2010 14th IEEE/ACM Symposium on Distributed Simulation and Real-Time Applications

From DEVS to RTA-DEVS

Hesham Saadawi
hsaadawi@connect.carleton.ca

Gabriel Wainer
gwainer@sce.carleton.ca

Carleton University
Ottawa, ON, CANADA

***Abstract*— Rational Time-Advance DEVS (RTA-DEVS) is an extension to DEVS that enables formal verification of simulation models using standard model-checking algorithms and tools. In order to enable formal verification of DEVS models, we introduce a procedure to approximate DEVS with RTA-DEVS. We include conditions for valid approximation and a calculation method for approximation errors that may be introduced. The resulting RTA-DEVS models are behaviorally equivalent to the original DEVS.**

Keywords:* *Discrete event simulation, DEVS, UPPAAL, DEVS verification, Timed Automata, model checking, RTA-DEVS.

I. Introduction

Real-time embedded systems (RTS) are highly computer reactive systems where the decisions taken can lead to catastrophic consequences for goods or lives; hence, correctness, and the timing of the executing tasks are critical. This correctness must be verified, and Timed Automata (TA) [1] proved a established theory of formal verification and analysis through *model-checking* [2][3]. *DEVS* [5] provides a formal method to model and simulates discrete-event systems. We are interested in using DEVS for modeling RTS, and to model-check the specifications. However, DEVS does not yet have a sound theory for formal verification (thus, models are mainly studied through simulation).

We defined a subclass of DEVS called RTA-DEVS, that can be translated to equivalent TA models and formally verified [4]. RTA-DEVS removes many of the obstacles to formal verification, while retaining most of the expressive power of classic DEVS. We introduce here a methodology to abstract classic DEVS models to RTA-DEVS models, showing how to avoid modeling errors that may produce wrong RTA-DEVS models. We also show a method to estimate an upper bound of approximation errors that may be introduced during the abstraction process.

II. Background

DEVS was originally defined in the '70s as a discrete-event modeling specification mechanism derived from systems theory. A system modeled with DEVS is described as a hierarchical and modular composite of models, each of them being behavioral (atomic) or structural (coupled). [5]. A DEVS atomic model is formally described by:

$$M = < X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta >$$

Each model uses input (***X***) and output (***Y***) ports to communicate with others. External inputs trigger the external transition function $\boldsymbol{\delta_{ext}}$. The internal transition function $\boldsymbol{\delta_{int}}$ is activated after the lifetime **ta** of the present state has been consumed. $\boldsymbol{\delta_{int}}$ can lead to a state change. Results are spread through output ports by the output function (**λ**), which executes before the internal transition.

A DEVS coupled model is composed of several atomic or coupled sub models. They are formally defined as:

$$CM = < X, Y, D, \{M_i\}, \{I_i\}, \{Z_{ij}\} >$$

Coupled models are defined as a set (**D**) of interconnected components ($\mathbf{M_i}$ atomic or coupled). A coupled model uses input (***X***) and output (***Y***) ports to communicate with others. The translation function ($\mathbf{Z_{ij}}$) is in charge of converting the outputs of a model into inputs for the others. To do so, an index of influencees ($\mathbf{I_i}$) is created for each model. This index defines that the outputs of the model M_i are connected to inputs in the model M_j, where j is an element of I_i.

When trying to apply model-checking to classic DEVS, we are faced with a number of issues that make the problem undecidable (i.e., reachability analysis would not terminate). These difficulties are summarized as follows:

- A model could have an infinite number of total states
- The codomain of **ta** can be an irrational number.
- The elapsed time *e* in the definition of external transition function could be an irrational number.

RTA-DEVS [4] has changed the atomic model definitions to avoid this problem. Now, **ta** has a codomain in the positive rational numbers, and its $\boldsymbol{\delta_{ext}}$ uses rational positive constants only for the elapsed time *e*. These changes enable the transformation of RTA-DEVS to equivalent TA models that are verifiable through model checking algorithms.

III. Model-checking DEVS

To be able to model check a system with infinite number of states these would need to be over-approximated to a finite number of states. By converting DEVS models to RTA-DEVS we remove the problems discussed above. To do so, we need to find a reasonable approximation for any irrational values that may exist in the DEVS model, while building a valid RTA-DEVS.

A. Irrational values in Time advance function

DEVS irrational values for the **ta** need to be approximated when converting to RTA-DEVS. An example of such DEVS model is given in Figure 1(a). The equivalent transformed RTA-DEVS and TA models are shown in (b) and (c) respectively. In this figure, a DEVS model is approximated with RTA-DEVS model as shown in (b).

1550-6525/10 $26.00 © 2010 IEEE
DOI 10.1109/DS-RT.2010.31

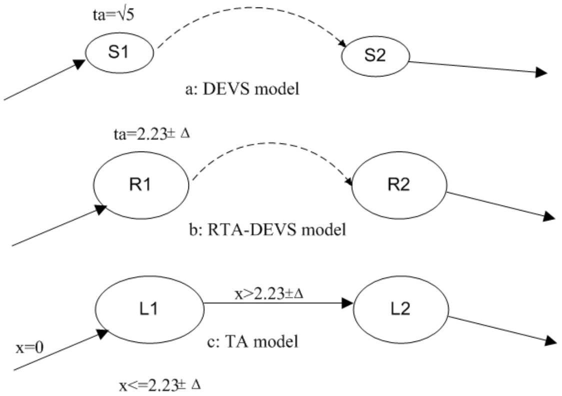

Figure 1: Approximation of Irrational time values. Internal Transition. a) DEVS b) RTA-DEVS c) TA.

The irrational time advance value is converted to a rational value with approximation error Δ. This error propagates on the equivalent TA. The questions here are: how is this approximation error Δ going to affect the validity of RTA-DEVS and TA models? Are these valid models? Do we have the same conclusions on the original DEVS and the TA?

To answer the first question (i.e. to guarantee building valid RTA-DEVS and TA models), we show a piece of a coupled DEVS model in Figure 2. Component A waits in state S1 for √5 time units, then executes the internal transition function which sends event *a,* and then goes to state S2.

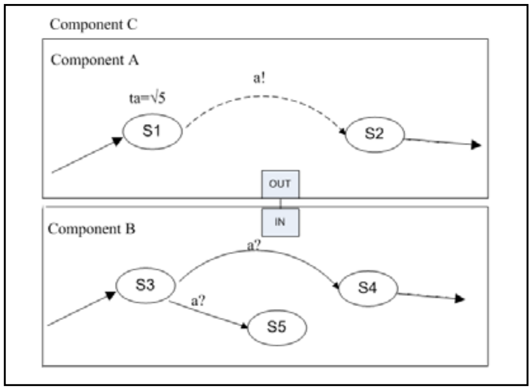

Figure 2: A coupled DEVS model

Component B is in state S3 waiting for event *a*, which triggers the external transition function as follows:

$$\delta_{ext}(S3,e,a) = (S4,0) \qquad \sqrt{5} \le e \prec \infty$$

$$\delta_{ext}(S3,e,a) = (S5,0) \qquad 0 \prec e \prec \sqrt{5}$$

By coupling components A and B together, the total behavior of the coupled DEVS component C would be:

$$(\text{S1,S3}) \xrightarrow{d=\sqrt{5},a} (\text{S2,S4})$$

The coupled system starts in total state of (S1,S3), and after a delay of √5 time units, A sends event *a* to B which triggers a transition to total state of (S2,S4). We then construct a behaviorally equivalent, approximated RTA-DEVS model (Figure 3) to the DEVS shown in Figure 2.

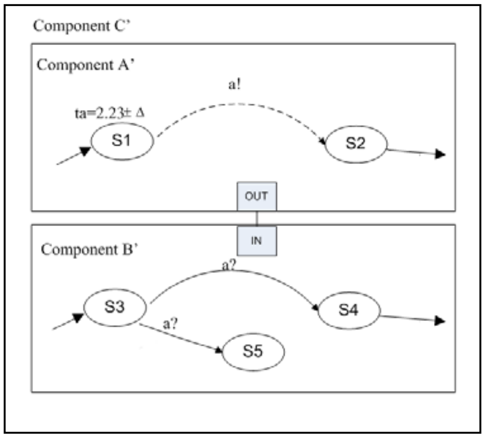

Figure 3: Coupled RTA-DEVS model

In this model, the lifetime of S1 was approximated by a rational value with error Δ. The value of Δ depends on the precision chosen; e.g., for 2 decimal digits, Δ ≤ 0.005. The external transition function would be approximated as:

Approximation 1:

$$\delta_{ext}(S3,e,a) = (S4,0) \qquad 2.23 + \Delta \le e \prec \infty$$

$$\delta_{ext}(S3,e,a) = (S5,0) \qquad 0 \prec e \prec 2.23 + \Delta$$

Or Approximation 2:

$$\delta_{ext}(S3,e,a) = (S4,0) \qquad 2.23 - \Delta \le e \prec \infty$$

$$\delta_{ext}(S3,e,a) = (S5,0) \qquad 0 \prec e \prec 2.23 - \Delta$$

However, the choice of the approximation would affect the validity of the RTA-DEVS model. For instance, if we approximate the **ta** of S1 with **ta** =2.23-Δ, and we choose Approximation 1 for model B, the coupled model C' would have a different behavior from the original DEVS model. Thus, component C' behaviour now becomes:

$$(\text{S1,S3}) \xrightarrow{d=2.23-\Delta,a} (\text{S2,S5})$$

Proposition 1: When approximating an irrational value triggering an internal transition that is coupled with an external transition, the choice of approximation value should be consistent for all constants using this irrational number.

Formally: if we have the following defined in DEVS:

$$\delta^{A}{}_{\text{int}}(S_i, C_{irr}) = S_j \;,\; \lambda^{A}(S_i) = a \;,\; ta^{A}(S_i) = C_{irr}$$

$$\delta^{B}{}_{ext}(S_k, e, a) = (S_l, 0) \qquad C_{irr} \le e \prec \infty$$

$$\delta^{B}{}_{ext}(S_k, e, a) = (S_m, 0) \qquad 0 \prec e \prec C_{irr}$$

It should be approximated in RTA-DEVS as:

$$\delta^{A}{}_{\text{int}}(S_i, C_r) = S_j \;,\; \lambda^{A}(S_i) = a \;,\; ta^{A}(S_i) = C_r$$

$$\delta^{B}{}_{ext}(S_k, e, a) = (S_l, 0) \qquad C_r \le e \prec \infty$$

$$\delta^{B}{}_{ext}(S_k, e, a) = (S_m, 0) \qquad 0 \prec e \prec C_r$$

Where:

C_{irr} : is an irrational real number.

C_r : is a rational real number.

$\delta^A{}_{\text{int}}$, λ^A , ta^A : Functions defined for component A.

B. *Irrational values in External Transition function*

For the example shown in Figure 4, a modeller may choose different approximations to form component B''

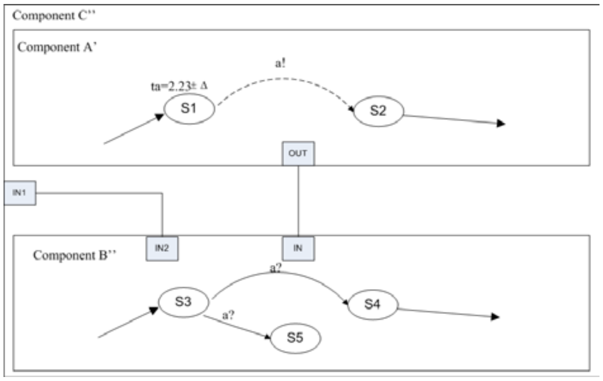

Figure 4: RTA-DEVS component with External Input.

Approximation 3:

$\delta_{ext}(S3, e, a) = (S4, 0) \qquad 2.23 + \Delta \le e \prec \infty$

$\delta_{ext}(S3, e, a) = (S5, 0) \qquad 0 \prec e \prec 2.23 - \Delta$

Or Approximation 4:

$\delta_{ext}(S3, e, a) = (S4, 0) \qquad 2.23 + \Delta \le e \prec \infty$

$\delta_{ext}(S3, e, a) = (S5, 0) \qquad 0 \prec e \prec 2.23 - \Delta$

Component C'' in Figure 4 accepts an external input from its environment on input port IN1 and this is connected to go into component B'' at its port IN2. In this case, the behaviour of C'' would not match the behaviour of C, in that the external transition function is not defined in the interval $2.23 - \Delta \prec e \prec 2.23 + \Delta$. Thus, C'' would contain an action-lock [6] which is a special case of a deadlock in which the system would not progress due to lack of any enabled transitions at the exact point-in-time in which an event occurs. This typically reflects a modelling error (or, in our case, a modelling fault due to the approximation error).

Proposition 2: When approximating an irrational value for elapsed time in external transition function definition, the choice of approximation value should be consistent for all constants using this irrational number.

Formally: If we have the following DEVS definition of external transition function

$\delta_{ext}(S_i, e, a) = (S_j, 0) \qquad C_{irr} \le e \prec \infty$

$\delta_{ext}(S_i, e, a) = (S_k, 0) \qquad 0 \prec e \prec C_{irr}$

It should be approximated in RTA-DEVS model on the following form to avoid creating action-locks:

$\delta_{ext}(S_i, e, a) = (S_j, 0) \qquad C_r \le e \prec \infty$

$\delta_{ext}(S_i, e, a) = (S_k, 0) \qquad 0 \prec e \prec C_r$

C. *Effect of approximation error on model checking results*

The next question is how the approximation of irrational constants in **ta** or $\boldsymbol{\delta_{ext}}$ affect the formal verification of RTA-DEVS models. Would a result obtained from model checking RTA-DEVS models apply to the original DEVS?

When we approximate an irrational constant C_{irr} with a rational constant C_r, we introduce an error Δ such that $C_{irr} = C_r \pm \Delta$. This error appears then in constants used for time advance function or external transition functions. Verification of RTA-DEVS through transforming it to equivalent TA is done with reachability analysis. Would this analysis differ by introducing the error Δ when we move from DEVS to RTA-DEVS? Answering this question directly would require reachability analysis of the original DEVS with irrational constant values, and for the transformed RTA-DEVS model with the rational values (then, comparing results). This approach however is not feasible as reachability analysis for timed models with irrational constants is proven to be undecidable [7].

Therefore, we need to use an approximate approach to estimate the effect of Δ on the reachability analysis. This problem is equivalent to that of *robustness* of timed automata [8]. In robust timed automata, a robust model accepts an input sequence of events within a time interval. This is called a *bundle* of events that are close in time and the model still behaves the same with this bundle input.

Puri [9] extended the notion of robust TA to be those models that their reachability analysis remains the same with small drifts in clock models. In this definition, a model is not robust if for any small drift in clock rate, the reachability results change. In [10], it was proved that clock drifts in TA are equivalent to having a reaction delay by the implementation that increases guard constants by a small positive value **Δ**. The robustness problem is then transformed to an implementation problem, in which one need to find a value Δ that makes verification results correct. Further work in [11] showed a methodology to assess a model for implementability by using standard TA model checking tools, and proof that if a model is tolerant to a certain value Δ, it would also be correct with any value Δ' such that Δ' < Δ.

The results from the robustness theory of TA would be useful to check if a RTA-DEVS model formal verification results correctly applies to the original DEVS model. Given an error **Δ** introduced by approximation of irrational numbers in DEVS models, we non-deterministically model the possible transition from a state within an enlarged time interval with **Δ**. For example:

$\delta_{ext}(S_i, e, a) = (S_j, 0) \qquad C_{irr} \le e \prec \infty$

$\delta_{ext}(S_i, e, a) = (S_k, 0) \qquad 0 \prec e \prec C_{irr}$

and $C_{irr} = C_r \pm \Delta$, then, we enlarge the interval in which the external transition is enabled, i.e. to define it as:

$\delta_{ext}(S_i, e, a) = (S_j, 0) \qquad C_r - \Delta \le e \prec \infty$

$$\delta_{ext}(S_i, e, a) = (S_k, 0) \qquad 0 \prec e \prec C_r + \Delta$$

The model is transformed to an equivalent TA as in [4]. This model is then checked against the desired properties. With non-determinism in the model, UPPAAL checks the transition as if enabled during the interval, covering the point around the irrational number value. Hence, if the model checking results were correct, we conclude that the approximation did not introduce errors to the RTA-DEVS model.

D. Elevator / Elevator-Controller Example.

We use an example introduced in [12], which defines an elevator system composed of an elevator, elevator-controller and an environment representing a user pressing different buttons. This example was transformed from RTA-DEVS [4] to TA and verified to work correctly in UPPAAL. The example was extended by changing an irrational value in the controller model. State *stdbyMov* in Figure 5 have **ta** of √1000007 ≈ 1000.003 or √1000007 ≈ 1000.004; Δ= 0.001.

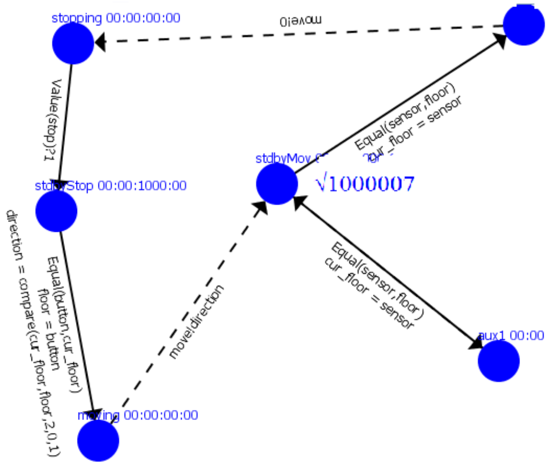

Figure 5: Elevator-Controller in DEVS Graphs notation.

The resulting TA models are shown in Figure 6.

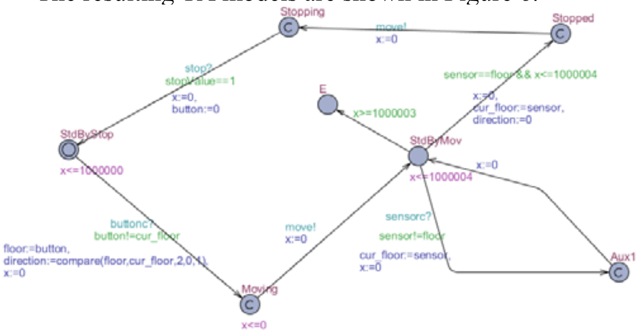

Figure 6: TA model with Non-deterministic behaviour.

In this model, we added node *E* and a transition from *StdByMov* to *E* that is enabled at elapsed time of x>=1000003. This TA is semantically equivalent to the DEVS model in Figure 5. However, this TA allows the transition from node *StdByMov* to node *Stopped* to be taken non-deterministically in the interval [0,1000004] while transition to E is enabled in [1000003, ∞]. This ensures covering the interval [0, √1000007] in UPPAAL model checking.

We ran the model checker to verify the non-deterministic version of the elevator-controller model along with the other components in the elevator system [4]. The results were successful and unchanged from results in [4]. The consistency of results in both non-deterministic and deterministic models indicates that the approximation error did not affect the verification results. Hence, for any value smaller than 0.001, the results would not be affected [11].

Although we could not verify the DEVS model in Figure 5 as a result of the irrational value of time advance function, our methodology approximates this model to a behaviorally equivalent RTA-DEVS and then to an equivalent TA, which can be used to verify the equivalent TA model.

IV. Conclusion

We introduced some of the problems that prevent classic DEVS models from being verified, and the conditions to obtain sound and behaviorally equivalent RTA-DEVS models from DEVS models. We also introduced a methodology based on recent theoretical work that can reveal if a given DEVS model being approximated by RTA-DEVS would have verification results unaffected by the approximation process. In reality, if a given DEVS model can not tolerate small approximation errors without changing its formal verification results, this DEVS model would be almost impossible to implement faithfully on a hardware platform as that platform would never be able to give exact timing due to digital clocks nature and transition delays. Inconsistency of verification results in our methodology would be an indication of such DEVS model.

References

[1] R. Alur, D. Dill. “Theory of Timed Automata". Theoretical Computer Science, volume 126, pg. 183-235, 1994.

[2] R. Alur. "Model Checking: From Tools to Theory", LNCS 5000, pg. 89–106, 2008.

[3] E. Clarke,"The Birth of Model Checking", Book: 25 Years of Model Checking, pg.1-26 , Springer Berlin / Heidelberg 2008.

[4] Hesham Saadawi, Gabriel A. Wainer. "Rational Time-Advance DEVS (RTA-DEVS)", Proceedings of 2010 Spring Simulation Conference (SpringSim10), DEVS Symposium, Orlando, FL, April 11-15 2010.

[5] BP Zeigler, H. Praehofer, T.G. Kim (2000) Theory of modeling and simulation, 2nd edn. Academic Press, New York.

[6] H. Bowman, R. Gomez, “Concurrency Theory: Calculi and Automata for Modelling Untimed and Timed Concurrent Systems”, Springer, 2006.

[7] J. Miller. "Decidability and complexity results for timed automata and semi-linear hybrid automata", Hybrid Systems: Computation and Control, LNCS Vol. 1790, 2000.

[8] V. Gupta, T. A. Henzinger, R. Jagadeesan. "Robust timed automata". Hybrid and Real-Time Systems, vol. 1201, p.331-345, 1997.

[9] A. Puri,"Dynamical properties of timed automata", Formal Techniques in Real-Time and Fault-Tolerant Systems, p. 210-227, 1998.

[10] M. De Wulf, L. Doyen, N. Markey,"Robustness and Implementability of Timed Automata", Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems, , p. 359-374, Springer Berlin 2004.

[11] M. De Wulf, L. Doyen, J-F. Raskin, "Almost ASAP Semantics: From Timed Models to Timed Implementations", Hybrid Systems: Computation and Control, p. 296-310, LNCS, 2004.

[12] H. Saadawi, G. Wainer. “Verification of Real-Time DEVS Models”, SpringSim'09, San Diego, CA March 2009.