Proceedings of the 2000 Winter Simulation Conference
J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, eds.

EFFECTIVE REAL-TIME SIMULATIONS OF EVENT-BASED SYSTEMS

C.A. Rabbath
M. Abdoune
J. Belanger

Opal-RT Technologies Inc.
1751 Richardson, Suite 2525
Montreal, PQ, H3K 1G6, CANADA

ABSTRACT

This paper presents a set of novel tools that allow the efficient simulation, at fixed time steps, of event-based dynamic systems. The so-called RT-Events library is an innovative toolbox that can be used with the Simulink™ graphical software and that solves the following two problems encountered in the simulations of event-based systems: (1) time consuming variable-step algorithms; and (2) inaccurate real-time simulations with fixed-step algorithms. One important application of the new RT-Events toolbox is its capability to effectively simulate automotive systems as real-time, hardware-in-the-loop systems. It is shown that the simulations performed with the new tools are more efficient than the conventional algorithms. In particular, the important problem of reset walk, which is inherent to the classical fixed-step simulation of event-based systems, is explained and its solution obtained with the use of the blocks of the new toolbox is demonstrated. Numerical examples illustrate the effectiveness of the new simulation tools.

1 INTRODUCTION

Nowadays, the simulation engineer is faced with the task of simulating complex dynamic systems. The high level of complexity may come from the very nature of certain continuous-time and discrete-time systems that have a changing dynamical behavior depending on the occurrences of so-called discrete events. These systems are named event-based systems in the present paper. The term hybrid dynamic system is also used; for instance, in (Cassandras 1993). A practical example of an event-based system is the internal combustion engine of an automobile. Such an event-based system is characterized by the crankshaft angle of the engine which determines one of four cycles; namely intake, compression, combustion and exhaust. Thus, the in-cylinder dynamics are determined by the crankshaft event. In the development of a power-train system, the simulation engineers usually perform a simulation of the engine dynamics in closed-loop with a simulated control unit, and eventually carry out a hardware-in-the-loop (HIL) simulation of the simulated engine dynamics in closed-loop with the actual electronic control unit (ECU). However, there are some problems encountered in the simulation, at fixed time steps, of event-based systems. For instance, the events associated with the crankshaft angle can take place between the fixed time steps at which the simulation is executed. This causes a discrepancy between the actual event occurrence and the discrete-time detection of the event by the simulated system. Consequently, there are differences between the simulated and the actual engine dynamics. It was shown in (Rabbath and Abdoune 1999) that the error between simulated and actual dynamics can in fact render the real-time simulations untruthful and misleading to the engineers testing the power-train control system.

One can easily understand the types of problems involved with fixed-step simulations of event-based systems by considering, for instance, the process of integrating a continuous signal with reset of the integrator done at time instants other than integral multiples of the fixed step size. In this case, there will be errors between the output of the simulated integrator and that of the theoretical integration. This is so simply because the simulated integrator can be reset at integral multiples of the fixed step size, and not at the time instants at which the theoretical integrator is reset. However, keeping in mind that the simulations should approximate as closely as possible the behavior of the actual integrator, the discrepancy between simulated and actual integrators creates a problem of accuracy of simulation. One approach to solve the problem of accuracy, that is to effectively simulate event-based systems, is to use steps of variable sizes. A simulation using variable step sizes is based on an algorithm that selects a step size according to signal tolerances, disregarding time-related constraints. This sort of scheme has the disadvantage of possibly resulting in time consuming simulations. Furthermore, if one is interested in assuring compatibility of the simulations with hard real-time constraints, the variable step size solution has to be discarded; that is, with an eventual connection to input/output boards and to a timer card for HIL simulations, a fixed step size must be selected. Another approach to the simulation of event-based systems that yields a relatively small discrepancy between the simulated system and the actual (theoretical) one is to utilize a relatively short step size (fixed step size). Yet, the relative accuracy of the target processor could limit the effectiveness of the simulation and, in the case of complex event-based systems, it is possible that infinitesimally small step sizes be necessary to achieve successful simulations.

This paper presents the problem of accuracy involved in the conventional simulations, at fixed time steps, of event-based systems and then proposes the use of a new Simulink™ toolbox, known as the RT-Events toolbox, to significantly reduce the error between simulated and actual event-based systems. The paper is divided as follows. Section 2 presents the conventional techniques used in the fixed-step size simulations of the basic integrator block with reset performed in between the iteration steps. Among the issues discussed in Section 2, the adverse phenomenon known as reset walk is explained. Section 3 proposes a new set of tools in the form of a Simulink™ toolbox: the RT-Events toolbox. The important blocks of the RT-Events toolbox are succinctly explained. Three numerical examples illustrating the effectiveness of the new simulation tools are given in Section 4. Finally, the conclusions are provided in Section 5.

2 CONVENTIONAL SIMULATION TECHNIQUES

The basic process taking place in event-based dynamic systems simulation is signal integration. Knowing the limitations associated with the simulation of a reset integrator is helpful in the understanding of the intricacies involved in the simulation of the majority of event-based systems. In this section, some of the conventional fixed-step reset integrator simulation techniques are presented, followed by an explanation of the problem known as reset walk. Furthermore, at the end of the section, the limitations of the conventional simulation techniques to accurately simulate the latch operation, or the aperiodic sample-and-hold, are discussed.

2.1 Integrator Simulation

The most widely used techniques to simulate the integration of a continuous signal can be divided into two categories: (1) those using a scheme to approximate the integration based on the evaluation of n points at each iteration step, such as the n-point Runge-Kutta methods (Pachner 1984); and (2) those calculating a discrete-time transfer function to approximate the continuous-time transfer function of the integrator, i.e. $1/s$, such as, for instance, the Forward and Backward Euler methods and the Trapezoidal technique (Astrom and Wittenmark 1990). The latter methods are not only used in the field of simulation but also in the digital redesign of continuous-time control systems (Franklin et. al. 1994, Rabbath et. al. 1999) to convert continuous-time controllers to discrete-time equivalents.

On the one hand, when the reset of the integration takes effect at an integral multiple of the fixed step size, the actual reset event and that obtained with the fixed-step simulation of the integrator are occurring at the same time instant. On the other hand, when the reset event of the theoretical model takes place at a non-integral multiple of the fixed step size, the integrator simulated with any of the aforementioned techniques resets at a step following the occurrence of the event, not necessarily the first iteration step following the time instant at which the reset event took place, as discussed in Section 2.2.

Therefore, there are two sources of errors when simulating, with fixed steps, a reset integrator: (1) the numerical approximation to the actual integration; and (2) the occurrence of a reset event at non-integral multiples of the fixed step size.

2.2 Reset Walk Effect

Consider Figure 1, which presents a model for the simulation of the integration of a continuous signal with reset based on a comparison between the integrator state and an external signal. For such a system, there is an adverse phenomenon that arises in simulations using the methods described in Section 2.1 when the reset events occur in between the sampling instants. This phenomenon is called reset walk and is explained as follows. Suppose that several reset events take place over a given time period, assuming that a maximum of one event arises between successive iteration steps. Moreover, assume that the integrator is approximated with any of the fixed-step techniques discussed in Section 2.1. Figure 2 shows the outputs of the actual (or theoretical) and the simulated integrators in the case of the integration of a constant signal with reset occurring every time the state of the integrator reaches a value superior or equal to π. In Figure 2, the reset events are denoted as $t_{e,i}$, where i represents the ith occurrence of a reset event, and the step size, denoted as T_s in this paper, is chosen to be much shorter than the time interval between successive events. From Figure 2, it is seen that there is a relative movement between the time instants at which the resets of the fixed-step simulated integrators take place and the instants at which the reset events of the actual integrator occur. To understand the difference in reset event occurrences, between the actual and the simulated models, and its increase with time, one has to keep in mind that the reset is based on the feedback of the state of the integrator, as shown in Figure 1. As the simulation moves forward in time, the difference in reset times grows since the error in integrator outputs increases. The consequence of the cumulative effect of the integration error is that the reset value of the state of the integrator (π in Figure 2) is reached at later time instants with the simulated integrators than with the actual integrator.

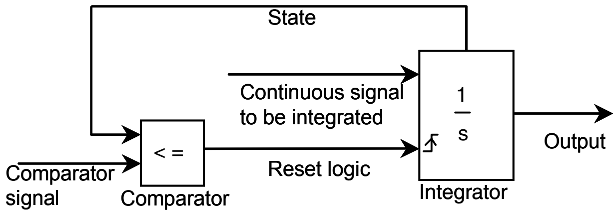


Figure 1: Model of a Reset Integrator

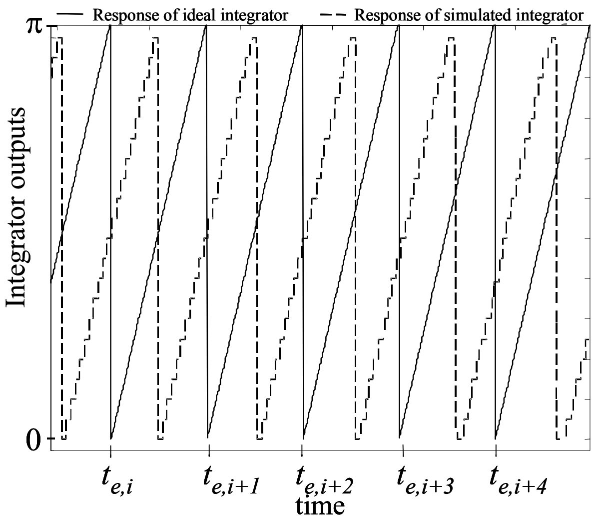


Figure 2: Integrator Outputs

2.3 Latch of a Signal

The latch operation can be modeled as a block that samples, and then holds, the value of an input signal at time instants for which a condition is satisfied. In short, the latch acts like a sample-and-hold operator except that, instead of performing the sampling operation periodically at fixed time instants, the sampling operation is carried out at discrete events. Figure 3 presents the latch operation, where *HS* samples the input signal at time instants determined by the value of the logic signal and then holds it until the next event.

When simulating the latch operation with a fixed step size, it is possible that the simulation omits the time instants at which the actual latch is triggered. Of course, in a fixed-step simulation, the input signal is sampled, and then held, at the first step following the triggering of the actual latch. However, the discrepancy between the latch output of the theoretical model and that of the simulated model is always present.

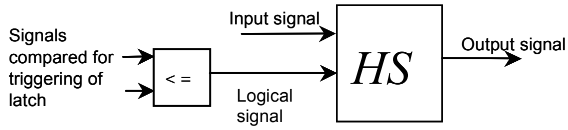


Figure 3: Model of a Latch

3 SIMULATIONS WITH RT-EVENTS TOOLBOX

In order to solve the problems inherent to the conventional fixed-step simulations of event-based systems, as discussed in Section 2, Opal-RT Technologies has created the RT-Events toolbox. The RT-Events toolbox comprises a set of discrete-time blocks that use a compensated discrete-time simulation algorithm, that are implemented within the Simulink™ environment and that execute with fixed steps; therefore, the RT-Events blocks are compatible with the Real Time Workshop™ and RT-LAB™.

The RT-Events library comprises a comparator block, a discrete-time integrator, a latch and several logical operators. The comparator block generates a trigger signal whose value depends on the result of a comparison between the two inputs to the block. The discrete-time integrator block uses numerical methods to approximate the integration of its input signal and comprises an algorithm that compensates for the occurrence of discrete events in between sampling instants. The key feature of the latch block is that it incorporates a method that compensates for the triggering taking place between sampling instants. Finally, the logical operators perform standard logical operations on signals while compensating for transitions occurring between the sampling times.

The major features of the RT-Events toolbox are summarized as follows:

- It compensates for the errors introduced by events occurring in between the simulation steps. Still, the simulation accuracy depends on the step size selected.
- It allows fast simulations of event-based systems.
- It is suitable for hard real-time applications.
- It is easily adaptable for distributed real-time simulations as obtained within the RT-LAB™ environment.

3.1 Key Concept

The key idea behind the RT-Events toolbox is that of including an algorithm that compensates for events taking place in between simulation steps (in the case of discrete-time systems, the simulation step size is assumed to be equal to the sampling period). For instance, in the case of an integrator with reset, at the sampling instants following the occurrence of an event, the discrete-time integrator does not reset its output to zero (as is the case with the conventional techniques), but rather it fixes its output to a value very close to that of the actual integrator; a value that is obtained with the compensatory algorithm. This provides a more accurate simulation with respect to the theoretical system.

As another example, the output of the latch is not taken to be the value of the input signal at the simulation step following the triggering of the latch (as is done with the current simulation tools), but instead this value is compensated to account for the event not taking place at the simulation step. This technique thus results in more accurate fixed-step simulations of the latch operation.

3.2 Solution to Reset Walk Effect

Using the integrator and comparator blocks of the RT-Events library solves the reset walk problem mentioned in Section 2.2. This is so since, as opposed to the conventional integrator simulation techniques, the RT-Events integrator guarantees a compensation for the reset event at the sampling instants following the occurrence of an event. Thus, the error between outputs of the RT-Events-simulated and the actual reset integrators cannot increase with time as it does with the classical simulation techniques.

4 NUMERICAL EXAMPLES

This section provides three examples that clearly illustrate the effectiveness of the new simulation tools. The first two examples are simple systems showing the compensated integrator and latch blocks, respectively, whereas the third example comes from the set of demos of Simulink™ and involves several blocks of the RT-Events library.

Sections 4.1 and 4.2 describe the simulations of the integrator and latch systems. These systems are simulated with fixed step sizes within the Simulink™ environment on a Windows-NT, Pentium II microprocessor. Section 4.3 presents a closed-loop engine system. This system is simulated in real-time on a QNX-Pentium II microprocessor within the RT-LAB™ environment. For all three examples, the variable-step simulations carried out on the Simulink™ software serve as the yardstick against which the fixed-step simulations are compared. In this section, the models executing at variable steps are referred to as theoretical models. Please note that relatively small tolerances are set for the variable-step simulations.

4.1 Integrator with Reset

This example is modeled as shown in Figure 1 for the variable-step and conventional fixed-step simulations. The continuous signal to be integrated is a unity constant and the comparator signal is chosen to be a constant value of 0.505. The fixed step size is selected as $T_s = 0.01$. With the fixed step size and the comparator signal values selected as such, the reset events take place in between integral multiples of T_s. The setup for the RT-Events-based simulation is shown in Figure 4. The integrator outputs are shown in Figure 5 for the beginning and the end of the simulations, so that it is easier to visualize the various signals. The reset walk is apparent for the conventional fixed-step simulation techniques (five-point Runge-Kutta, Forward and Backward Euler, and trapezoidal methods), whereas no such effect can be seen for the simulation performed with the RT-Events blocks. In fact, the augmentation of the integration error with time, as obtained with the conventional fixed-step simulation techniques, is easily seen by comparing Figures 5(a) and 5(b).

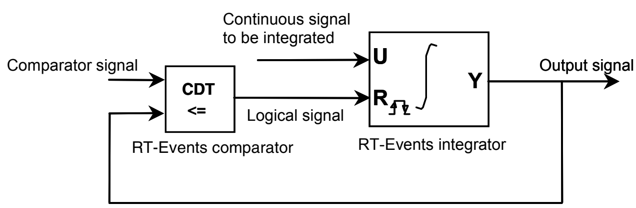
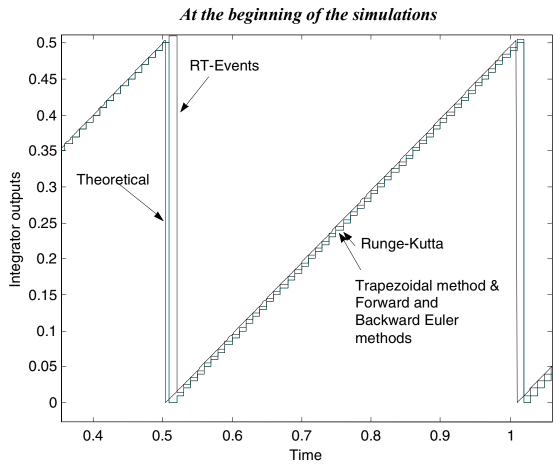


Figure 4: RT-Events-based Model of Integration

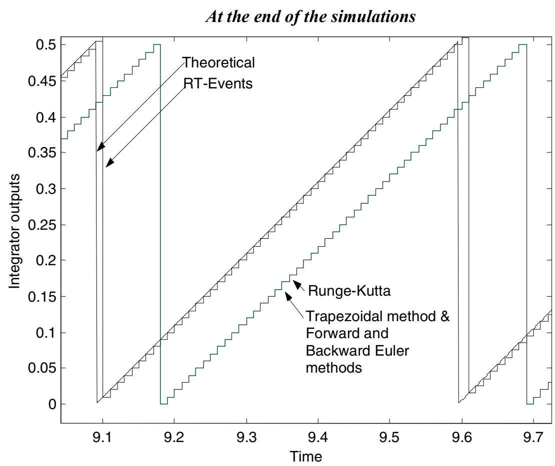
This example clearly favors the RT-Events toolbox over the so-called native Simulink™ blocks since no accumulation of error is present with the RT-Events -based simulations.

4.2 Latch of a Continuous Signal

Figure 6 shows the setup of the simulation for the RT-Events-based model. The latch obtained with the Simulink™ library is the triggered feed through block and is not shown for brevity. The input to the latch is a unit ramp signal. The latch takes effect at the time the input reaches a value of 1.675. The theoretical output of the latch is thus a constant signal of value 1.675 starting at time t = 1.675. The fixed-step simulations are carried out with T_s = 0.01. The outputs of the simulated and the theoretical latches are shown in Figure 7. The compensated latch (RT-Events-based model output in the figure) shows a superior behavior than that achieved with the conventional fixed-step simulations. The time delay between the actual latch event occurrence and the latch of the fixed-step simulations is expected since the actual latch event occurs at 1.675, which is between the fixed simulation steps 1.67 ($167 \cdot T_s$) and 1.68 ($168 \cdot T_s$).



(a)



(b)

Figure 5: Integrator Outputs (a) at the Beginning and (b) at the End of the Simulations

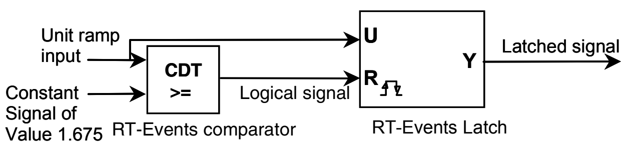


Figure 6: Schematics of the Latch System

4.3 Closed-Loop Engine System

Four schematic block diagrams of the RT-Events-based engine model are shown in Figures 8, 9, 10 and 11. In the figures, the shaded blocks are those of the RT-Events library. The engine diagrams representing the other fixed-step and the variable-step simulated models are not shown since they are similar to the RT-Events-based system, although the native Simulink integrator, triggered feed through, comparator and logical operators replace the RT-

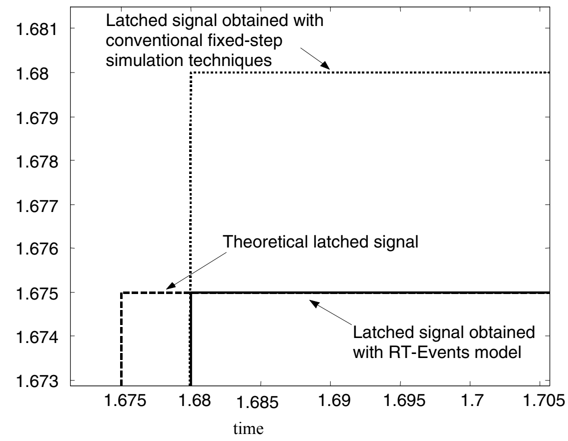


Figure 7: The Various Latched Signals

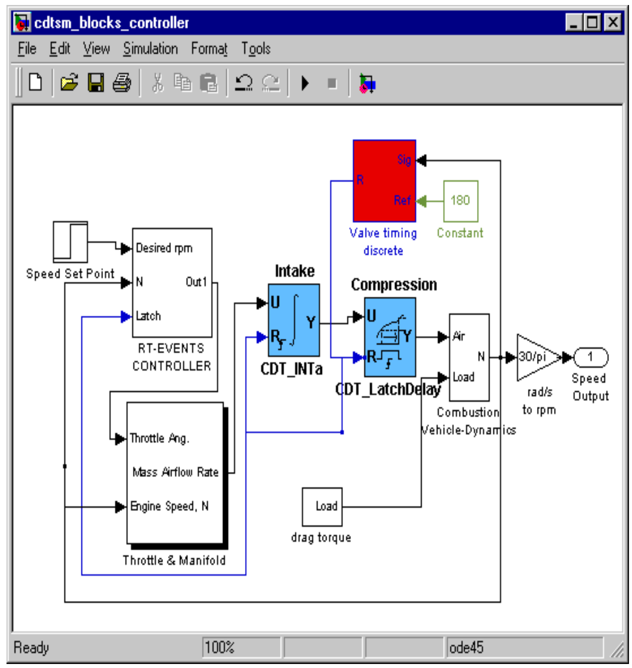


Figure 8: Top Level of Closed-Loop Engine Model Using RT-Events Blocks

Events integrator, latch, level detector and logical operators, respectively. The figures present several blocks of the RT-Events library. It should be noted that, associated to each RT-Events block, there is a mask with parameters that can be set by the simulation engineer; for instance, parameters such as reset edge, initial conditions, and sampling period can be adjusted on the mask of the integrator block.

In this example, an engine in closed-loop with a discrete-time controller is simulated. The events are related to the timing of the crankshaft angle of the engine with respect to the value of π. The discrete-time controller of each simulated model is executed at a sampling period of $T_s = 0.001$ second, and so are the approximations to the

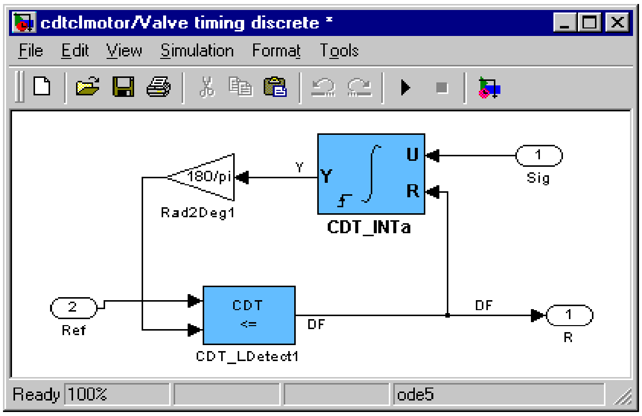


Figure 9: Valve Timing Subsystem of RT-Events Model

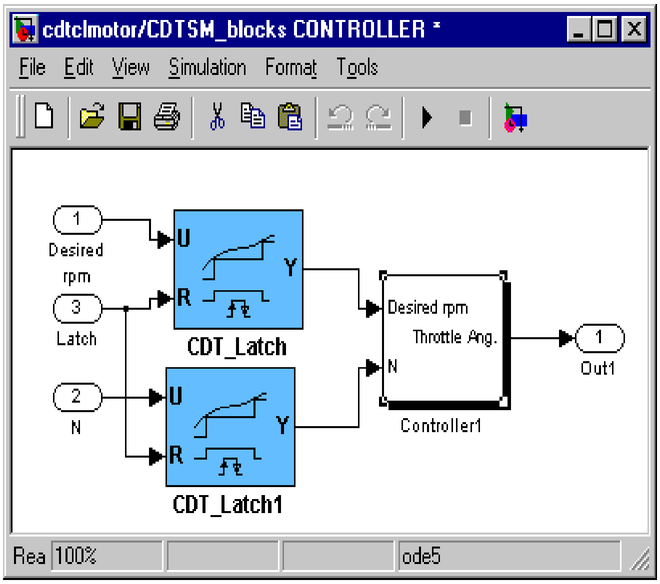


Figure 10: Inputs to Controller Subsystem of RT-Events Model

continuous-time integrators found in the engine system. The signals used in the comparisons are the mass airflow and the engine speed (in units of rpm).

The simulation results are shown in Figure 12. In Figure 12(a), the mass airflow of the RT-Events-based simulation, i.e. the output of the reset integrator associated with the intake dynamics according to Figure 8, is close to that obtained with the theoretical model whereas the mass airflow of the native Simulink™ fixed-step simulation is completely different from that of the theoretical model. In fact, a reset walk takes place with the native Simulink™ fixed-step simulation. It should be mentioned that only a portion of the entire simulation duration is presented in Figure 12(a) in order to clearly show the detrimental reset walk effect occurring with the native Simulink™ fixed-step simulation.

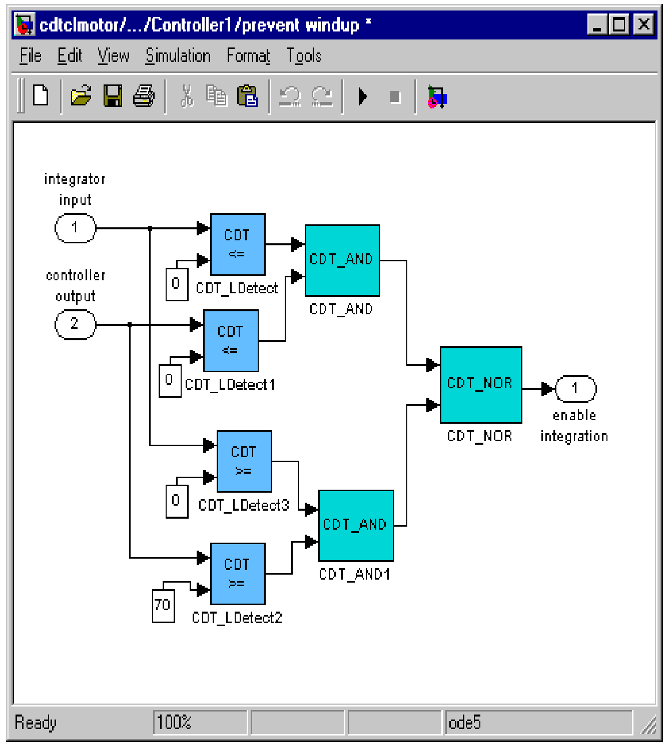


Figure 11: Anti-windup of Controller Subsystem of RT-Events Model

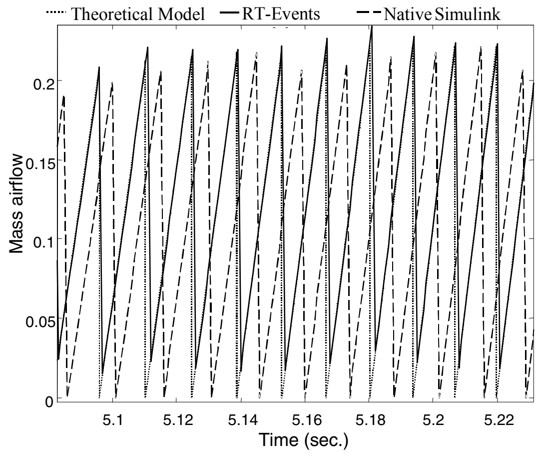
In Figure 12(b), the time trajectory of the engine speed obtained with the model using the RT-Events blocks is almost indistinguishable from that of the variable-step simulation using the native Simulink™ blocks (i.e. the theoretical model). This is not the case with the outputs of the conventional fixed-step simulations (i.e. the native Simulink™ model executing at fixed steps).

It should be noted that the variable-step simulation executed more slowly than its fixed-step counterparts. This was so because of the stringent tolerances specified at the outset of the simulation.

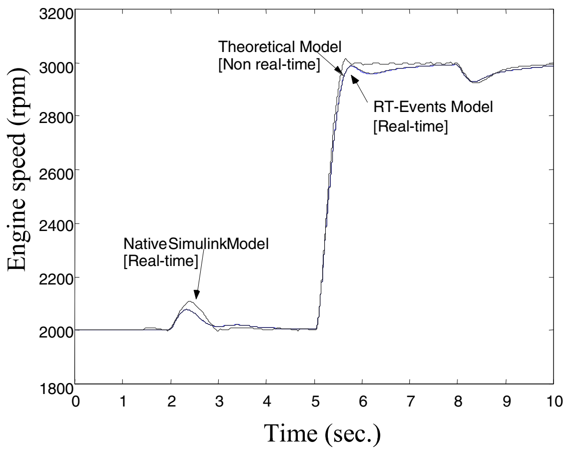
Clearly, the real-time simulation of the closed-loop model utilizing the RT-Events blocks, where appropriate, shows a superior time-domain performance than that obtained with the real-time simulation of the engine model comprising only the native Simulink™ blocks.

5 CONCLUSIONS

It was shown that the problems inherent to the techniques currently available to simulate event-based systems, namely the time-consuming variable-step algorithms and the inaccurate conventional fixed-step algorithms, can be solved by using the novel RT-Events toolbox. The key concept behind the new simulation tools is one of compensation for the events taking place in between the sampling instants. The potential applications of this new toolbox are clear. To name one, the Ford Motor Company is now



(a)



(b)

Figure 12: (a) Mass Airflow and (b) Engine Speed vs. Time

investigating the use of the RT-Events toolbox to effectively simulate internal combustion engines.

ACKNOWLEDGMENTS

The authors would like to thank Vincent Lapointe and Simon Abourida for their significant contribution to the development of the RT-Events toolbox.

REFERENCES

Astrom, K.J. and B. Wittenmark. 1990. *Computer-Controlled Systems: Theory and Design*. NJ: Prentice Hall.

Cassandras, C. 1993. *Discrete Event Systems: Modeling and Performance Analysis.* Burr Ridge, IL: R.D. Irwin, Inc. and Aksen Assoc., Inc.

Franklin, G.F., J.D. Powell and A. Emami-Naeini. 1994. *Feedback Control of Dynamic Systems*. Reading, MA: Addison-Wesley.

Pachner, J. 1984. *Handbook of Numerical Analysis Applications*. NY: McGraw-Hill.

Rabbath, C.A., N. Hori, P.N. Nikiforuk and K. Kanai. 1999. Order Reduction of PIM-Based Digital Flight Control Systems. In *Proceedings of the IFAC World Congress*, Vol. Q, 127-132.

Rabbath, C.A. and M. Abdoune. 1999. The use of the CDTSM blocks for Engine Simulations. Opal-RT Technologies, Engineering Report CDTSMV211, Montreal, Canada.

AUTHOR BIOGRAPHIES

C. A. RABBATH is a Control Systems Specialist and Consultant for Opal-RT Technologies, and an Adjunct Professor in the Department of Mechanical Engineering at McGill University. He received the B. Eng., M. Eng., and Ph.D. degrees from McGill University. His current research interests include engine simulations, sampled-data control techniques, and multi-rate digital control. His email address is `<alain.rabbath@opal-rt.com>`.

M. ABDOUNE is a Control Systems Specialist and Consultant for Opal-RT Technologies as well as a certified nuclear engineering specialist. He received the M. Eng. Degree from Ecole de Technologie Superieure in the field of control systems. He is a member of the order of professional engineers of the province of Quebec. His interests include optimal and flight control. His email address is `<moussa.abdoune@opal-rt.com>`.

J. BELANGER is president of Opal-RT Technologies, Inc. He is a specialist in real-time simulations, with more than 20 years of experience in the field, including many years as part of the simulation division of Hydro-Quebec. He received the MScA degree from Laval University and is a member of the order of professional engineers of the province of Quebec. His email address is `<jean.belanger@opal-rt.com>`.