Developments and Applications of Multi-rate Simulation

J. G. Pearce
ISIM International Simulation Limited, UK
johnpearce@isimsimulation.com

R. E. Crosbie, J. J. Zenor, R. Bednar, D. Word
California State University, Chico, USA
rcrosbie,jzenor,rbednar,dword @csuchico.edu

N. G. Hingorani
Independent Consultant, Los Altos Hills CA , USA
nghingorani@sbcglobal.net

Abstract

Multi-rate simulation techniques offer advantages to the computer simulation of large scale dynamic systems. Each part of the system is solved using the most appropriate time step and numerical integration method. The approach can be particularly advantageous in real-time applications, where it is essential to complete each set of calculations in the allotted real-time interval. The ESL Simulation language has parallel segment features which makes it particularly suited to the realization of multi-rate simulations.

Experiences of using ESL and the Virtual Test Bed (VTB) to realize a multi-rate simulation of an underwater unmanned vehicle (UUV) are described. A non-real-time version of the simulation with 5 different frame rates has achieved speed increases of the order of 500 times with no significant loss of accuracy, making a real-time implementation feasible.

This research has included analysis of the stability of multi-rate methods and these are summarized in the paper.

1. Introduction

Many large dynamic systems consist of clearly identifiable modules which have very different associated time constants. An example could be an electrically propelled ship: the time constants associated with the electronic power control systems may be of the order of microseconds; the mechanical dynamics of the motors of the order of milliseconds whilst the dynamics of the complete vessel moving in the sea of the order of seconds or more. To simulate the complete system as a single program would dictate integration algorithm step-lengths consistent with the shortest time constants and hence result in excessive solution times. With multi-rate simulation the complete system is first partitioned into modules corresponding to the different time constants. Techniques then allow each partition of the system to be solved using the most appropriate step-lengths and integration algorithms with data being communicated between the partitions at fixed communication points.

This paper presents the results of the application of multi-rate simulation techniques to a real-time simulation of an underwater unmanned vehicle (UUV). The system consists of three main subsystems, the power supply consisting of a battery and inverter with switch controller, the electric drive, and the vessel. This system has been simulated using up to five different frame times. Execution speed has been increased several hundred times without significant loss of accuracy compared to a simulation in which all parts of the system are simulated with the fastest time step.

This project demonstrates several features of modern, multi-disciplinary simulation problems in which it is increasingly common for different parts of the simulation to be developed by different disciplinary groups using different programming approaches to produce a multi-rate and, in this case, real-time simulation. These include proper choice of integration algorithm and step-length for different multi-rate segments, careful integration of components of the simulation into a unified simulation, and avoiding problems due to inappropriate sampling intervals.

Two software products proved helpful in realizing the simulation. The Virtual Test Bed [4] provided an environment for execution of the simulation in which provided some useful native models (battery and motor), and both graph plots and 3-D animation of system behavior. The ESL language, with its built-in multi-rate capability, proved to be a valuable aid in the development of the simulation.

978-0-7695-3593-7/09 $25.00 © 2009 IEEE
DOI 10.1109/UKSIM.2009.23

2. Multi-rate simulation

The basic principle of multi-rate simulation has been well known and frequently used for many years. It consists of using different integration time-steps and, in some circumstances, even different integration algorithms, for different subsystems in a larger system simulation. However, in order to ensure accurate and stable performance from a multi-rate simulation, it is important not only to partition the system and choose step-sizes with care, but also to manage the exchange of information between partitions carefully. It may be necessary to perform some form of averaging of outputs from the faster components to provide suitable inputs to the slower partitions, and it may be advisable to estimate intermediate values of the outputs from the slower partitions that are used as inputs to faster partitions. For example, as is well known from sampling theory, aliasing can occur if the output of a faster partition contains frequencies that are greater than twice the frame rate of a slower partition to which it is used as an input. This problem can be avoided by an appropriate choice of partition boundaries so as to avoid the appearance of the high-frequency components in the critical output or alternatively by introducing anti-aliasing filters into the model.

Multi-rate methods can be particularly useful in support of real-time simulation in which it is necessary to complete necessary computations within the allotted frame time. This is particularly true of real-time simulations with components that cover a wide dynamic range. In these real-time simulations some subsystems will require very short frame times for accuracy, but the other, slower components will not in general be capable of completing all computations in the same short time step. Multi-rate simulation in such situations becomes a very appealing option. An example of such a real-time simulation is described in Section 4 below.

3. The ESL simulation language

ESL [1,2,3] is an advanced high-level simulation language for modelling large-scale systems from a variety of disciplines. ESL comprises two components: the language itself and a graphical user interface - the Integrated Simulation Environment (ISE). ESL is a continuous system simulation language and is used for modelling dynamic systems which are usually described by ordinary and partial differential equations. ISE provides the environment from which all stages of the simulation process can be managed. The software was developed mainly through a series of contracts with ESTEC - the European Space Research and Technology Centre - part of ESA with additional support from various industrial simulation consultancy activities.

3.1 Segments in ESL

A feature of ESL that lends itself to the realization of multi-rate simulation is its *segment* structures. Segments were originally included in ESL as a means of providing a parallel processing capability. The idea was that a large simulation could be broken down into self-contained segments that could be executed in parallel on different processors or networked computers. Communication takes place between segments at pre-determined communication points through a TCP IP protocol. There are three types of segment in ESL:

- *Remote segments* – these can be assigned to different processors for truly parallel operation.
- *Emulated segments* – these allow parallel operation to be emulated on a single computer – useful for testing parallel segments before assignment to separate processors and for implementing multi-rate simulations.
- *Embedded segments* – used where an ESL model is to be integrated with another application.

The use of the emulated segment feature of ESL to implement multi-rate models was evaluated in a non-real-time simulation of the UUV (section 4.1). The embedded segment feature provides a means of interfacing ESL models to the VTB (section 6).

4. The UUV simulation

As a means of investigating multi-rate simulation techniques including real-time multi-rate simulation, a benchmark was developed based on an unmanned underwater vehicle (UUV). The UUV system is illustrated in Figure 1. The d.c. power source simulation is taken from the VTB simulation library. The discharge characteristics of each battery cell are simulated, and cells are connected in a series-parallel configuration to achieve the desired voltage and capacity.

The d.c. power source is connected to the six-pulse, three-phase d.c. to a.c. PWM converter. The converter contains a six-switch network that produces a synthesized, three-phase, variable-frequency a.c. waveform. The a.c. output from the converter is filtered to remove high-frequency harmonics and is then passed to the induction motor via a natural coupling.

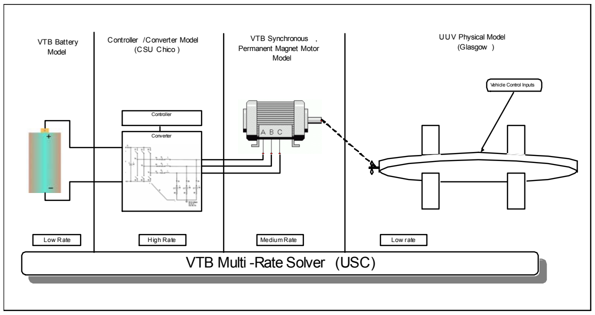

Figure 1: Simplified representation of UUV model
(Details of vessel I/O variables and controls are omitted)

The converter switches are operated by on-off commands from the PWM switch controller. A feedback controller is used to control the frequency of the converter output, which ultimately controls the propeller RPM of the UUV platform. The controller also has the capability to use a PI scheme to control output current or power supplied to the motor drive. The controller determines switch timing by the comparison of sine waves at the output frequency and five-kilohertz triangular waves. Their relative amplitudes are adjusted by the feedback control and switching occurs when the sine and triangular waves intersect. The phase of the a.c. output may also be controlled.

The converter simulation is implemented in C++ as a native VTB model (see section 6). The d.c. input connection and the a.c. output connections are natural couplings. The connections to the controllers are signal couplings. Controller models contain only logical elements and do not have any differential equations. The converter simulation uses trapezoidal integration for the filter components, and Euler integration for the input d.c. filter capacitor. The sine and triangular waveforms are generated by table lookup, scanning the tables at a rate determined by the desired frequency. Linear interpolation is used, with enough table entries to have a maximum of a one-bit error after interpolation.

The converter is used to supply a.c. power to the induction motor. The motor simulation is implemented in C++ as a native VTB simulation, and uses trapezoidal integration. The motor applies torque to the propeller shaft and rotates the propeller at a speed determined by the input a.c. frequency, driving the UUV forward. The mechanical connection between the motor simulation and the ship simulation is a signal connection. The motor model outputs an RPM and the combined ship and propeller model returns the torque on the shaft. The input a.c. power to the motor uses a natural connection.

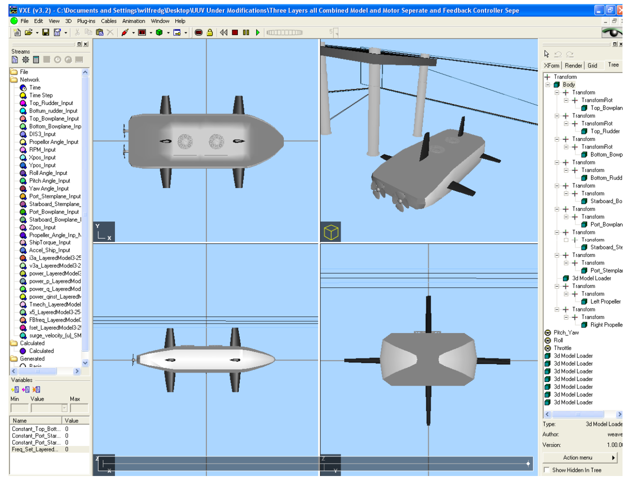

Figure 2: VXE Screen-Shot of UUV Simulation

The ship has a six degree of freedom simulation, originally implemented in Matlab and uses fourth-order Runge Kutta integration. The fin deflections are input via the user interface and propeller shaft RPM as a signal input. The Matlab simulation was translated to C++ and implemented as a native VTB model, although the original model may still be used via the VTB/Matlab interface. Typical simulation frame rates used in this simulation are:

Battery, ship, 3D graphics: 100 mS
Converter, switch controller: 2 µS
Feedback controller: .8 mS
Motor: 100 µS

A screen-shot of the output provided by the VXE (Virtual Extension Engine), which provides graphical support to the VTB, is shown in Figure 2.

This simulation is not synchronized to real-time but the multi-rate implementation runs approximately 2 times faster than real time on a modern laptop. A real-time implementation is under development in which the high-speed components (converter, switch controller) will run on an FPGA and the rest of the simulation on a real-time Linux platform. The high-speed subsystem has already been implemented on a Xilinx Virtex 5 FPGA and is capable of executing with frame times as short as 400 nS..

4.1 UUV simulation using ESL

An ESL version of the UUV simulation was developed to test the ESL segment feature. This used three segments; for the vessel with a fixed frame time of 0.1 second; for the motor with a fixed frame time of 100 µS; and for the converter and controller with a frame time that was varied from 2 µs to 10 µs. The results of the simulation were in good agreement with other versions of the program. One advantage of using ESL in this study was that it was possible to compare runs in which the ESL discontinuity detection and correction feature (using a variable-step algorithm) was first disabled and then enabled. This approach provides a good guide to the magnitude of switch timing errors that occur in real-time when fixed-step integration algorithms must be used. These results clearly confirmed the need to use frame times of no more than 2 µs for switch controllers with a PWM frequency of 5 KHz as in this simulation. When the frame time for the converter-controller segment was increased from 2 µs the difference between the results obtained with discontinuity detection enabled and disabled grew rapidly. At 2 µs these differences were insignificant.

5. Stability analysis of multi-rate methods

The research performed so far has been limited to extending analysis of single-rate simulations to bi-rate simulation. This was done as a starting point for a more general multi-rate analysis. However, analysis of simulations using more than two rates is much more difficult than the bi-rate case and no satisfactory method of completing it has so far been identified.

The bi-rate analysis that has been completed so far has included analysis of the Euler, state transition (3,2 method), and implicit trapezoidal methods [6]. The effect of different methods of data transfer (zero-, first- and fractional-order hold) with different integration algorithms has also been studied. The results of these analysis have confirmed that, for the piecewise continuous linear systems that form the basis of the power electronic models on which this research is focused, the state-transition 3,2 method offers the best performance in terms of accuracy, stability, and computational cost. The bi-rate stability analysis provides a good basis for determining the ranges of parameter values that can be relied upon to guarantee stable simulations. This can guide the choice of algorithm. The state transition approach provides a class of methods of which (3,2) is one of the simplest while providing excellent performance in most cases. When (3,2) proves unsatisfactory a more complex method involving more terms can be a better choice. It is a feature of the state transition approach that the complete infinite series solution is always stable if the original differential equations are stable. There is, therefore, a trade-off between stability and speed that can be exploited in difficult cases.

6. Virtual Test Bed (VTB)–ESL interface

The Virtual Test Bed (VTB) [4] is a software environment for developing simulations of large scale multidisciplinary dynamic systems. It allows alternative designs to be analysed and tested before being committed to manufacture. The main application that is driving the development of the VTB is a need to model advanced power systems for navy ships. In such systems there are many different energy generation and storage devices including nuclear, fuel cells and gas turbines. The distribution networks are also of unconventional design having dc power busses and high numbers of interconnections that can be rapidly reconfigured. Constructing complete coherent simulations of such large scale systems, involving widely differing technologies poses a serious challenge. Each discipline group will use their preferred simulation tool to model their part of the whole system. The VTB aims to satisfy this challenge by providing a common platform in which component models developed by different teams using different tools can be merged. The professional version of VTB (VTB Pro) uses a companion program, Eye-Sys, to realize three dimensional visualizations of a VTB simulation.

ESL provides a utility for generating a .NET assembly from an *Embedded segment* which can then be imported into a VTB Pro simulation using a special VTB *ESL Importer tool* [5] (See Figure 3). Using these tools, the multi-rate ESL UUV simulation can be run in the VTB. This will enable the simulation to be run under the full power of VTB Pro and provide a 3D visualization.

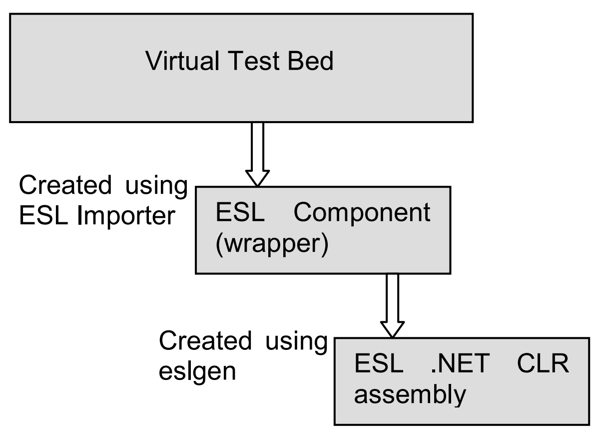

Figure 3 VTB - ESL Interface

7. Conclusion

Multi-rate simulation is a technique that has been used extensively for many years. In recent years its ability to speed up simulation execution times has tended to become less important as the rapid increases in computing power have outstripped in many fields the complexity of the systems being simulated. The challenges posed by some of today's simulation problems have revived interest in multi-rate methods. It is a technique that is well worth revisiting to provide cost-effective solutions to these challenging problems. This is particularly true where the requirement is for a real-time simulation with a wide dynamic range.

8. Acknowledgements

The authors wish to express their appreciation of funding provided by the US Office of Naval Research in support of this project.

They also acknowledge the contribution of Chico graduate student Tian He for his work on the ESL-based simulation of the UUV system.

9. References

[1] Crosbie, R.E., Hay, J.L. and Pearce, J.G., "Simulation Studies with Modern Computer Structures", *Final Report, (ESTEC Contract 4155/79)*, ESTEC, Noordwijk The Netherlands, 1981.

[2] Hay, J.L., Pearce, J.G., Crosbie, R.E. and Pallett, S., "ESL Simulation Tool", *Final Report, (ESTEC Contract 10011/92/NL/JG Work Order No. 2)*, ESTEC, Noordwijk The Netherlands, 1994.

[3] Pearce, J.G. and Crosbie, R.E., "ESL-ISE - A Simulation Tool Developed for the Space Industry", *Proceedings of the 2000 International Conference on Simulation and Multimedia in Engineering Education*, (San Diego, CA, Jan 23-27). SCS, San Diego, CA, 2000, pp 115-120.

[4] Dougal, R.A., Langland, B. and Monti, A. "Virtual Prototyping as a Mechanism for Simulation-based Design", *Proceedings of Summer Simulation Multiconference,*SCS, San Diego, CA, July 2007

[5] Pearce J.G. "Simulation advances using the ESL Simulation Language and the Virtual Test Bed", *Proceedings of Grand Challenges in Modeling and Simulation Conference*, SCS, Edinburgh UK, June 16-18, 2008, pp 285-290.

[6] Josef Stoer and Roland Bulirsch, *Introduction to Numerical Analysis*, Springer-Verlag, New York, 1980. *(See Chapter 3.)*