Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

TOWARD PRECISE SEMANTICS OF ACTIONS

Abdurrahman Alshareef

Arizona Center for Integrative Modeling & Simulation
School of Computing, Informatics and Decision Systems Engineering
Arizona State University
699 S. Mill Avenue
Tempe, AZ, 85281, USA
alshareef@asu.edu

ABSTRACT

Action is the fundamental unit of behavioral specification in models. We propose the use of Discrete EVent System Specification (the DEVS formalism) to specify the semantics of actions. Then, coupling is used to form different kinds of behavioral models. The statecharts and activities are two different approaches by which the system behavior can be described. Actions are at the core of these two approaches and therefore their specifications can collectively serve as a significant part of the overall behavior alongside with behavior of other parts such as control. Thus, we propose an approach introducing the concepts of time and state as defined in DEVS for actions; these serve as an abstraction for modeling a wide range of systems.

1 INTRODUCTION

It is important to use abstractions, languages, and metamodels for behavioral specifications to overcome complexity and scale demand. Any tiny change in the behavioral specification of a model may result in vastly different dynamics. Thus, it is necessary to tame this intrinsic characteristic in behavioral modeling. The problem is approached through the means by which the behavioral specifications are described. Using an ad-hoc approach may not scale especially due to the complexity arising from a mesh of actions.

2 THE ATOMIC MODEL AND THE ACTION

The goal is ultimately to create a means for actions to be specified as precise and flexible as possible for systems that interact in arbitrary, but well-formed, fashion. When modularity is maintained, the system can grow according to the principles of coupling and composability. Thus, we propose DEVS specification creating a correspondent atomic model for each action (see Figure 1). The formal specification of the atomic model $DEVS = (X_M, Y_M, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta)$ is defined with respect to the semantics of the corresponding action. In conjunction with the other defined atomic models for describing a certain semantics for some behavioral diagram element, such as decision node (Alshareef and Sarjoughian 2017), these elements can together formulate the correspondent chains of actions (i.e., coupled models) (Sarjoughian 2017). The processing time represent a wider range of execution semantics including an execution step. The time advance function is utilized in the correspondent action model to establish the linkage to the time base whether it is a real value, discrete, or any other as discussed in the literature in some DEVS variant formalisms. Such restrictions can be leveraged to work around some computational compromises to satisfy certain needs.

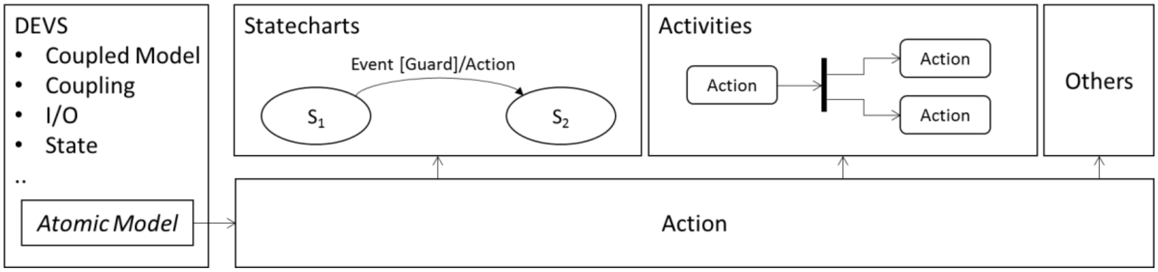

Figure 1: The action abstraction is situated at the heart of many behavioral specifications and thus used as a bridge between the formal specification and other semi-formal or informal modeling approaches.

3 A PROCESSOR MODEL

An Activity-based DEVS abstraction is specified for a processor model described in (Zeigler, Praehofer, and Kim 2000). The processor receives a bag of inputs. It distinguishes between inputs for different kinds of communication. Data is commonly exchanged as well as control according to their intended purposes. Instances of some classes with variables are created at some locus. Therefore, inputs can be used for communicating information about the model to different components. Although using the notion of time may significantly differ in the context of action, the processing time can take various values and then can be used to model ordering or concurrency semantics such as in the join node defined in activity abstraction. It also may refer to duration assuming that actions are not instantaneous. Another aspect is to consider the instant of time on which an action may occur or start. Since action is considered to be the fundamental unit of behavioral specification, their influence on the state is specified explicitly in order to be able to provide guarantees across the system of interest. This elaboration is certainly beneficial especially when considering the possibility to extend the notion of action with time. This forms a strong basis toward achieving the goal of having a precise time-based semantics for actions.

The processor remains idle while not receiving inputs. It may store received inputs in a unitary storage or multiple inputs in a queue. Actions also have access to other resources. Their situation in the context of the object model is yet to be examined. In the UML (OMG 2012), the action can be created within a context of behavior classifier. Along with control elements, they constitute the overall behavior. Different mechanisms of control then reveal the nature of the behavioral model as shown in Figure 1. As far as the action is concerned, it can initially or subsequently takes some inputs. Then, the semantics of the action takes place considering these inputs causing some changes on the state and possibly producing outputs.

To conclude, the capability of developing the behavioral specification in stages is beneficial but yet quite challenging. In our approach, we investigate metamodels, frameworks, and tools to increase accessibility without compromising our models or any of the artifacts thereof. The precise semantics of actions can take place as an essential step in enabling simulation-based studies. Along with supporting control structures, useful simulations for Systems of Systems can be attained.

REFERENCES

Alshareef, A., and H. S. Sarjoughian. 2017. “DEVS specification for modeling and simulation of the UML activities”. In Proceedings of the Symposium on Model-Driven Approaches for Simulation Engineering. Society for Computer Simulation International.

Sarjoughian, H. S., 2017. “Restraining complexity and scale traits for component-based simulation models”. In Proceedings of the 2017 Winter Simulation Conference, Las Vegas, NV, USA, IEEE Press.

Zeigler, B. P., H. Praehofer, and T. G. Kim. 2000. Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems. Academic press.

OMG 2012. “Unified Modeling Language version 2.5”.