Real Time Software-in-the-Loop Simulation for Control Performance Validation

Xiang Chen, Meranda Salem, Tuhin Das and Xiaoqun Chen

SIMULATION 2008; 84; 457

DOI: 10.1177/0037549708097420

Real Time Software-in-the-Loop Simulation for Control Performance Validation

Xiang Chen
Department of Electrical and Computer Engineering
University of Windsor
Windsor
Ontario N9B 3P4, Canada
xchen@uwindsor.ca

Meranda Salem
Department of Electrical and Computer Engineering
University of Western Ontario
London
Ontario N6A 5B9, Canada

Tuhin Das
Department of Mechanical Engineering
Rochester Institute of Technology
Rochester, NY 14623, US

Xiaoqun Chen
Hobart Canada
2875 East Bay Shore Road
Owen Sound
Ontario N4K 5P5, Canada

This paper illustrates an effective real-time software-in-the-loop (SIL) simulation technique for control design performance validation through two case studies in automotive systems: electric power steering (EPS) control and drive control for a switch reluctance motor (SRM) powered by a fuel cell. This technique, if implemented appropriately, will be able to shorten the prototyping time for control system research and development in both academic and industrial areas. The two cases presented involve complicated dynamics: road/tire steering dynamics and chemical/electrical dynamics in a fuel cell, for which software packages are available to simulate. Therefore, for the purpose of steering and SRM drive control performance validation, successful real-time simulation is desired through interfacing with these software, i.e. making software package in the loop. The case studies presented in this paper demonstrate the effectiveness of this concept. The presented real-time SIL simulation is conducted on a two-node computer platform engineered by RT-Lab, operating in fixed-step real time. Comparison between real-time and off-line simulation is also presented.

Keywords: Electro-mechanical system, fuel cell power system, motor drive, real-time SIL simulation, switch reluctance motor

SIMULATION, Vol. 84, Issue 8/9, August/September 2008 457–471

DOI: 10.1177/0037549708097420
Figures 1, 5–9, 12, 14–18 appear in color online: http://sim.sagepub.com

1. Introduction

When simulation goes to academic and industrial research and development practices, it is always expected to provide reliable and as accurate results as possible for validation of design concepts. In addition, there is need for great flexibility for implementing the simulation so that the structure of the simulation platform can be altered for design changes and can be re-used for other applications.

Without any question, the most reliable and accurate validation of intended design results should be the full-scale real test, i.e. testing the developed product in a real system for performance validation. Unfortunately, the cost of real tests is normally tremendous and, to some extent, may not be affordable and economically efficient for either academic or industrial researchers and developers. On the other hand, pure numerical simulation is now deemed as too fragile to accommodate complicated real processes, due to uncertainty in both mathematical models of physical systems and programming procedures, regardless of its extremely high flexibility and low cost. Therefore, it is always an economic and technical challenge to build a reliable yet low-cost real-time simulation platform for practical systems. This kind of simulation would lay down a solid base for real tests, if not replacing the real tests completely. This will reduce both the cost and the cycle time for product prototyping.

The importance and significance of simulation has been discussed [1]. In general, a good real-time simulation platform must possess two important features: fixed-step real-time computing and friendly interface to real hardware. Therefore fixed-step real-time operating systems such as QNX/Neutrino take the lead. Allowing real hardware to be in the simulation loop to communicate and exchange data with computing software in real time makes it possible to validate the design performance in a close-to-real testing environment, which constitutes the so-called Hardware-in-the-Loop (HIL) simulation [2]. The beauty of an HIL simulation platform is that it uses real hardware to replace the corresponding software codes to imitate the real test locally for the part to be tested. An HIL simulation can demonstrate all the benefits of numeric simulation for other parts of the whole system represented by software, provided they could be well modeled, as can be seen from a reported HIL simulation for vehicle system control [3].

The cost of an HIL simulation is a trade-off between a full-scale real test and a pure numerical simulation. However, the validation result from an HIL simulation could be made sufficiently reliable, comparable with that of a real test. Rapid prototyping is another approach that uses a software package to represent the design in a testing environment.

We also have the so-called software-in-the-loop (SIL) approach [4]. This approach may not be as robust as the HIL simulation. Howvever two successful case studies presented in this paper show that, if reliable software packages are available to simulate complex dynamics in various practical systems and could be used to replace real systems or hardware, then the real-time SIL simulation will also provide a reasonably reliable alternative solution to HIL simulation for validation and/or rapid prototyping purposes. In practice, the SIL simulation could either constitute the base for further real tests or provide confidence and guidance for implementing an HIL simulation.

In this paper, fixed-step real-time SIL simulation is presented for performance validation of an electric power steering (EPS) control system and drive control for a switch reluctance motor (SRM) powered by a fuel cell. The focus of this paper is on the concept proof of real-time SIL simulation mechanism instead of on the control design. The simulation reported is conducted in fixed step on a two-node personal computer platform hosted by RT-Lab. Real time interface has been successfully established between the relevant control blocks and two software packages: CarSim and Fuel Cell Simulator. The former simulates vehicle dynamics and the later simulates chemical/electrical dynamics in a fuel cell stack. The prototypes of these two simulations have their roots in automotive systems: an electric motor assisted steering system and fuel-cell powered traction in an electric car. Both simulations could be potentially and conveniently converted into an HIL simulation to include hardware such as a real controller or a real motor in the loop.

This paper is divided into five sections: after the Introduction, a brief introduction to a real-time simulation platform is presented; in Section 3 the EPS control case is presented from modeling to simulation set-up; in Section 4, simulation of drive control for SRM powered by a fuel cell is presented; Section 5 concludes the paper.

2. Real-Time Simulation Platform

The real-time SIL simulation presented in this paper is built and operated on the RT-Lab [2] platform produced by Opal-RT Inc.

2.1 RT-Lab Platform

The RT-Lab system, powered by real-time distributed computing technology, integrates fixed-step computing, high-speed communication and real-time interfacing technologies to reduce design and implementation time and cost, while increasing scalability and flexibility with no loss of performance. The RT-Lab platform enables model separation for distributed real-time execution and facilitates functions of automatically generating, downloading and operating (in real time) deadlock-free distributed simulation software codes.

The whole platform is configured consisting of a host station, a compilation software, target nodes, high-speed communication links and I/O boards.

1. Host Station/Computer: The host station is usually a PC workstation with a Windows operating system, which serves as the user interface. The host station allows users to:
    - edit and modify models with any popular model builder software such as Simulink^{TM};

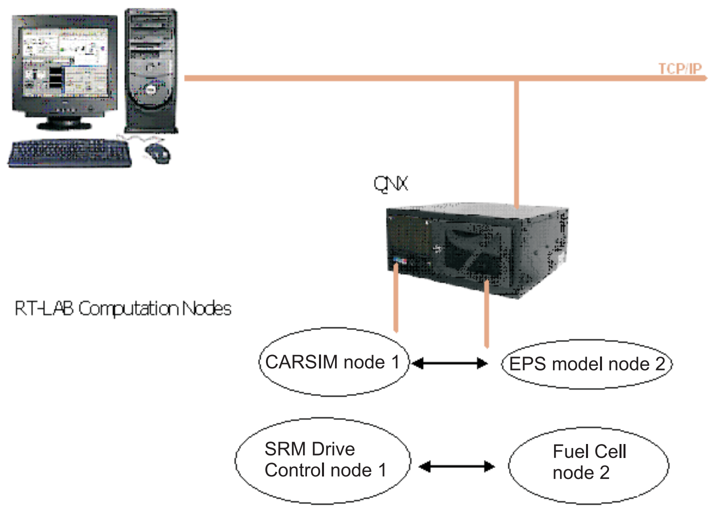

Figure 1. Schematic diagram of computing node

- generate machine codes;
- separate codes as needed; and
- control the simulator's GO/STOP sequences.

2. Target Nodes: Target nodes are simulation computers running in fixed-step real-time operating systems such as QNX/Neutrino. They are interconnected by high-speed communication protocols such as IEEE 1394 (Firewire) or cLAN (depending on the selected OS), as well as Input/Output boards for accessing external hardware equipments. Target nodes:
    - perform execution of simulation codes in real time; and
    - exchange data between nodes and between nodes and I/Os.
3. Compilation software: The compilation software is used to:
    - compile simulation blocks on the host station and generate C codes;
    - load codes onto each target node according to instruction; and
    - debug the user source codes.
4. Input/Output Board: Both analog and digital Input/Output boards are supported by RT-LAB. These allow real connection between the simulated part and external real equipment if used for applications such as real-time Hardware-In-Loop(HIL) simulation.

2.2 System Integration

One of the key issues in building the simulation mechanism illustrated in this paper is to integrate together the EPS control model or the SRM drive control model, which are designed from the relevant research, with CarSim or Fuel Cell Simulator software packages so that two different blocks could exchange data and are operated in synchronous real time. The schematic diagram of the system integration in terms of the computing workload is shown in Figure 1.

CarSim is a commercial software for simulating complicated vehicle dynamics. Therefore, in the EPS control performance validation, it is equivalent to replacing a real vehicle. The Fuel Cell Simulator is another software package developed for simulating chemical/electrical dynamics in a fuel cell stack. Significant work has been done to bring two different software blocks (see Figure 1) operating on different nodes to work together to achieve performance validation. The results show that SIL real-time simulation would be an effective low-cost solution for performance validation before conducting either real test or HIL simulation, as indicated by two case studies presented in the next section. A successful SIL simulation would then be able to reduce the prototyping time and cost for designers.

3. Case Study I: EPS Control System

An EPS system has been used to replace a traditional hydraulic power steering system in vehicles. There have been some research/development activities on power steering systems, either hydraulic or electric [5–8]. The modeling issue for an EPS system has been addressed in [9–12]. In principle, a power steering system is expected

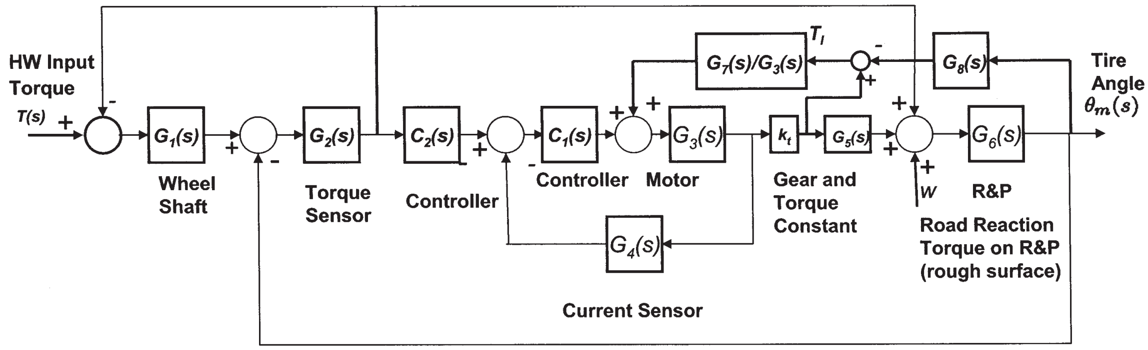

Figure 2. Block Diagram Model for EPS Control System

to yield a similar steering dynamic process as that produced by a manual steering system on a straight flat road surface, which should serve as the benchmark. Therefore, the goals of EPS control are not only to guarantee smooth redistribution of the steering torque load taken away from the driver to the electric motor, but also to be able to effectively attenuate various disturbance signals such as sensor noise, device parameter deviation and reaction disturbance due to various road surface roughness in order to make the driver's steering feeling enjoyable (i.e. light and less wheel vibrating).

In this section, we shall explain the SIL simulation for performance validation of EPS control design. The schematic structure of an EPS system normally consists of a hand wheel (HW), a torque sensor, a reduction gear, an intermediate shaft (I-shaft) and a rack and pinion structure. The block diagram model [10] of the EPS control system addressed is shown in Figure 2, where

$$G_1 = \frac{1}{J_{\mathrm{hw}}s^2 + B_{\mathrm{hw}}s}$$

represents the hand–wheel shaft dynamics; C_1 is the propotional, integral (PI) controller for motor; C_2 is the EPS motion controller; $G_2 = B_{\mathrm{s}}s + K_{\mathrm{s}}$ represents torque sensor dynamics;

$$G_3 = \frac{J_{\mathrm{m}}s + B_{\mathrm{m}}}{(J_{\mathrm{m}}s + B_{\mathrm{m}})(L_{\mathrm{m}}s + R_{\mathrm{m}}) + k_e k_t}$$

represents motor dynamics; $G_4 = K_c$ (constant) is current sensor; $G_5 = n$ is motor reduction gear;

$$G_6 = \frac{1}{(J_{\mathrm{r}} + n^2 J_{\mathrm{m}})s^2 + (B_{\mathrm{r}} + n^2 B_{\mathrm{m}})s}$$

is the equivalent rack and pinion dynamics;

$$G_7 = \frac{k_e}{(J_{\mathrm{m}}s + B_{\mathrm{m}})(L_{\mathrm{m}}s + R_{\mathrm{m}}) + k_e k_t};$$

$G_8(s) = n(J_{\mathrm{m}}s^2 + B_{\mathrm{m}}s)$; $T(s)$ is steering torque from the driver; w is disturbance torque from road surface; and $\theta_{\mathrm{m}}(s)$ is pinion angle.

The torque sensor G_2, including torsion bar mechanism, estimates the difference between the hand wheel shaft angle due to the driver's input torque and the tire angle position transmitted to the I-shaft through the rack and pinion structure. It communicates with the master controller C_2 which is derived through applying the $\mathcal{H}_2$ optimization approach [6]. This controller is designed to handle the disturbance signal while generating the desired torque load signal for the electric motor. C_1 is a PI controller to ensure that closed-loop block for the motor G_3 generates the required assisting steering torque effectively, according to the desired torque load signal. The reduction gear G_5 amplifies the turning torque further from the motor and the I-shaft, through the rack and pinion structure G_6, transfers the steering torque to the front tires. w is the road reaction disturbance torque consisting of both low- and high-frequency components due to both normal road reaction to the tires and (random) roughness of road surface.

3.1 Vehicle Steering Dynamics

Vehicle steering dynamics is essentially the tire steering dynamics on the road which involves highly nonlinear features and varies with changing road conditions. Therefore, for control design purpose, this information is not included in the model in order to avoid design complexity. Instead, the effect of vehicle/tire dynamics is modeled as a road reaction disturbance torque only (Figure 2).

However, to validate the performance of the obtained controller, it is important to conduct a simulation of the EPS control system. This includes full information regarding vehicle/tire dynamics, i.e. a fully modeled road reaction disturbance torque. For this account, CarSim [13] is implemented in the simulation. This software simulates full-scale vehicle dynamics and can be operated in both off-line (variable step) and fixed-step real-time modes on the RT-Lab platform [4]. In the interests of steering dynamics, CarSim simulates a manual steering process for a given hand wheel angle with respect to various road sur-

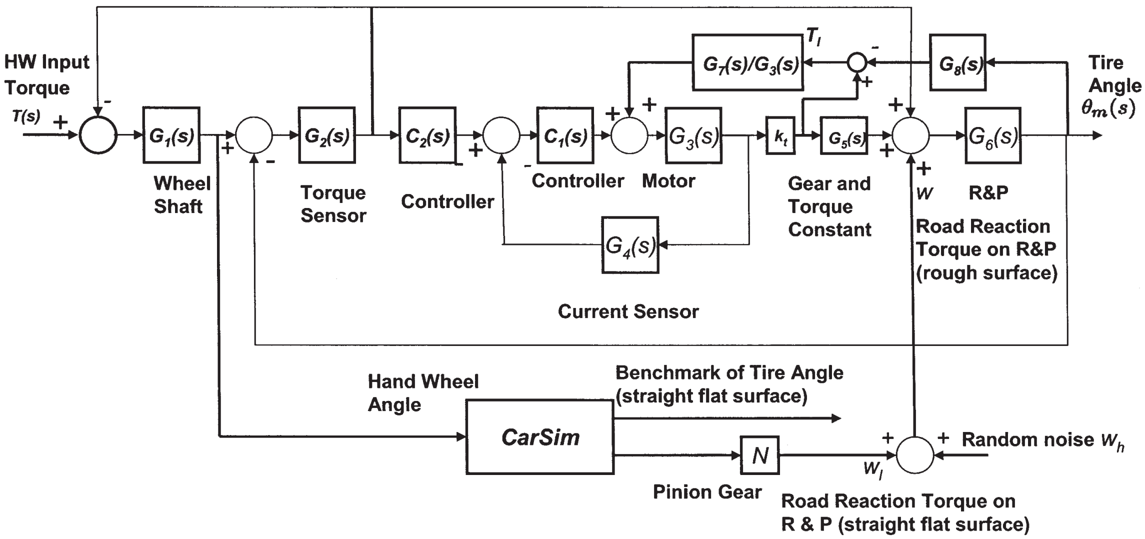

Figure 3. Block diagram of EPS control system with CarSim

face conditions, while the road reaction torque is available for the user. For the real-time SIL simulation of EPS control, CarSim is used to:

1. generate the benchmark of the tire position response to the hand wheel angle from the output of G_1, assuming straight flat road conditions;
2. generate an estimation of the road reaction torque w_l that is applied to provide low-frequency information of w, since no road roughness is assumed.

The block diagram of the EPS control system including CarSim is shown in Figure 3. Note that a random signal (high frequency) w_h is added so that the full information of road reaction torque $w = w_l + w_h$ is simulated for a rough road surface. In our simulation setup, the model in Figure 3 (including interfacing communication between the CarSim and other blocks of the simulation model) is first built in Matlab/Simulink on the Host Computer. The division of the model into two groups (CarSim and all other blocks) and the compiling of the divided model are done by the RT-Lab compiler to generate binary codes, downloaded into two target nodes with CarSim block on one and all other blocks on the other. Since the computing on target nodes is conducted in fixed steps of 0.1 ms (which is the minimum step required by the RT-Lab for the two nodes case) with the QNX/Neutrino operating system and the data exchange between the target nodes is through the firewire (IEEE1394), the real-time computing is guaranteed for all blocks in the model.

3.2 System Integration and Simulation Results

One of the key issues in building the SIL simulation mechanism for the EPS control system is to integrate the EPS control model and CarSim software package together in the RT-Lab platform so that the two software models could exchange data and are operated in real time. In the simulation, the CarSim is downloaded to the distributed node 1 and the EPS control model is on node 2. The real-time data exchange between the control model and the CarSim is done through IEEE 1394 (Firewire).

To verify the EPS control design results, real-time simulation has been conducted on RT-Lab platform for a sample model of EPS control systems including a motor. The simulation is conducted for w_h with different variance, reflecting different road surface roughness. In the simulation the vehicle speed is assumed to be a constant, otherwise a parameter adjustment would be needed for safety concerns. The hand wheel gives a left-turn torque signal with $1\,\mathrm{N\,m^{-1}}$ torque with a 5 s duration and is then released (Figure 4).

For clarification, the following signals are chosen to verify the simulation results: road reaction torque, torque sensor output, hand wheel and tire positions. The simulation results are shown in both off-line and real-time operation in comparison. The off-line simulation results are used as the simulation benchmarks and the success of real-time SIL simulation is judged by its closeness to the off-line counterparts. The results are presented below.

1. *Road reaction torque and torque sensor output*: Road disturbance signal (high frequency) is chosen as a zero-mean random number with variance of 10. The variable step (off-line) simulation results are shown in Figure 5 for $\mathcal{H}_2$ control, while the real-time simulation results are shown in Figure 6. The following conclusions are drawn from Figures 5 and 6.

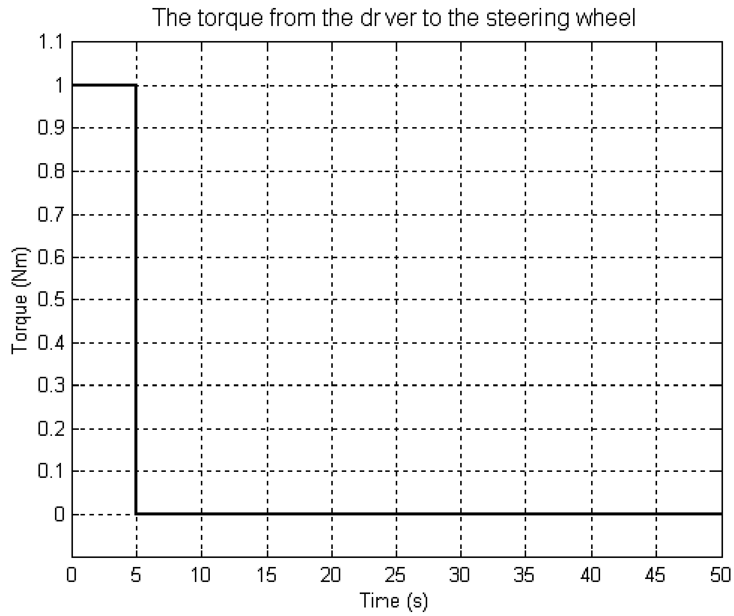

Figure 4. Human turning torque applied to hand wheel

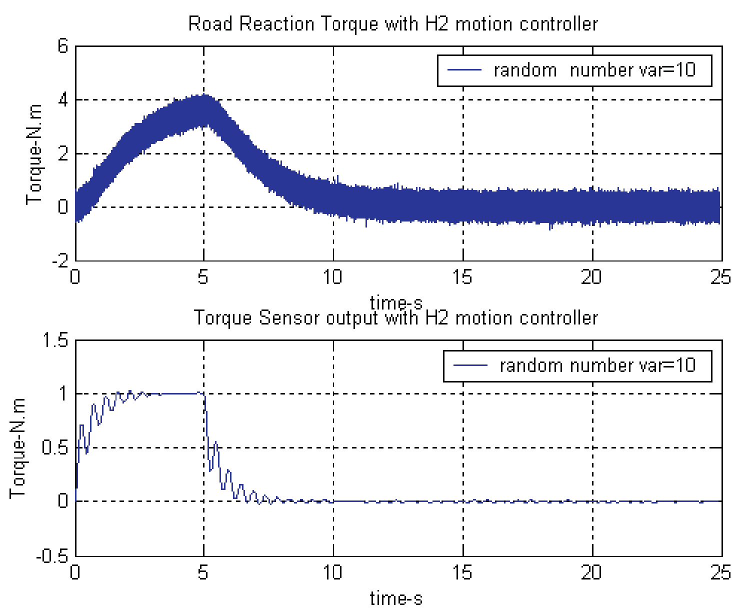

Figure 5. Off-line road reaction torque and torque sensor output with $\mathcal{H}_2$ control (variance = 10)

1. (a) Performance is validated: it is obvious that the effect of rough road surface is very much attenuated by the additional $\mathcal{H}_2$ control as the driver's rough feeling (the output of torque sensor) is very much suppressed.

   (b) Comparing corresponding results in variable and fixed-step simulation, it can be seen that the trend of real-time results are close to that of off-line simulation results, which validates the effectiveness of the real-time SIL simulation mechanism.

2. *Hand wheel and tire positions*: The system response tire angle is shown in contrast with the hand wheel

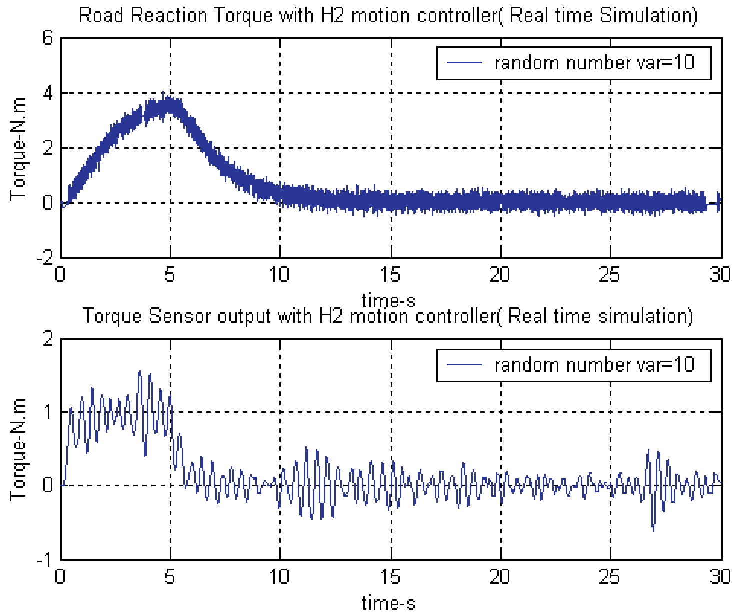

Figure 6. Real-time road reaction torque and torque sensor output with $\mathcal{H}_2$ control (variance = 10)

angle as the reference input in Figures 7 and 8, for both off-line and real-time SIL simulation, under a random disturbance with variance 10. It can be seen that the tire angle follows the reference hand wheel angle (which is generated by applying the human steering torque) in time. Once again, the closeness between off-line and real-time simulation results are observed.

3. *Comparison with steering dynamics benchmark*: As stated before, the benchmark for the steering dynamics is reasonably presented as the tire angle of CarSim responding to the hand wheel shaft angle signal in the EPS control system, assuming straight flat road surface (see Figure 3). The tire angle signal in the EPS control system, under road reaction torque and a random number disturbance signal with variance of 10 reflecting the roughness of road surface, is compared in a real-time simulation with the benchmark, as shown in Figure 9.

It can be judged from these simulation graphs that the proposed EPS controllers yield good performance in terms of bringing the steering process on rough road surface close to that on a flat road surface, while keeping the driver's steering feeling enjoyable-light and less vibration.

4. Case Study II: SRM Drive Control

Switched reluctance motor (SRM) drive has been considered as a possible alternative in variable speed applications because of its obvious advantages [14, 15]: rugged and simple construction, inherent variable speed capability and ease of control. SRM drive so far has been used in aircraft starter/generator systems, automotive and home appliance applications. SRM drive is also known for its fault-tolerant operation. In this section, a model of SRM drive control [16] is described and will be used to illustrate the real-time SIL simulation mechanism. The block diagram model of SRM drive speed control is shown in Figure 10.

SRM is a salient machine with an unequal number of stator and rotor poles. Normally, the ratio of the number of stator poles to rotor poles is 6:4 or 8:6. In this paper, the inductance-based model of an 8:6 four-phases SRM drive is used for simulation, which is summarized as follows.

1. The kth ($k = 1, 2, 3, 4$) phase voltage equation is given by

$$V_k = R_k i_k + \Omega^T(\theta)\left(\Lambda(i_k) + i_k \frac{d\Lambda(i_k)}{di_k}\right)\frac{di_k}{dt} + i_k \omega \frac{d\Omega^T(\theta)}{d\theta}\Lambda(i_k), \quad (1)$$

where

$$L(i_k, \theta) = \Omega^T(\theta)\Lambda(i_k),$$

$$\Lambda(i_k) = \begin{bmatrix} L_a(i_k) & L_{\mathrm{m}}(i_k) & L_u(i_k) \end{bmatrix}^T,$$

and

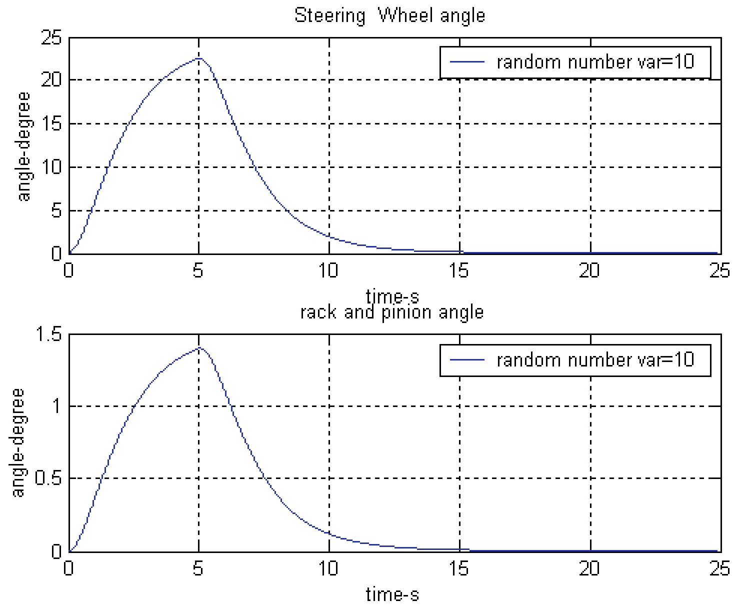

Figure 7. Off-line hand wheel position and tire position with $\mathcal{H}_2$ control (variance = 10)

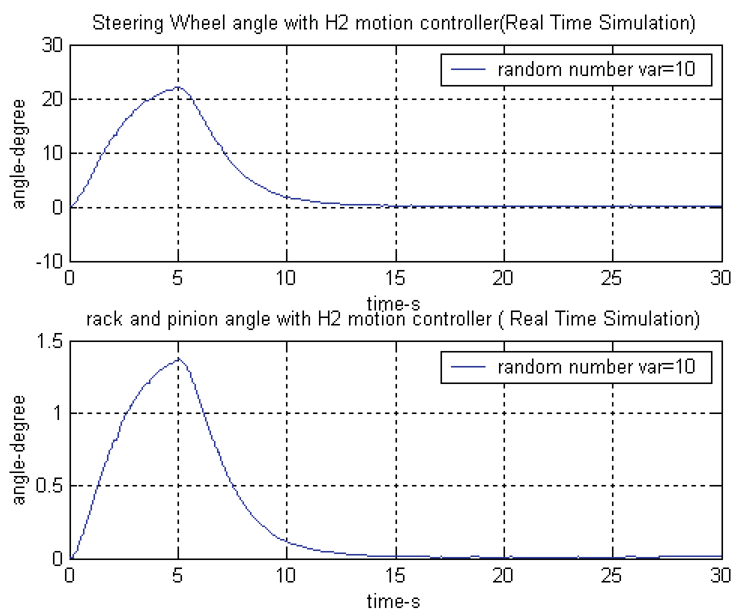

Figure 8. Real-time hand wheel position and tire position with $\mathcal{H}_2$ control (variance = 10)

$$\Omega^T(\theta) = \begin{bmatrix} 0.25 + 0.5\cos(6\theta) + 0.25\cos(12\theta) \\ 0.5 - 0.5\cos(12\theta) \\ 0.25 - 0.5\cos(6\theta) + 0.25\cos(12\theta) \end{bmatrix}.$$

2. Torque and mechanical equations are given by

$$T_\mathrm{m} - T_l = J\frac{d\omega}{dt} + B\omega, \qquad (2)$$

where

$$T_\mathrm{m} = \sum_{k=1}^{4} \frac{\partial \Omega^T(\theta)}{\partial \theta} \int_0^{i_k} \Lambda(i_k) i_k d i_k.$$

In the simulation presented in Section 4.2, $J = 0.0016\,\mathrm{kg\,m^2}$ and $B = 0.004\,\mathrm{N\,m\,s\,rad^{-1}}$ while $T_l = 1\,\mathrm{N\,m}$ where $t > 0$, i.e. a constant load is assumed for simplicity.

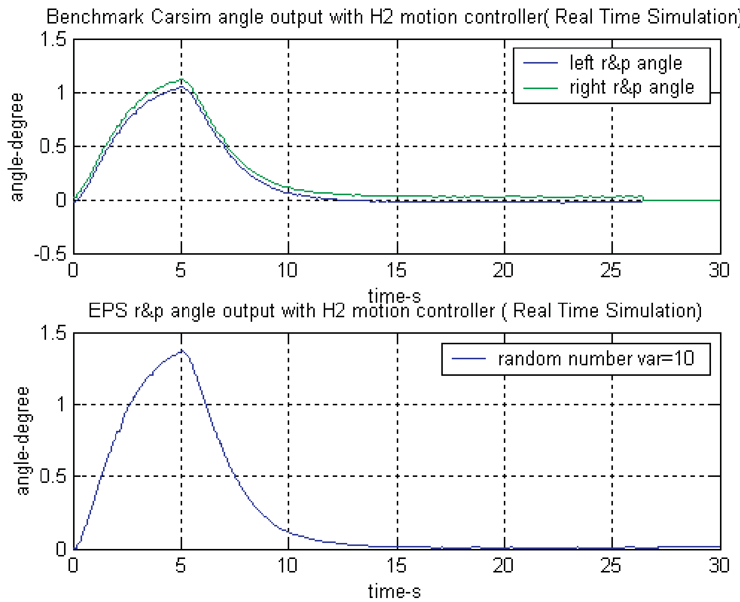

Figure 9. Comparison with benchmark

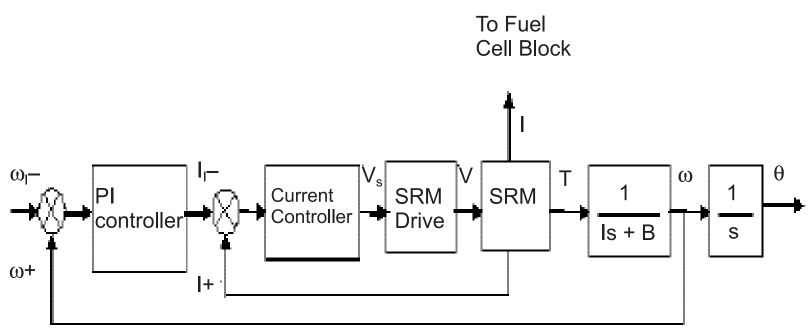

Figure 10. Block Diagram of SRM Drive Speed Control

3. PI control equations are given by

$$I_r = K_p\delta\omega + K_i \int_0^t \delta\omega dt, \quad \delta\omega = \omega_r - \omega. \quad (3)$$

The PI control is designed to regulate the desired current so that the SRM drive can vary and keep the desired speed. The controller parameters are obtained by aiming to keep overshoot less than 5% and the rise time less than 5 ms, while assuming that the electric process inside the SRM is much faster than the mechanical process of the SRM drive. For this account, $K_p = 0.892$ and $K_i = 256$ [17].

4.1 Fuel Cell Stack

A fuel cell [18] is an electrochemical device that converts the chemical energy of gaseous fuel directly into electricity and is widely regarded as a potential alternative stationary and mobile power source. A fuel cell reduces the ubiquitous dependence on fossil fuels and therefore has significant environmental implication. A fuel cell stack system is under intensive development by several manufacturers, with the Proton Exchange Membrane (PEM) fuel cell (also known as the Polymer Electrolyte Membrane fuel cell) currently considered to be in a relatively more advanced stage for ground vehicle applications [19].

There is much interest in developing fuel cell powered vehicles from both the government and automobile original equipment manufacturers. In the vehicular application of fuel cell power, transient behavior is one of the key issues for the success of fuel cell powered vehicles. During the transient process, the fuel cell stack control system is required to maintain optimal temperature, membrane hydration and partial pressure of the reactants across the

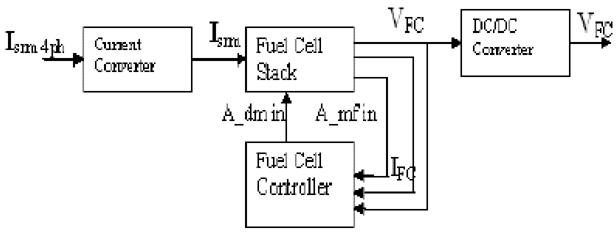

Figure 11. Block diagram of a fuel cell stack and control

membrane in order to avoid degradation of the stack voltage, therefore maintaining high efficiency and extending the life of the stack.

There are different types of fuel cells, distinguished mainly by the type of electrolyte used. A particular fuel cell may only be suitable for a narrow range of applications, due to the differences in characteristics such as cell material, operating temperature and fuel diversity. In the real-time SIL simulation presented in this paper, a fuel cell is assumed to power the SRM drive. The block diagram is shown in Figure 11.

The transients of a fuel cell stack system including the fuel cell controller are modeled by the software Fuel Cell Simulator, provided by Emmeskay. This fuel cell stack model is applied to power the SRM drive and the SRM four-phase currents are fed back through a current converter to control the fuel cell voltage. At the fuel cell voltage output, a DC/DC converter is added to boost the fuel cell voltage to drive the SRM and to provide isolation between fuel cell and SRM.

4.2 System Integration and Simulation Results

As in the EPS control system example, one of the key issues in building the SIL simulation mechanism illustrated in this example is to integrate the SRM drive speed control model and the fuel cell stack package (Figure 12) together in the RT-Lab platform so that both models could exchange data and operate in real time. In the simulation, the Fuel Cell Simulator software package is downloaded to node 2 while the SRM drive control blocks are on node 1. Once again, the data exchange between these two parts is done through IEEE 1394(Firewire) with a computing step time of 0.2 ms for the real-time simulation. The block diagram of the whole system is shown in Figure 13.

As shown, the fuel cell provides voltage through the DC/DC chopper to the SRM drive system while the SRM four-phase currents are fed back through the current converter to regulate the fuel cell voltage. Through these connections, the SRM drive speed control model exchanges data in real time with the fuel cell stack model operating on a two-nodes RT-Lab platform with the SRM drive speed control model on one and the fuel cell stack model on the other. Although the electrical process in the SRM drive is ignored when designing the PI control, circuit models for the hysteresis current controller and the SRM driver [20] are added in the simulation. This guarantees that a high-fidelity model is used in simulation to test the validity of the speed control performance designed based on the low-order model of SRM.

For the real-time SIL simulation presented in this paper, an SRM drive with the following specifications is chosen. The mechanical load is assumed to be a constant $T_l = 1\,\mathrm{N\,m}$, $t > 0$ for simplicity. Parameters of PI controller used in the simulation are $K_p = 0.892$ and $K_i = 256$ [17].

The fixed-step real-time SIL simulation results are validated by its off-line (variable step) simulation counterparts, as shown in Figures 14 and 15 for SRM torque and speed. As references, both off-line and real-time simulation results of SRM current and fuel cell voltage before and after the DC/DC chopper are also presented (Figures 16–18).

4.2.1 Notes on Simulation Results

1. Although the difference between off-line and real-time simulation results is small, the trends indicate the validity of real-time SIL simulation mechanism to a great extent. It is worth pointing out that this real-time SIL simulation mechanism could be potentially converted into a hardware-in-the-loop simulation to include real devices such as a real fuel cell controller, real SRM drive circuits, a real PI controller or even a real SRM itself, depending on what needs to be tested.

2. It is noted that, due to the regulating action of the PI controller, both speed and torque yield quite smooth responses with no ripples.

3. Off-line and real-time results of SRM current are shown in four phases.

4. Regulating action of fuel cell control (inside Emmeskay model) yields the desired smooth output voltage.

5. Conclusion

In this paper, two case studies are presented to illustrate real-time SIL simulation mechanisms: an electric power steering control system and an SRM drive control system. It is shown that the real-time integration of the design blocks with commercial software packages is an encouraging idea to conduct effective real-time control performance validation.

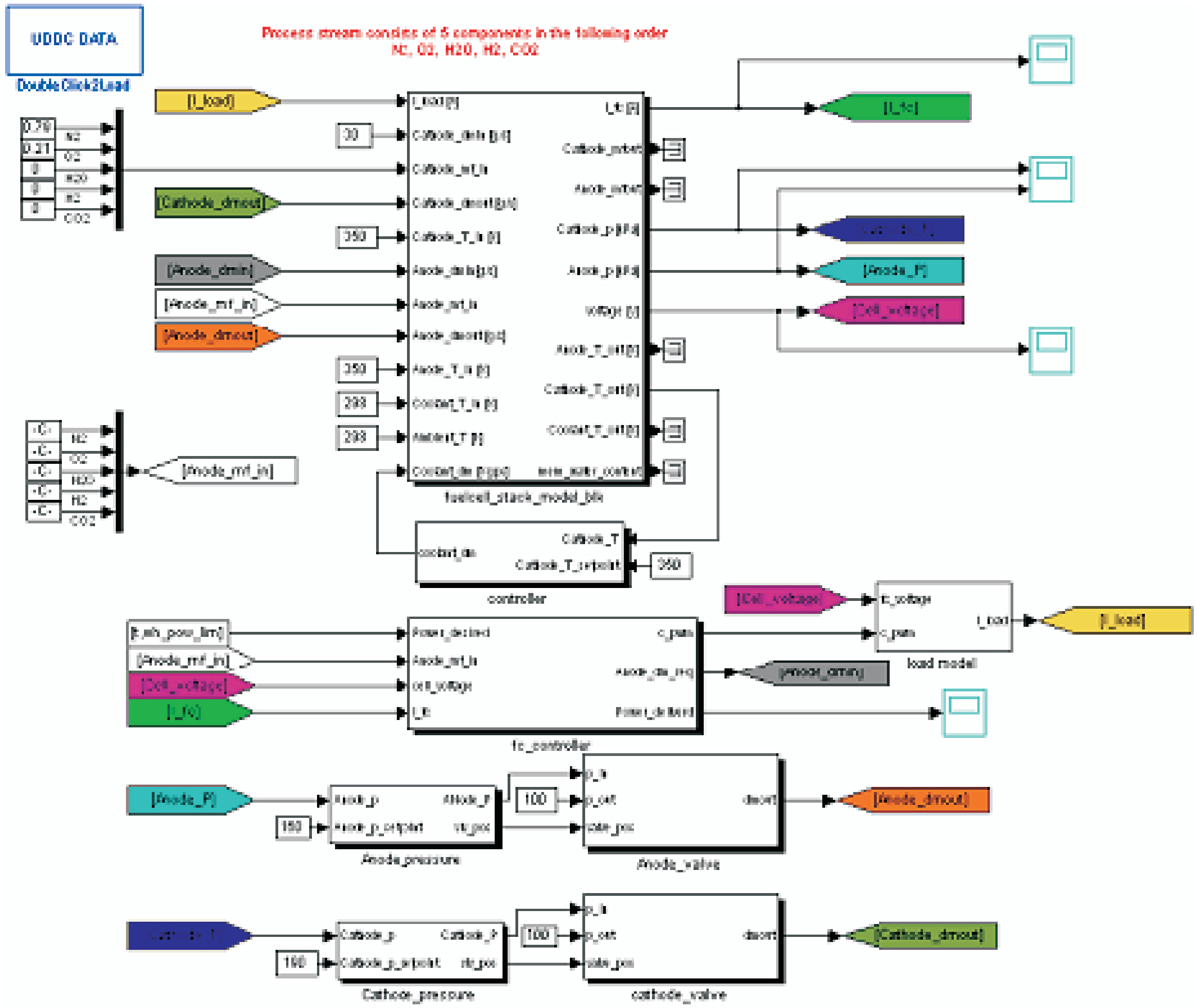

Figure 12. Simulation software for fuel cell stack developed by Emmeskay

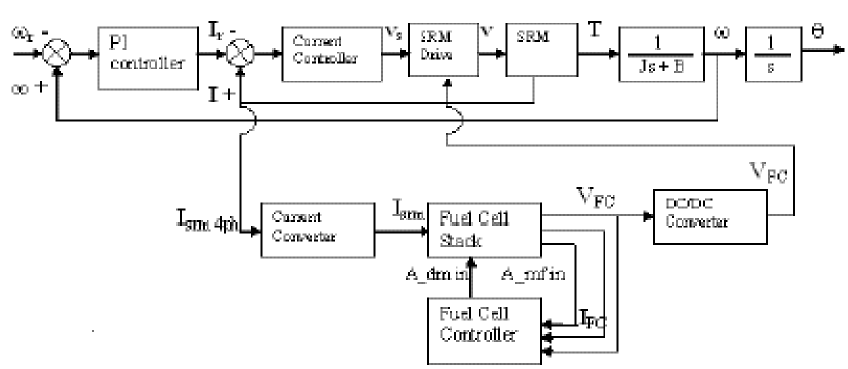

Figure 13. Real-time simulation of integrated SRM drive and fuel cell stack

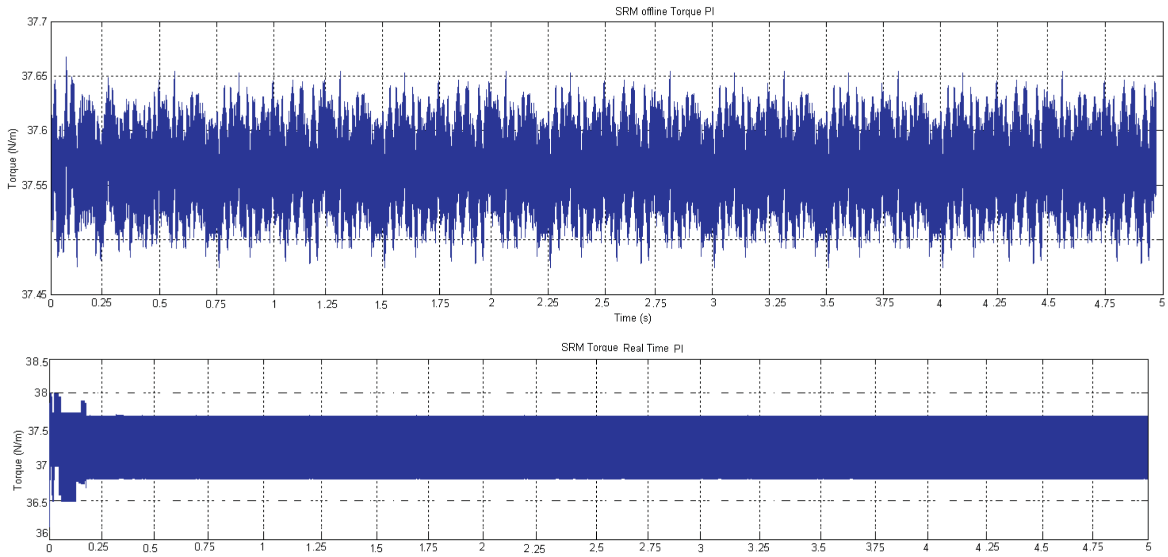

Figure 14. Off-line and real-time simulation of SRM drive torque

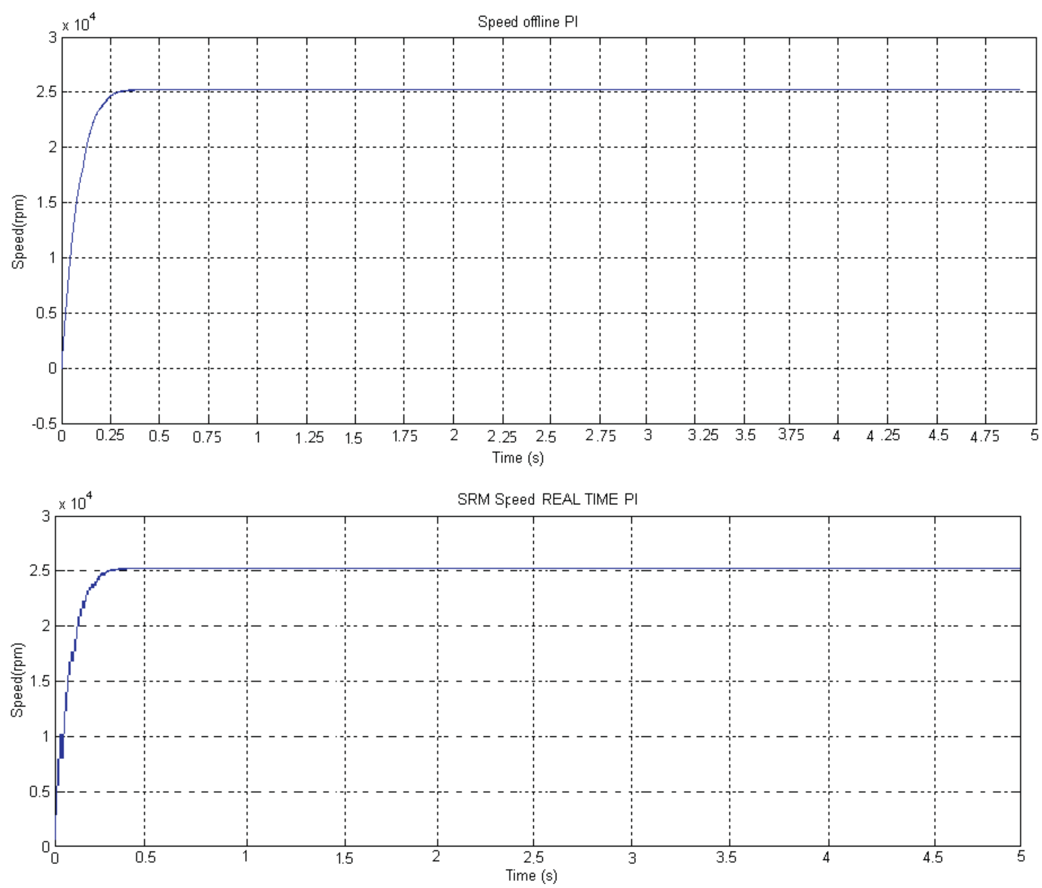

Figure 15. Off-line and real-time simulation of SRM drive speed

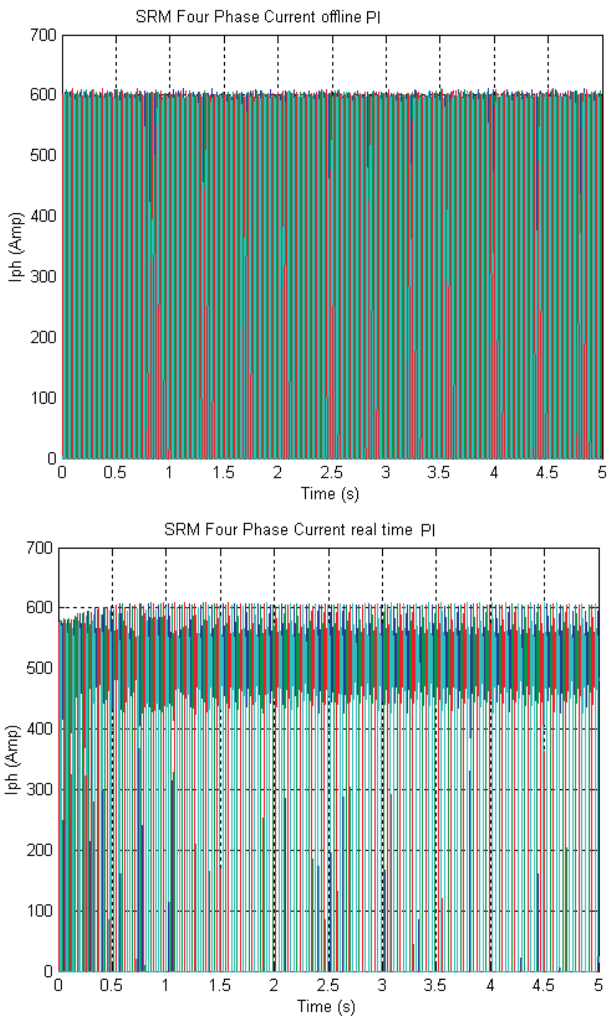

Figure 16. Off-line and real-time simulation of SRM current

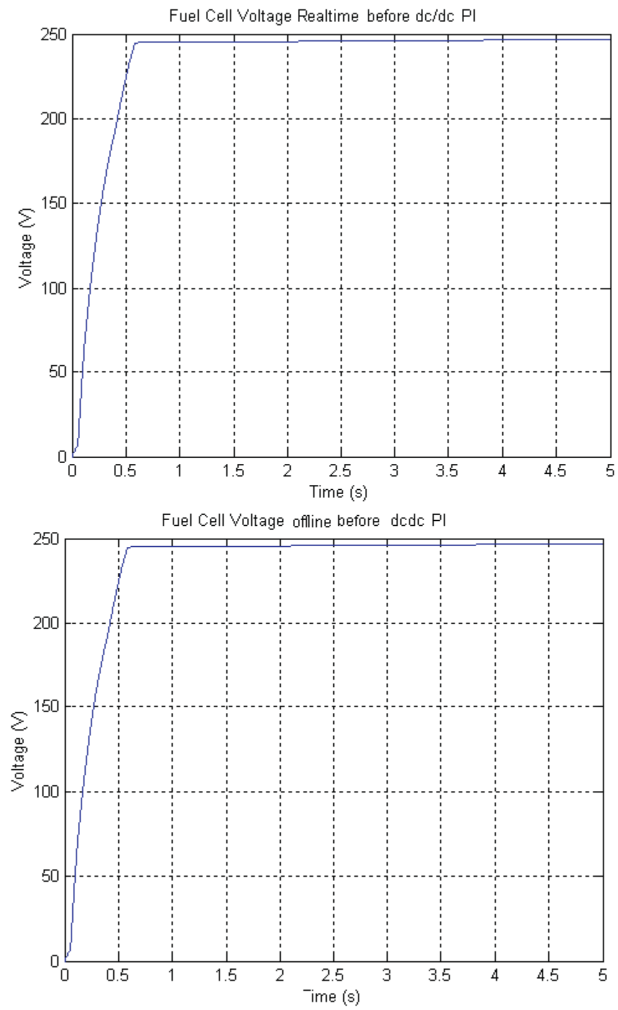

Figure 17. Off-line and real-time simulation of fuel cell voltage before DC/DC chopper

For the electric power steering control system, the two-controller structure is designed based on a simplified EPS system model but the real-time SIL simulation is done by incorporating details of vehicle dynamics simulated by software CarSim to test the performance.

For the SRM drive control system, the PI speed controller is designed based only on SRM mechanical dynamics, allowing a simplified model. However, the real-time SIL simulation is done by incorporating all details of the current controller and the driver of SRM as well as a fuel cell stack to test the performance. Both off-line and real-time SIL simulation results for both cases are presented for validation purposes. The results show that this simulation mechanism is a low-cost feasible approach to support electro-mechanical system related research and development work as well as fuel cell and motor drive control related research and development work. It is also noted that this real-time SIL mechanism could be converted easily into a hardware-in-the-loop simulation to include real hardware.

7. Acknowledgements

This research is partly supported by NSERC Grant to the first author. The authors wish to acknowledge the technical support provided by Opal-RT and Mechanical Simulation Corp.

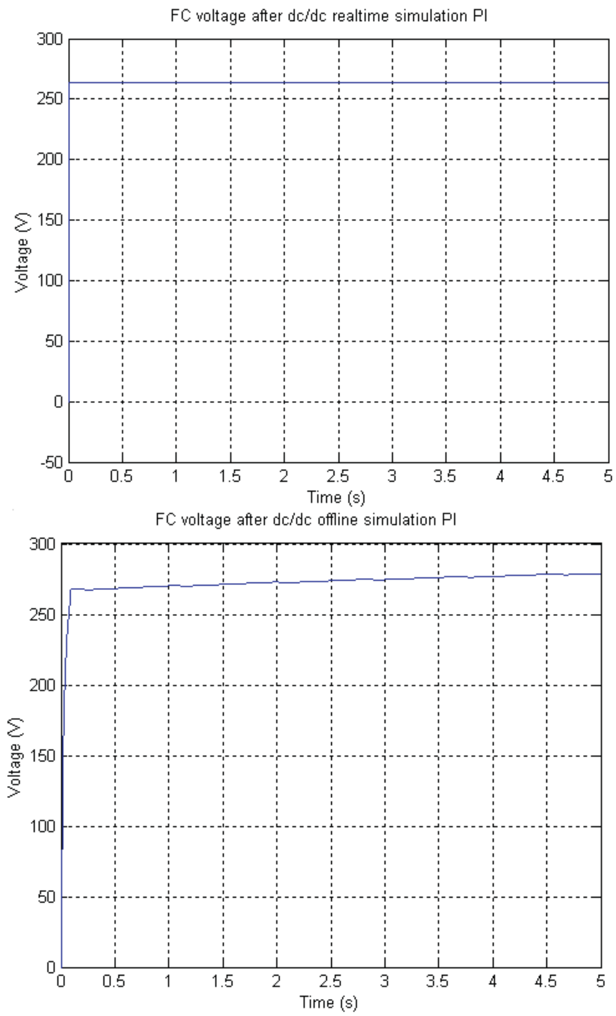

Figure 18. Off-line and real-time simulation of fuel cell voltage before DC/DC chopper

8. References

[1] Astrom, K. J., H. Elmqvist and S. E. Mattsson. 1998. Evolution of Continuous-Time Modeling and Simulation. In *Proceedings of the 12th European Simulation Multiconference*, pp. 1–6.

[2] Baracos, P., G. Murere, C. A. Rabbath and W. Jin. 2001. Enabling PC-Based HIL Simulation for Automotive Applications. *Proceedings of the IEEE International Electric Machines and Drives Conference*. Technical Report, Opal-RT Technologies, Inc. (IEMDC'01), Cambridge, MA, 2001, USA.

[3] Yamazaki, M., S. Sureshbabu, M. Loftus, R. Crandell and M. Brackx. 2002. Analysis of Automatically Generated Vehicle System Control Software in a HIL Environment. In *Proceedings of American Control Conference*, pp. 3135–3140.

[4] Kwon, W. and S. G. Choi. 1999. Real-Time Distributed Software-In-the-Loop Simulation for Distributed Control Systems. *Proceedings of IEEE. International Symposium of Computer-Aided Control System Design*, Hawaii, pp. 115–119.

[5] Burton, A. W. 2003. Innovation Drivers for Electric Power-Assisted Steering. *IEEE Control Systems Magazine* 6, 30–39.

[6] Chen, X., X. Chen and K. Zhou. 2005. Optimal Control of Electric Power-Assisted Steering System. In *Proceedings of 2005 IEEE Conference on Control Application*, pp. 1403–1408.

[7] Ferries, G. R. and R. L. Arbanas. 1997. Control/Structure Interaction In Hydraulic Power Steering Systems. In *Proceedings of the ACC*, Vol. 2, pp. 1146–1151.

[8] Zaremba, A. T. and R. I. Davis. 1995. Dynamic Analysis and Stability of a Power Assist Steering System. In *Proceedings of the ACC*, pp. 4253–4257.

[9] Badawy, A., J. Zuraski, F. Bolourchi and A. Chandy. 1999. Modeling and Analysis of an Electric Power Steering System. SAE Technical Paper, 1999-01-0399.

[10] Chen, X. and X. Chen. 2006. Control-Oriented Model for Electric Power Steering System. SAE Paper, 2006-01-0938, 2006.

[11] Gene Liao Y. and H. I. Du. 2003. Modeling and analysis of electric power steering system and its effect on vehicle dynamic behavior. *International Journal of Vehicle Autonomous Systems (IJVAS)* 1, 153–166.

[12] Parmar M. and J. Y. Hung. 2002. Modeling and Sensorless Optimal Controller Design for an Electric Power Assist Steering System. In *Proceedings of the IECON 02*, Vol. 3, pp. 1784–1789.

[13] Han, S. *User's Manual of CarSim, the Mechanical Simulation Corporation, Version 5.11*. November 2001.

[14] Lawrenson, P. J., J. M. Stephenson, P. T. Blenkinsop, J. Corda and N. N. Folton. 1980. Variable-Speed Switched Reluctance Motors. *Proceedings of IEEE* 127B(4), 253–265.

[15] Miller, T. J. E. 1993. *Switched Reluctance Motor and Their Control*. Magna Physics Publishing and Oxford Science Publications, London.

[16] Ehsani, M. and S. Abourida. 2003. Design, Analysis and Control of Electrical Motor Drives Using Hardware-in-the-Loop Simulation. In *Proceedings of Opal-RT Workshop*, Livonia, Michigan.

[17] Salem, M., X. Chen, T. Das and S. Gopalswamy. 2005. Drive Control for Switched Reluctance Motor in A Fuel Cell Power System. In *Proceedings of American Control Conference*, pp. 2727–2732.

[18] Appleby, A. J. and F. R. Foulkes. 1989. *Fuel Cell Handbook*. Van Nostrand Reinhold, New York.

[19] Fronk, M. H., D. L. Wetter, D. A. Masten and A. Bosco. 2000. PEM Fuel Cells System Solution for Transportation. SAE Paper no. 2000-01-0373.

[20] Salem, M., T. Das, X. Chen, S. Akella and S. Sivashankar. 2005. Real Time Simulation for Switched Reluctance Motor Powered by A Fuel Cell System. In *Proceedings of ASME-Power Conference*, PWR2005–50090.

Dr. Xiang Chen *received his MSc and PhD degrees from Louisiana State University, in 1996 and 1998 respectively. He is currently an Associate Professor at the Department of Electrical and Computer Engineering, University of Windsor, Canada. His research interests include robust control, complex systems and networks, as well as real time hardware/software-in-loop simulation.*

Meranda Salem *received a BASc and MASc Degrees in electrical and computer engineering from University of Windsor, Canada, in 2001 and 2004 respectively. Presently, she is a PhD candidate at University of Western Ontario, Canada. Her research interests are in the area of Resistance Spot Welding (RSW) control in the automotive industry and in the automotive control in general.*

__Dr. Tuhin Das__ received a BTech. degree in Mechanical Engineering from the Indian Institute of Technology, and MS and PhD degrees from Michigan State University. He is an assistant professor of Mechanical Engineering at Rochester Institute of Technology. He was also a visiting assistant professor in the Mechanical Engineering department at Michigan State University. His research interests are in the area of modeling and control of energy systems.

__Xiaoqun Chen__ received a BEng. degree from University of Electronic Science and Technology of China, and an MASc degree from University of Windsor, Canada, in 1999 and 2005, respectively. She is an electrical engineer with Hobart Food Equipment Group, Owen Sound, Canada. From 2004 to 2005, she was a research assistant with the Department of Electrical and Computer Engineering at University of Windsor. Her research interest is in control algorithms and real-time simulations for automotive applications.