(WIP) Approach to Reachability Analysis of Finite and Deterministic DEVS Models

Hae Young Lee
Seoul Women's University
Seoul, Republic of Korea
haelee@swu.ac.kr
+82-2-970-5605

ABSTRACT
This work-in-progress paper presents an approach to reachability analysis of finite and deterministic discrete event system specification (FD-DEVS) models, in which the transition system semantics of FD-DEVS is defined and the transition systems corresponding to FD-DEVS models are analyzed based on the underlying technology of timed automata.

Author Keywords
Model checking; reachability analysis; modeling and simulation; discrete event system specification (DEVS); timed automata.

ACM Classification Keywords
I.6.1 Simulation Theory. : Systems theory

INTRODUCTION
As discrete event system specification (DEVS) [1], which is a system-theoretic formalism for modeling and simulation of discrete event systems, has been applied to the engineering of embedded systems [2], several researchers have recently investigated reachability analysis for DEVS models [3-6]. Several approaches [3-5] use model checkers of timed automata (TA) [7], such as UPPAAL [8], for reachability analysis of DEVS. However, as shown in Figure 1(a), the conversion of DEVS models into TA may involve manual approximation, due to a subtle difference between them in semantics [3]. Another approach [6] is, as shown in Figure 1(b), to develop subclasses of DEVS whose reachability can be analyzed, together with verification algorithms for them. However, whenever a subclass is developed, a verification algorithm for the subclass may be required to be designed and proved.

In this paper, I propose an approach to reachability analysis of finite and deterministic DEVS (FD-DEVS) [5] models, in which the underlying technology of model checking is reused. To that end, I define the semantics of FD-DEVS for reachability analysis based on transition systems (TSs), low-level descriptions of systems, as in model checking of TA; most high-level descriptions are first interpreted as TSs (or region transition systems), as shown in Figure 1(c) [9]. Compared to the existing approaches, the merits of the proposed approach may include: 1) The underlying technology of model checking can be reused. 2) Manual conversion processes can be virtually eliminated. 3) The basic characteristics of DEVS can be kept during system modeling, while some subclasses (e.g., time constrained DEVS [3]) may lose that of DEVS. Thus, it might lay a more practical foundation for model checking of DEVS models.

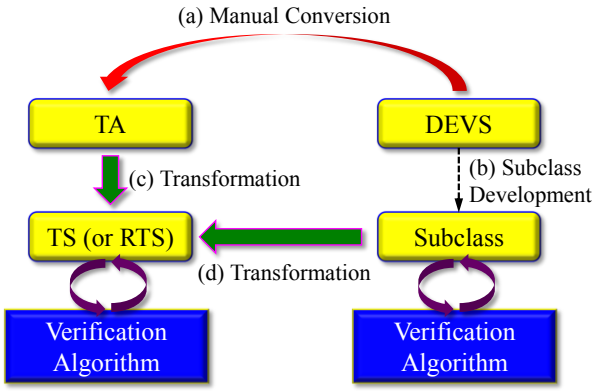

Figure 1. Approaches to reachability analysis of DEVS models.

FD-DEVS FORMALISM
In FD-DEVS, atomic models describe the behavior of deterministic and timed systems, while the hierarchical structure of the systems is expressed by coupled models.

Definition 2.1. Atomic FD-DEVS Model

An atomic FD-DEVS model is a tuple $M=<X, Y, S, s_0, \delta_{ext}, \sigma, \delta_{int}, \lambda, ta>$[1] where
- X is the finite set of input events,
- Y is the finite set of output events,
- S is the finite set of states,

[1] Some functions can be combined as in [5].

- $s_0 \in S$ is the initial state,
- $\delta_{ext}: S \times X \rightarrow S$ is the external transition function,
- $\sigma: S \times X \rightarrow \{0,1\}$ is the schedule update function,
- $\delta_{int}: S \rightarrow S$ is the internal transition function,
- $\lambda: S \rightarrow Y \cup \{\varnothing\}$ is the output function, where $\varnothing \notin Y$ denotes nonevent,
- $ta: S \rightarrow \mathbb{Q}_{[0,\infty]}$ is the time advance function. ■

The model starts in the initial state, s_0. If no external event occurs, it will stay in the state for the lifespan of the state, which is determined by the time advance function, *ta*. Upon reaching the lifespan, (an internal transition) it transitions to a new state by the internal transition function, δ_{int}. Simultaneously, an output event is produced by the output function, λ. The lifespan of the new state is determined by *ta*. In case of the occurrence of an input event before or on reaching the lifespan, (an external transition) it can transition to another new state by the external transition function, δ_{ext}. By the output of the schedule update function, σ, the lifespan of the new state can be updated or not. If the output is 1, (an external transition with event scheduling) the lifespan is determined by *ta*. If not, (an external transition without event scheduling) the lifespan is not updated. Note that each of its total state set consists of a state, the lifespan of the state, and the time elapsed since the last transition [6].

Definition 2.2. Coupled FD-DEVS Model

A coupled FD-DEVS is a tuple $N=\langle X, Y, D, \{M_i\}, EIC, EOC, IC\rangle$ where

- X is the finite set of input events,
- Y is the finite set of output events,
- D is the finite set of component references,
- for each $i \in D$, M_i is an FD-DEVS model,
- $EIC \subseteq X \times \bigcup_{i \in D} X_i$ is the set of external input couplings,
- $EOC \subseteq \bigcup_{i \in D} Y_i \times Y$ is the set of external output couplings,
- $IC \subseteq \bigcup_{i \in D} Y_i \times \bigcup_{j \in D} X_j$ $(i \neq j)$ is the set of internal couplings. ■

The formal semantics of atomic and coupled FD-DEVS models is found in [6].

SEMANTICS OF FD-DEVS FOR REACHABILITY ANALYSIS

As in TA, the first step towards the verification of FD-DEVS models is to define low-level descriptions of the models, TSs[2]. Thus, in this section, the semantics of FD-DEVS models for reachability analysis is defined in the forms of TSs.

Semantics of Atomic FD-DEVS Models for Reachability Analysis

For reachability analysis, an atomic FD-DEVS model can be interpreted as a TS. The underlying transition system of an atomic FD-DEVS model results from unfolding. As in [6], each of its states consists of a state of the FD-DEVS model, the lifespan of the state, and the time elapsed since the last external/internal transition that led to the state. Its initial state is comprised of the initial state of the model, the lifespan of the initial state of the model, and 0. Starting from the initial state, there are three possible ways in which an atomic FD-DEVS model can proceed: 1) by taking an external transition, 2) by taking an internal transition, or 3) by letting time progress while staying in the current state. In the first case, the corresponding transition of the underlying TS is labeled with the input event that causes the transition in the model. In the second case, it is labeled with the output event (including nonevent) produced by the output function at the internal transition. In the last case, it is labeled with a positive rational number indicating the amount of time that has progressed. Thus, the actions of the TS include: 1) the input events of the model, 2) the output events, and 3) nonnegative rational numbers.

Definition 3.1. Transition System Semantics of an Atomic FD-DEVS Model

The transition system $TS(M)$ of an atomic FD-DEVS $M=\langle X, Y, S, s_0, \delta_{ext}, \sigma, \delta_{int}, \lambda, ta\rangle$ is the tuple $\langle Q, Act, \rightarrow, I\rangle$ where

- $Q=\{(s, l, e) | s \in S, l \in \mathbb{Q}_{[0,\infty]}, e \in \mathbb{Q}_{[0,\infty]}\}$ is the set of states, where l is the lifespan of s and e is the time elapsed since the last *schedule-updating* transition,
- $Act=X \cup Y \cup \{\varnothing\} \cup \mathbb{Q}_{[0,\infty]}$ is the set of actions,
- the transition relation, →, is defined by the following rules:
  (a) external transition:

$$\frac{x \in X \wedge \delta_{ext}(s,x) = s' \wedge \sigma(s,x) = 1}{(s,l,e) \xrightarrow{x} (s', ta(s'), 0)}$$

  or

$$\frac{x \in X \wedge \delta_{ext}(s,x) = s' \wedge \sigma(s,x) = 0}{(s,l,e) \xrightarrow{x} (s', l, e)}$$

  (b) internal transition:

$$\frac{y \in Y \wedge e = l \wedge \delta_{int}(s) = s'}{(s,l,e) \xrightarrow{y} (s', ta(s'), 0)}$$

  (c) delay transition:

$$\frac{d \in \mathbb{Q}_{[0,\infty]} \wedge e + d \leq l}{(s,l,e) \xrightarrow{d} (s, l, e+d)}$$

- $I=\{(s_0, ta(s_0), 0)\}$ is the set of initial states. ■

The system initially starts in the initial state, $q=(s_0, ta(s_0), 0)$. (c) Idling in the state for a non-negative amount of time (a delay transition) is allowed within the lifespan of the state. For an input event x, there is the likelihood of receiving x within the lifespan. Thus, (a) an external transition corresponding to $\delta_{ext}(s_0, x)=s$ can be selected non-deterministically and taken, within the lifespan. Some events make event scheduling, while the others do not. In case of an external transition with schedule update (i.e. $\sigma(s_0, x)=1$), the lifespan of a new state is updated and the elapsed time is reset. But they are not updated in case of a transition without schedule update. (b) Upon reaching the lifespan, the internal transition, corresponding to $\delta_{int}(s_0)=s'$, labeled

[2] or region transition systems.

with the output event, is taken. Note that a set of atomic propositions, AP, and a labeling function, L:$S \rightarrow 2^{AP}$, are omitted since they are not defined in FD-DEVS. But if necessary, we can simply use $AP=S$ and $L(s)=\{s\}$.

Strictly speaking, $TS(M)$ and M differ in the semantics of transitions with respect to model execution (i.e., simulation); external transitions in $TS(M)$ are taken non-deterministically, while every external transition in M is taken only upon the occurrence of an input event. However, in terms of reachability analysis, they are equivalent: if a state is reachable in M, it is also reachable in $TS(M)$, and vice versa.

Semantics of Coupled FD-DEVS Models

A coupled FD-DEVS model can be also interpreted as a TS through unfolding: Its 'global' states is comprised of 'local' states of the TSs of the components (atomic or coupled DEVS models) in the model. Its initial state consists of the initial states of the TSs. A coupled FD-DEVS model can proceed: 1) by receiving an input event, 2) by the occurrence of an internal event in a component, or 3) by letting time progress. Thus, the actions of the TS include the input events of the model, the output events of the components, and nonnegative rational numbers. In the model, events are passed in a similar way to handshaking of channel systems (i.e., synchronous message passing). For an input event, the TSs of the components, which are affected by the event, take the corresponding external transitions. For an internal event, the TS of the 'source' component takes the corresponding internal transition, while the TSs of the 'target' components, which are connected to the 'source' component, take the corresponding external transitions. For a nonnegative rational number, the corresponding delay transition is taken.

Definition 3.2. Transition System Semantics of a Coupled FD-DEVS Model for Reachability Analysis

The transition system $TS(N)$ of coupled FD-DEVS $N=<X, Y, D, \{M_i\}, EIC, EOC, IC>$ is the tuple $<Q, Act, \rightarrow, I>$ where

- $Q=Q_1 \times \cdots \times Q_n$, where Q_i is the set of states of $TS(M_i)$,
- $Act=X \cup \bigcup_{i \in D} Y_i \cup \{\varnothing\} \cup \mathbb{Q}_{[0,\infty]}$, where and Y_i is the set of output events of M_i,
- $\rightarrow$ is defined by the following rules:
  (a) external transition for $x \in X$:
  $(\cdots, q_i, \cdots) \xrightarrow{x} (\cdots, q_i', \cdots)$,
  where
  $q_i \xrightarrow{x_i}_i q_i'$ if $(x, x_i) \in EIC \wedge x_i \in X_i$,
  $q_i \xrightarrow{0}_i q_i'$ otherwise,
  (b) internal transition for $y \in \bigcup_{i \in D} Y_i$:
  $(\cdots, q_i, \cdots) \xrightarrow{y} (\cdots, q_i', \cdots)$,
  where
  $q_i \xrightarrow{y}_i q_i'$ if $y \in Y_i \wedge e_i = l_i$,
  $q_i \xrightarrow{x_i}_i q_i'$ if $(y, x_i) \in IC \wedge x_i \in X_i$,
  $q_i \xrightarrow{0}_i q_i'$ otherwise,
  (c) delay transition for $d \in \mathbb{Q}_{[0, \min_{i=1,\cdots n} l_i - e_i]}$:
  $(\cdots, q_i, \cdots) \xrightarrow{d} (\cdots, q_i', \cdots)$,
  where
  $q_i \xrightarrow{d}_i q_i'$,
- $I=\{(q_1, \cdots, q_n)\}$. ■

The initial state of the system is $(q_1, \cdots, q_n)$. (c) A delay transition is allowed until the occurrence of the earliest internal event in the components; each of the TSs of $\{M_i\}$ takes the transition. (a) An external transition for an input event x can be taken non-deterministically until the occurrence of the earliest internal event; each of the TSs affected by x takes the corresponding external transition, while the other TSs take the zero-delay transition. (b) When an internal event occurs in a component, the TS of the model takes the corresponding internal transition. In the transition, the TS of the component and the TSs affected by the output event of the component take the corresponding internal and external transitions, respectively, while the other TSs take the zero-delay transition. Although $TS(N)$ and N differ in terms of simulation, they are equivalent with respect to reachability analysis.

REACHABILITY ANALYSIS APPROACH

Since an FD-DEVS model and the TS of the model are equivalent in terms of reachability analysis, to examine the reachability of a state in an FD-DEVS model, we have only to: 1) obtain the TS of the model, as shown in Figure 1(d), and 2) check that of the state in the TS. The TS could be automatically obtained by Definitions 3.1 and 3.2. The reachability in the TS could be analyzed by model checking of TA; it might be implemented with a slight modification of an open source model checker. Therefore, with the proposed approach, the reachability analysis for FD-DEVS models could be fully automated.

In fact, the TSs of FD-DEVS models may not be directly verified since each of them has infinitely many states and transitions. In model checking of TA, another finite description of infinite systems, such as region transition systems (RTSs), is internally used to verify TA. Although this issue is not covered in this paper, RTSs could be obtained from the TSs of the models, while minimizing human expertise.

CONCLUSIONS AND FUTURE WORK

This paper presented an approach to the reachability analysis of FD-DEVS models, in which TSs are obtained from FD-DEVS models by the TS semantics, and then verified based on the underlying technology of model checking. Compared to Saadawi and Wainer's approach [4-5], it may virtually eliminate manual conversion processes, which might be error-prone and even time-consuming. With the proposed approach, the reachability analysis could be fully automated, as in model checking of TA. Compared to Hwang and Zeigler's approach [6], it can reuse the

underlying technology of model checking, so that extra efforts to develop and prove verification algorithms may be minimized; for each new subclass, we just need to define its TS semantics. Also, the implementation efforts of DEVS model checkers may be drastically reduced since it might be done with a slight modification of an existing model checker.

The proposed approach might be just a first step towards modeling checking of DEVS. Thus, many issues should be further investigated in theoretical and practical aspects. For example, reachability analysis of more expressive DEVS formalisms (e.g., nondeterministic and elapsed-time-sensitive DEVS [10]) should be enabled to be exhaustively verified, in order to provide easiness of modeling. More to the point, it must be implemented for practical applications of DEVS model checking.

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (NRF-2013R1A1A1006542).

REFERENCES

1. Zeigler, B.P., Paraehofer, H., and Kim, T.G. *Theory of Modeling and Simulation*. 2nd Ed. Academic Press, 2000.
2. Molter, H.G., Kohlmann, J., and Huss, S.A. Automated generation of embedded systems software from timed DEVS model of computation specifications. In *Proc. 15th Euromicro Conference on Digital System Design*, IEEE (2012), 700-707.
3. Dacharry, H.D., and Giambiasi, N. A formal verification approach for DEVS. In *Proc. 2007 Summer Simulation Multiconference*, ACM (2007), 312-319.
4. Saadawi, H., and Wainer, G. From DEVS to RTA-DEVS. In *Proc. 14th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications*, IEEE (2010), 207-210.
5. Saadawi, H., and Wainer, G. Principles of discrete event system specification model verification. *Simulation* 89, 1 (2013), 41-67.
6. Hwang, M.H., and Zeigler, B.P. Reachability graph of finite and deterministic DEVS networks. *IEEE Transactions on Automation Science and Engineering* 6, 3 (2009), 454-467.
7. Alur, R., and Dill, D.L. A theory of timed automata. *Theoretical Computer Science* 126, 2 (1994), 183-235.
8. UPPAAL, http://www.uppaal.org/.
9. Baier, C., and Katoen, J.P. *Principles of Model Checking*. MIT Press, 2008.
10. Yoon, J.H. and Lee, H.Y. Nondeterministic and elapsed-time-sensitive DEVS. In *Proc. 2014 Spring Simulation Multiconference*, IEEE (2014).