methods to calculate the maximum overvoltage to achieve an efficient design of the power system under study. It is also important to understand that a significant number of tests may be needed, depending on the study. The total number of tests needs to be large enough to obtain acceptable precision on the statistic distribution. The criteria used to determine the number of tests can be decided upon by calculating the evolution of the mean value (V50%) and/or, for instance, the evolution of the maximum overvoltage having a 2% probability of occurring (V2%). The evolution of these two quantities, according to the number of tests performed, is illustrated in Figure 4.13(d).

The above study can be performed in off-line mode using conventional software or with parallel software such as Hypersim or eMEGAsim to execute more tests in less time. However, in several cases, actual protection or controller hardware must be interconnected with the simulator to analyse the effect of fast control and protection actions on system overvoltage, overcurrent and arrester energy. Such real-time statistical tests also enable testing of control and protection system performance under realistic transient conditions and discovering unwanted random controller operations.

4.7.3 Modular Multilevel Converter in HVDC Applications

The inclusion of high voltage direct current (HVDC) and Flexible AC Transmission System (FACTS) devices in electric power grids is expanding rapidly. Also relatively recently, the use of voltage source converters (VSCs) based on insulated gate bipolar transistors (IGBTs) is becoming more attractive, mainly due to their cost and higher performance. The recent modular multilevel converter (MMC) topology, based on half-bridge modules connected in series [22], offers significant benefits compared to

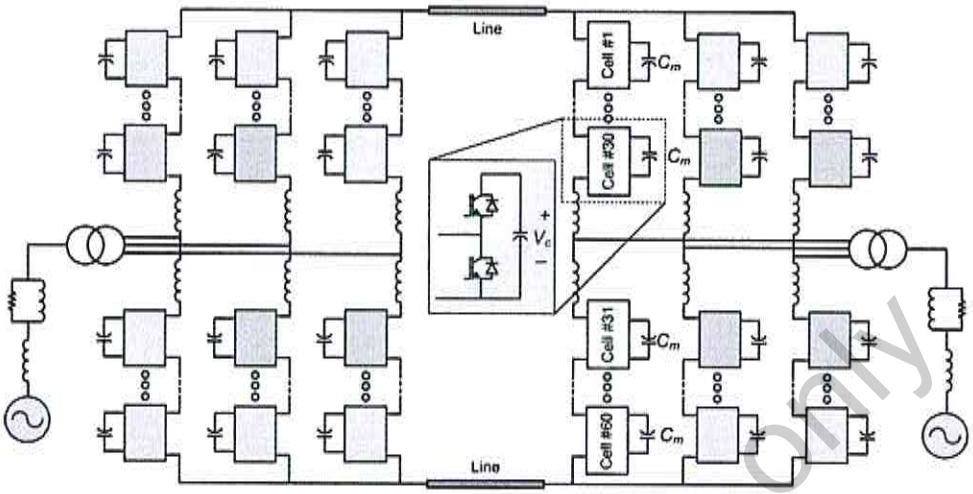

Figure 4.14 An MMC-based HVDC link.

previous VSC technologies, such as two-three level and neutral-point diode-clamped (NPC) topologies. By using a significant number of levels per phase in the MMC, the filter requirements can be eliminated. Moreover, switching frequency and transient peak voltages on IGBT devices are lower in the MMC, which reduces converter losses. An HVDC systems made of MMCs is shown in Figure 4.14. Scalability to higher voltages is easily achieved and reliability is improved by increasing the number of submodules (SMs) per phase.

The excessive number of power switches in the MMC, creates significant computational difficulties in EMT-type simulation tools. The numerous and nonlinear devices in the converter require an iterative process to solve the global matrix, which significantly increases the computational burden. Thus, in real-time simulations, modelling a highly accurate switching device is out of reach with current computational technology and some form of simplification is required to accomplish network integration and hardware-in-the-loop studies.

Real-time simulation of these large MMC systems enables users to develop and test actual controller equipment by interfacing them to the simulator in closed-loop. Such an HIL setup makes it possible to test the control system performance under several operating conditions before its installation in the real power grid. Real-time simulators with the ability to simulate MMC systems and HVDC grids are therefore essential and are used by MMC manufacturers, R&D centres and utilities.

In [23], full real-time digital simulation of a static MMC HVDC link interconnecting two AC networks is discussed. The converter has 60 cells per arm; each cell has two power switches with antiparallel diodes and one capacitor. The simulated model can be used to study the natural rectifying mode, which is very important in the energizing process of the converter, whether a ramping voltage or a charging resistance is used. Current injection with delay is the method used in this approach. This facilitates the use of HIL in realistic-sized MMC systems that can have more than a thousand cells.

In [24], the MMC is simulated using the SSN approach. The advantage of this approach is to avoid injection delays that can destabilize the simulation. The difficulty in using this approach is to effectively include the I/O signals with small latency when the cell number increases.

4.7.4 *High-End Super-Large Power Grid Simulations*

Because of the importance of grid reliability, utilities have very stringent testing requirements, and many of them have developed their own real-time simulator for these purposes. Hypersim, the Hydro-Quebec

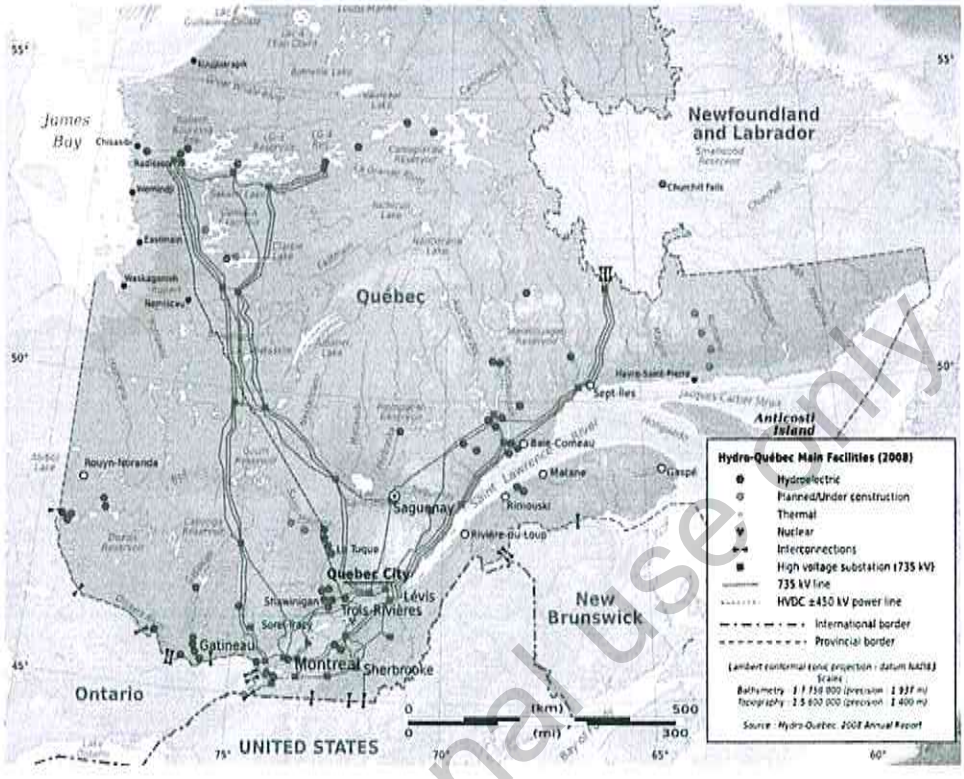

Figure 4.15 Quebec grid simulated in real-time by Hypersim.

Research Institute (IREQ) real-time simulator, is a good example of these high-end simulators, which have been developed and used over the past 20 years.

As an example, the complete power network of the Province of Quebec, depicted in Figure 4.15, including 25 DFIG-based wind power plants, was simulated in real time on the Hypersim real-time simulator [2]. The network contained the following elements: 643 three-phase buses, 34 hydroelectric generators (turbine, AVR, stabilizer), one steam turbine generator, 25 wind power plants with DFIG generators, seven static VAr compensators, six synchronous condensers, 167 three-phase lines, on multiterminal DC-link and 150 three-phase transformer with saturation modelling.

Hypersim used 72 processors in an SGI supercomputer to perform the real-time simulation of this network at a time-step of 50 μs. Larger systems are simulated by China State Grid using Hypersim [25].

One key aspect of this type of 'super-large' EMT simulations is that the Hypersim simulator comes with automatic task allocation. The Hypersim GUI is also designed to cope with super-large grid diagrams including single-line schematics as only one example. Continuously developed and improved internally at IREQ since the mid 1990s, Hypersim has been marketed by OPAL-RT Technologies since 2012.

4.8 Motor Drive and FPGA-Based Real-Time Simulation Applications

4.8.1 Industrial Motor Drive Design and Testing Using CPU Models

Mitsubishi Electric Co. has used real-time simulation technologies to design motor drives for machine tool applications. The challenge faced on this project was that the motor itself, a permanent magnet

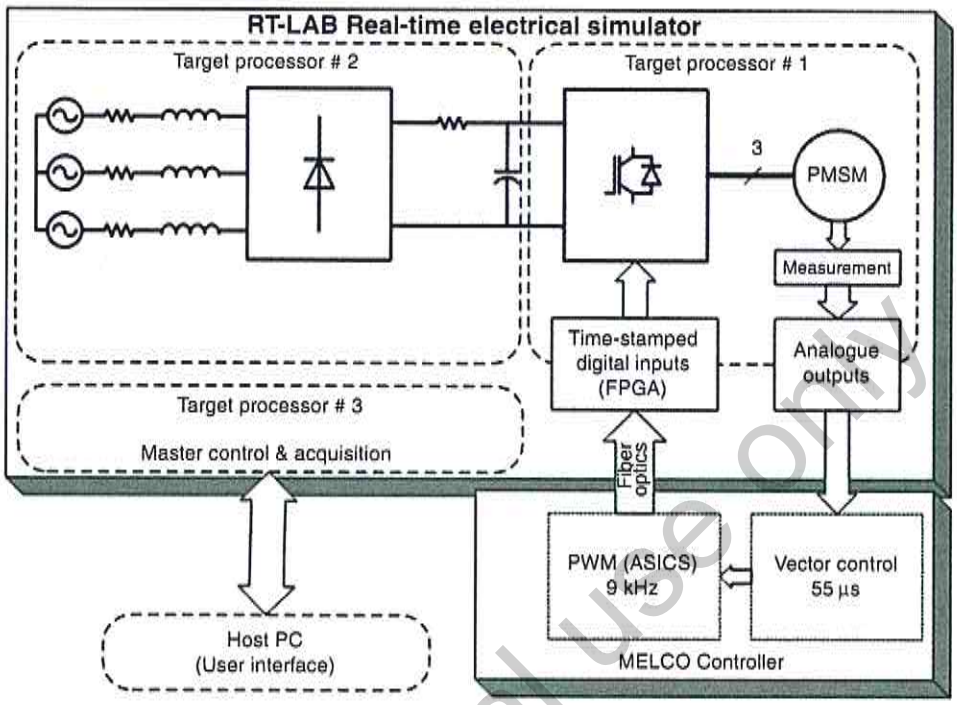

Figure 4.16 Real-Time Simulation of a PMSM drive.

synchronous motor and its related controllers were designed simultaneously. Therefore, a physical motor was not available for controller tests. The solution was to use a virtual motor, simulated in real time, during the controller testing phase [26].

The setup, depicted in Figure 4.16, consists of two main parts: the controller and the motor drive circuit. The controller includes a control module and PWM generator board. The vector control runs at 55 µs, and the PWM carrier frequency can be varied up to 9 kHz. The motor drive circuit was implemented in Simulink and simulated in real time on an RT-LAB electrical simulator. The motor drive was simulated more than 10 years ago by two Pentium 4 target processors, each operating at 2.8 GHz: one for the AC-DC part, and one for the DC-AC part, including the motor. A third Pentium processor was used for master control of the simulator, and for data acquisition sent to a remote host by a 100 Mb Ethernet link.

The simulator used a blockset called RTeDRIVE, designed by OPAL-RT, which uses interpolation techniques to solve under-sampling problems of the PWM waveform by the real-time simulator. As can be seen in Figure 4.17, HIL simulations closely match actual system results despite a 10 µs sample time and a nominal 9 kHz PWM frequency. Furthermore, one can observe that the current ripple amplitude decreases when PWM frequency is increased, just as in a real drive system.

Note that, today, such HIL simulation can be done with a time-step of less than 1 µs, using a complex finite element motor model implemented on a single FPGA chip, as discussed in the following sections.

4.8.2 *FPGA Modelling of SRM and PMSM Motor Drives*

In [27], an FPGA implementation of a switched reluctance motor drive (SRM drive) and an H-bridge Buck-Boost converter targeted for HIL testing of modern SRM controllers.

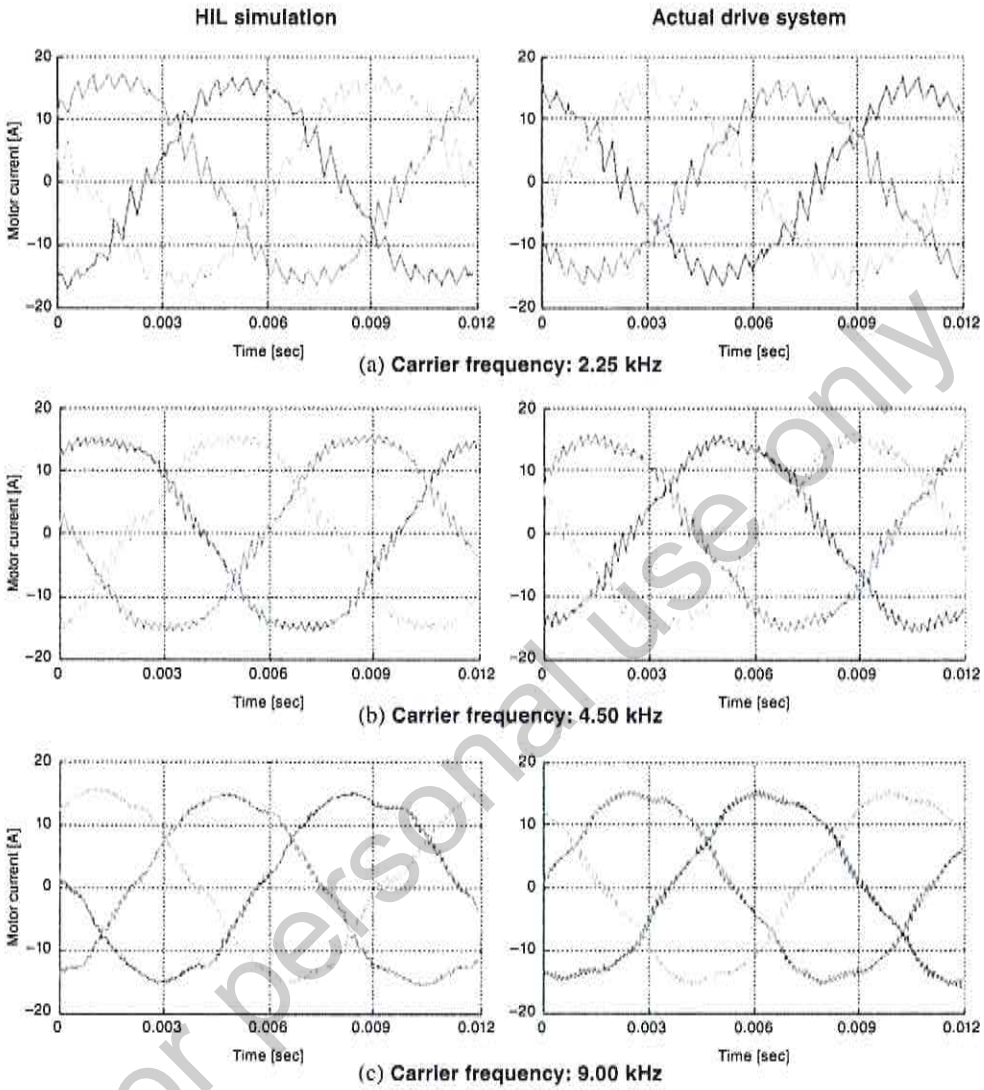

Figure 4.17 Effect of PWM carrier frequency.

These FPGA models, shown in Figure 4.18, allow the HIL simulation of SRM drive and boost converter with switching frequencies in the 50–100 kHz range because of the very high sampling rate of the FPGA. The models are also integrated into the RT-LAB real-time environment and directly linked with the simulator I/Os, providing ultra-low HIL gate-in-to-current-out latency, suitable for testing motor controllers with ultra-low latency requirements.

Similar FPGA models have also been designed for PMSM drive [28]. This type of model is used to test hybrid vehicle drivelines, similar to the Prius car, composed of two PMSMs, depicted in Figure 4.19. These PMSM models include finite element analysis (FEA) data and are sufficiently accurate to replace the real driveline, as tests made by the OEM show – see Figure 4.20.

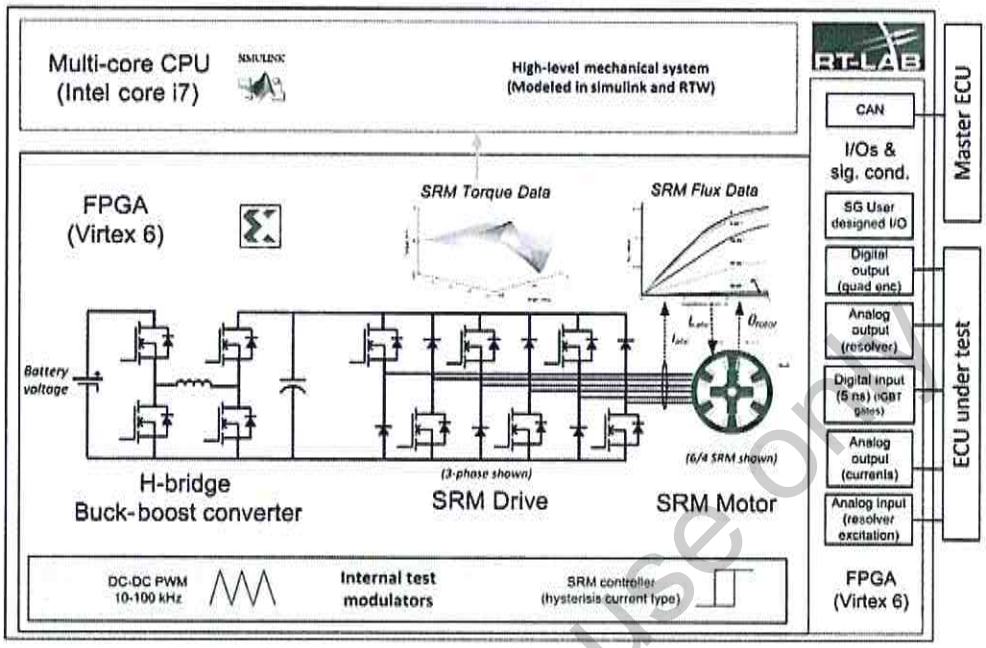

Figure 4.18 FPGA-based SRM Drive HIL simulator with DC-DC converter.

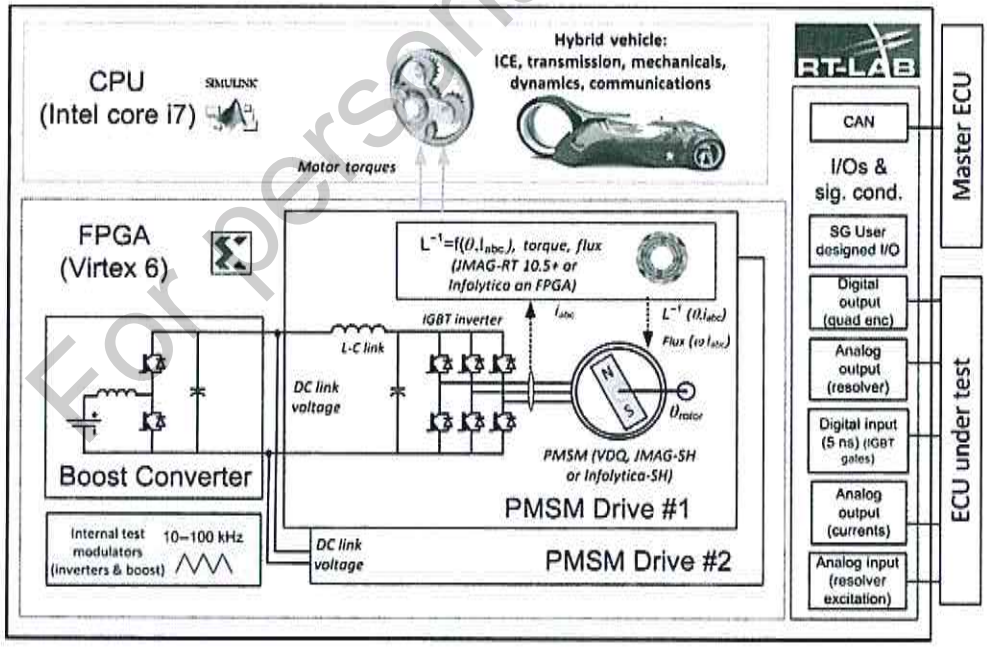

Figure 4.19 Dual-PMSM drive with boost converter of FPGA.

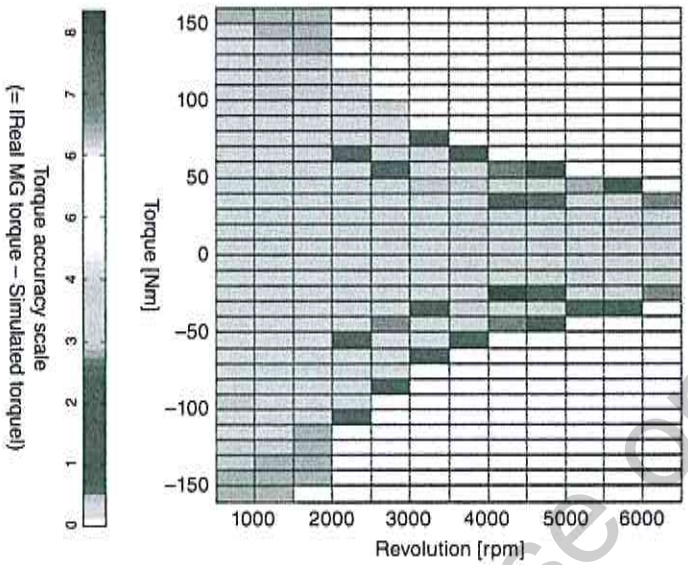

Figure 4.20 FEA model accuracy check with real driveline.

4.9 Educational System: RPC-Based Study of DFIM Wind Turbine

Working with fully virtual systems may not be advised at the undergraduate level. The fear of smoke certainly still has some educational value. In this case, it is possible to combine a real-time simulator, used as an RPC device, in conjunction with a medium-level power device. An example of such a product is produced by OPAL-RT and Lab-Volt, in which a student can learn about wind turbine controls and make various experiments with it. Two power levels are available: 200 W and 2 kW.

The system, shown in Figure 4.21, has been specially designed to be simple and robust for use in educational laboratories, but is sufficiently open to allow professors or students to expand the system to meet their unique requirements, and to develop new control strategies and test them on the platform.

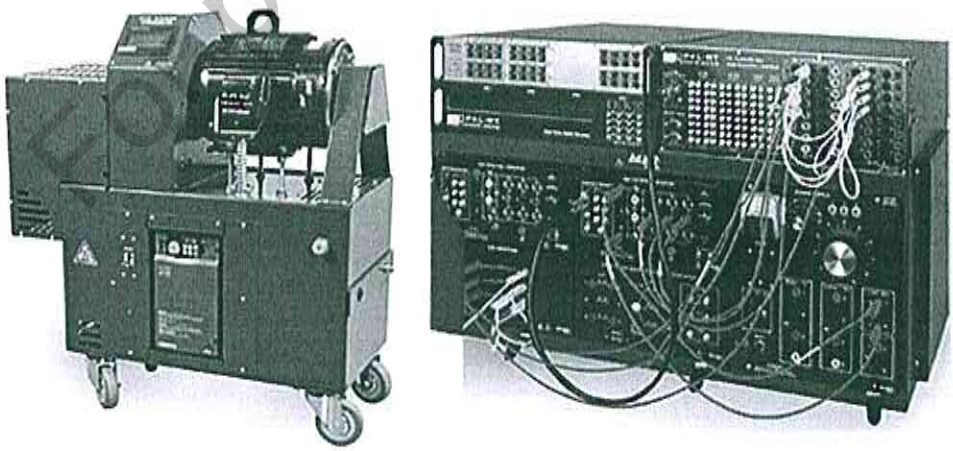

Figure 4.21 DFIM training kit OPAL-RT/Lab-volt.

4.10 Mechatronic Real-Time Simulation Applications

4.10.1 Aircraft Flight Training Simulator

Aircraft manufacturer Empresa Brasileira de Aeronáutica (Embraer) is using the RT-LAB real-time simulation software platform as the core component of a real-time simulator executing a high-fidelity aircraft model of a fighter plane [29]. In this application, the RT-LAB system runs in closed loop with the real onboard aircraft computer and the real aircraft cockpit. RT-LAB also provides control of force feedback (e.g. flight control joystick), visual display and sound generator.

The RT-LAB simulator is composed of more than 200 digital I/O points, more than 64 analogue I/Os (force feedback and pot reading), two MIL1553s and two Arinc 429 boards (for communication with onboard computer and aircraft components). TCP/IP communication handles sending data to the visual display and sound generator. Custom-made I/O developed by OPAL-RT reads the aircraft rotary variable differential transformer (RVDT) throttle position sensor (high speed analogue I/O). This type of I/O was designed using the Xilinx system generator blockset, supported by the RT-LAB system. The whole simulation is controlled by training software designed by Embraer. This software uses the RT-LAB API to control the entire simulation.

4.10.2 Aircraft Flight Parameter Identification

The development of the Embraer 170 Jet [30] has benefitted from an expanded modelling and simulation capability at Embraer. Development of a highly accurate aerodynamic model became an important part of the design development phase. This modelling was primarily conducted using Simulink. To obtain an accurate model, parameter identification needed to be done using actual flight data. This was performed using a software suite called RT-LAB/Dinamo 0. A real-time simulator is necessary for handling this task since multiple batch runs are required to fit the model parameters to actual flight data using the Nelder–Mead simplex and Levenberg–Marquardt optimization algorithms. The latter algorithm is used to fit model coefficients using manoeuvre time history – a particularly computation-intensive task. These fittings are made using captured flight data of actual aircraft manoeuvres such as take-off, jet Dutch roll and jet stall.

The final aircraft model includes all aspects of the plane, including motion and aerodynamic equations, engine, hydraulic systems and sensors. The different teams involved in the development of the jet use the resulting model. Control engineers use it to develop the control laws, the autopilot team uses it to make sure that their design meets specifications, and the systems team uses it to verify all possible contingencies and faults.

4.10.3 International Space Station Robotic Arm Testing

As a partner in the International Space Station (ISS), Canada is responsible for the verification of all tasks involving the special purpose dextrous manipulator (SPDM), depicted in Figure 4.22.

Testing the SPDM before deployment on the ISS represented a unique challenge, since it could not support its own weight when subjected to Earth's gravity! Therefore, the best way to test the robot controls was through HIL simulation [31].

One of the main technical challenges is to verify the feasibility of the insertion/extraction tasks by the robot hand self adaptive robotic auxiliary hand (SARAH) [32]. The forces involved are the result of complex frictional contact between the payload and the work site. Good contact models are being developed for non-real-time simulation but their real-time equivalents are still lacking precision. Moreover, the evaluation of different parameters is still mostly a trial and error process. Therefore, while most of the verification can rely on pure simulation, the contact part needs to be verified using a real robotic hand.

In the HIL test system, based on RT-LAB, a ground robot is driven by the output of the remote-operated space robot simulation. The system is shown in Figure 4.23. It is a HIL simulator consisting of

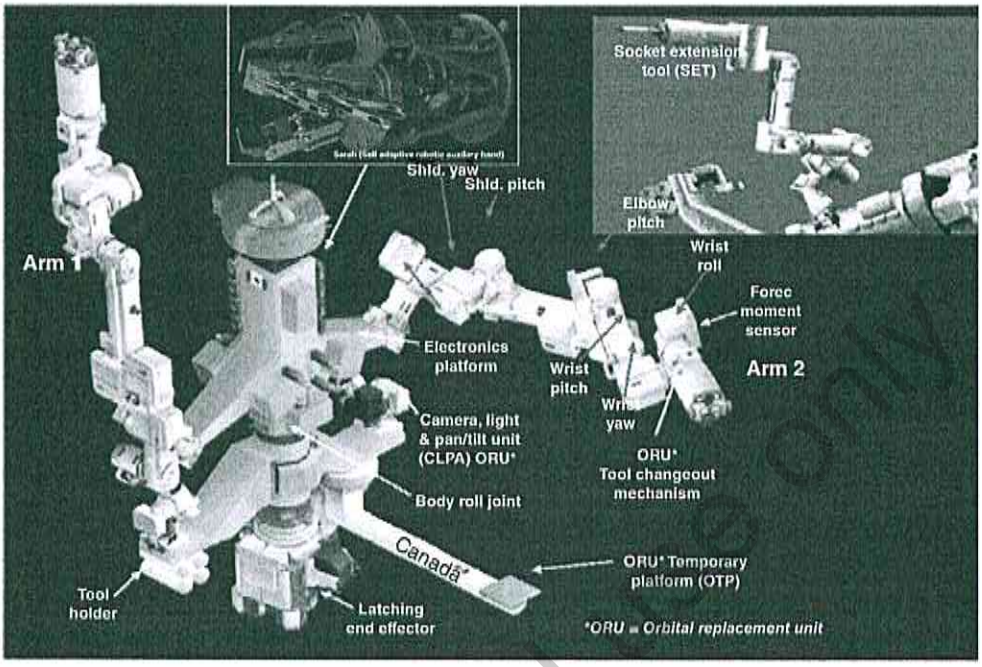

Figure 4.22 The Special Purpose Dextrous Manipulator of ISS.

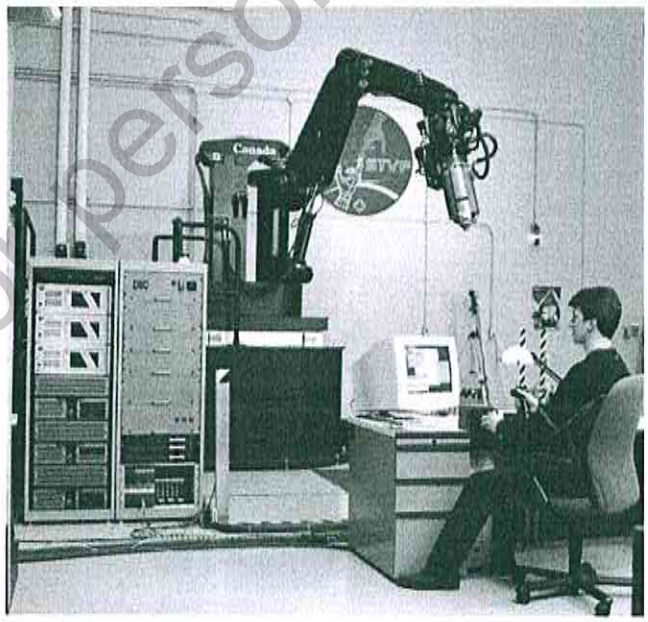

Figure 4.23 SPDM Hardware-in-the-Loop testing system.

a rigid robot with its control, a simulation of SPDM dynamics and a visualization engine. An operator controls the motion of the SPDM through the simulation engine that generates the endpoint motion of the SPDM. It is then used as a set-point for the robot controller that ensures that the robot endpoint follows the same trajectory as the SPDM model running in real-time. The contact forces are measured using force/moment sensors and fed back into the simulator to allow the dynamic simulation engine to react to external contact forces. This concept is very flexible since it can accommodate vibration of the space robot base and other phenomena. It can also be used to represent different space robots.

The control laws of the SARAH were also developed and tested using RT-LAB prior to integration on the SPDM.

4.11 Conclusion

This chapter has briefly presented various industrial applications of real-time simulation in the fields of power systems, motor drives, avionics and robotics. A brief description of the model-based design methodology was also presented, together with a discussion of the main challenges encountered in the design of real-time simulators. It was demonstrated that the most complex applications found during the integration of very complex grids can now be simulated in real-time to test actual control equipment using simulators based on standard computer systems.

As modern engineering projects become more complex, often with tight budgets and shortened development times, simulation technologies are becoming increasingly crucial to their success. We believe that modern engineering curricula would benefit from the inclusion of real-time simulation technology courses because of the widespread use of simulation technology, both by industry and by researchers.

References

[1] Kuffel, R., Geisbrecht, J., Maguire, T. *et al.* (1995) RTDS – a fully digital power system simulator operating in real time. WESCANEX – IEEE Conf. on Communications, Power, and Computing, Winnipeg, MA, Canada, 15–16 May 1995.

[2] Gagnon, R., Fecteau, M., Prud'Homme, P. *et al.* (2013) Hydro-Québec strategy to evaluate electrical transients following wind power plant integration in the Gaspésie transmission system. *IEEE Transactions on Sustainable Energy*, 3(4), 880–889.

[3] Dufour, C., Mahseredjian, J., Bélanger, J. and Naredo, J.L. (2010) An advanced real-time electromagnetic simulator for power systems with a simultaneous state-space nodal solver. IEEE/PES T&D – Latin America, São Paulo, Brazil, November 8–10, 2010.

[4] Devaux, O., Levacher, L. and Huet, O. (1998) An advanced and powerful real-time digital transient network analyser. *IEEE Transactions on Power Delivery*, **13**(2), 421–426.

[5] Dufour, C., Ould Bachir, T., Grégoire, L.-A. and Bélanger, J. (2012) Real-time Simulation of Power Electronic Systems and Devices, Chapter 15, in *Dynamics and control of switched electronic systems: advanced perspectives for modeling, simulation and control of power converters* (eds F. Vasca and L. Iannelli), Springer Series on Advances in Industrial Control, Springer. ISBN 978-1-4471-2885-4.

[6] Dufour, C., Cense, S., Ould Bachir, T. *et al.* (2012) General-purpose reconfigurable low-latency electric circuit and drive solver on FPGA. 38th IEEE Industrial Electronics Society (IECON), Montreal, QC, Canada, 25–28 October 2012.

[7] Wang, C., Li, W. and Bélanger, J. (2013) Real-time and faster-than-real-time simulation of modular multilevel converters using standard multi-core CPU and FPGA chips. 39th IEEE Industrial Electronics Society (IECON), Vienna, Austria, 10–13 November 2013.

[8] Dufour, C., Mahseredjian, J. and Bélanger, J. (2011) A combined state-space nodal method for the simulation of power system transients. *IEEE Transactions on Power Delivery*, **26**(2), 928–935.

[9] Dufour, C. and Bélanger, J. On the use of real-time simulation technology in smart grid research and development, to be published in *IEEE Transactions on Industry Applications*.

[10] Tremblay, O., Fecteau, M. and Prud'homme, P. (2013) Precise algorithm for nonlinear elements in large-scale real-time simulator. CIGRE conference, Montreal, QC, Canada, 24–26 September 2013.

[11] Dufour, C. and Tremblay, O. (2014) Iterative algorithms of surge arrester for real-time simulators. 18th Power Systems Computation Conference (PSCC), Wroclaw, Poland, 18–22 August 2014.

[12] Anderson, P.M. and Fouad, A.A. (1994) *Power System Control and Stability*, IEEE Press.

[13] Jalili-Marandi, V., Robert, E., Lapointe, V. and Bélanger, J. (2012) A real-time transient stability simulation tool for large-scale power systems. IEEE PES General Meeting, San Diego, CA, USA, July 2012.

[14] Hogan, E., Cotilla-Sanchez, E., Halappanavar, M. *et al.* (2013) Towards effective clustering techniques for the analysis of electric power grids. 3rd International Workshop on High Performance Computing, Networking and Analytics for the Power Grid (HiPCNA-PG), Denver, CO, USA, 22 November 2013.

[15] Pruvost, F., Cadeau, T., Laurent, P. *et al.* (2011) Numerical accelerations for power systems transient stability simulations. 17th Power System Computation Conference (PSCC), Stockholm, Sweden, 22–26 August 2011.

[16] Viehweider, A., Lauss, G. and Lehfuss, F. (2011) Stabilization of power hardware-in-the-loop simulations of electric energy systems. *Simulation Modelling Practice and Theory*, **19**(7), 1699–1708.

[17] Hong, M., Miura, Y., Ise, T. *et al.* (2010) Stability and accuracy analysis of power hardware-in-the-loop simulation of inductor coupled systems. *IEEJ Transactions on Industry Applications*, **130**(7), 902–912.

[18] Ren, W., Steurer, M. and Baldwin, T.L. (2008) Improve the stability and the accuracy of power hardware-in-the-loop simulation by selecting appropriate interface algorithm. *IEEE Transactions on Industry Applications*, **44**(4), 1286–1294.

[19] Ocnasu, D., Bacha, S., Munteanu, I. and Dufour, C. (2009) Real-time power-hardware-in-the-loop facility for shunt and serial power electronics benchmarking. 13th European Conference on Power Electronics and Applications (EPE), Barcelona, Spain, 8–10 September 2009.

[20] Vahidi, B., Ghorat, M. and Goudarzi, E. (2007) Overvoltage calculation on Bam substation by Monte Carlo method with accurate substation components modeling. IEEE PowerTech Conference, Lausanne, Switzerland, 1–5 July 2007.

[21] Paquin, J.N., Bélanger, J., Snider, L.A. *et al.* (2009) Monte-Carlo study on a large-scale power system model in real-time using eMEGAsim. IEEE Energy Conversion Congress and Exposition (ECCE), San Jose, CA, USA, 20–24 September 2009.

[22] Lesnicar, A. and Marquardt, R. (2003) An innovative modular multilevel converter topology suitable for a wide power range. IEEE Power Tech Conference, Bologna, Italy, June 2003.

[23] Grégoire, L.-A., Li, W., Bélanger, J. and Snider, L. (2011) Validation of a 60-level modular multilevel converter model – Overview of offline and real-time HIL testing and results. International Power Systems Transients (IPST), Delft, The Netherlands, July 2011.

[24] Saad, H., Dufour, C., Mahseredjian, J. *et al.* (2013) Real time simulation of MMCs using the state-space nodal approach. International Power Systems Transients (IPST), Vancouver, BC, Canada, 18–20 July 2013.

[25] Li, X., Liu, Y., Zhu, Y. *et al.* (2011) Real-time simulation of dynamic performance of multiinfeed UHVDC transmission system to be connected to North China power grid before 2015. *Power System Technology*, 2011-08.

[26] Harakawa, M., Yamasaki, H., Nagano, T. *et al.* (2005) Real-time simulation of a complete PMSM drive at 10 μs time-step. International Power Electronics Conf. (IPEC), Niigata, Japan, 4–8 April 2005.

[27] Dufour, C., Cense, S. and Bélanger, J. (2013) FPGA-based switched reluctance motor drive and DC-DC converter models for high-bandwidth HIL real-time simulator. 15th European Conference on Power Electronics and Applications (EPE-ECCE Europe), Lille, France, 3–5 September 2013.

[28] Dufour, C., Cense, S., Yamada, T. *et al.* (2012) FPGA permanent magnet synchronous motor floating-point models with variable-dq and spatial harmonic finite-element analysis solvers. 15th Int. Power Electronics and Motion Control Conference (EPE-PEMC), Novi Sad, Serbia, 4–6 September 2012.

[29] (2008) BANDEIRANTE: Information from Embraer to Brazil and abroad. *Embraer Journal*, **Ano 38**(730), 13–15.

[30] Cavalcanti, S. and Papini, M. (2004) Preliminary model matching of the Embraer 170 Jet: flight vehicle system ID-Part 1. *Journal of Aircraft*, **41**(4), 703–710.

[31] Piedboeuf, J.C., Aghili, F., Doyon, M. *et al.* (2001) Emulation of space robot through hardware-in-the-loop simulation. 6th Int. Symposium on Artificial Intelligence and Robotics & Automation in Space (i-SAIRAS), Canadian Space Agency, St-Hubert, QC, Canada, 18–22 June 2001.

[32] Martin, É., Lussier-Desbiens, A., Laliberté, T. and Gosselin, C.M. (2004) SARAH hand used for space operations on STVF robot. International Conference on Intelligent Manipulation and Grasping, Genoa, Italy, 1–2 July 2004.