Transformations to and from the CIF and behavioral semantics

R.R.H. Schiffelers

January 7, 2010

1 Compositional Interchange Format for Hybrid Systems (CIF

The main purpose of the Compositional Interchange Format (CIF)[1], that has originally been developed in HYCON, see [2] and [3, 4, 5, 6], is to establish inter-operability of a wide range of tools by means of model transformations to and from the CIF. In addition, the CIF provides a generic modeling formalism and tools for a wide range of untimed, timed and hybrid systems. An overview on previous related work on interchange formalisms, such as found in [7], [8], [9], can be found in [3, 4].

The concepts in the CIF and the relations between them are defined in a so-called conceptual or meta model, see [1]. This model is defined in terms of (Ecore) class diagrams [10]. From these class diagrams, XML Schema definitions (XSDs) [11] have been generated. The XML Schema definitions as well as the Ecore models can be obtained electronically via [12].

Regarding concrete syntax and behavioral semantics, the CIF consists of an *abstract* format, which is specified using a mathematical notation and is used for the definition of the formal semantics, and a *concrete* format, as defined in [5], which is specified in the ASCII character set by a formal grammar and is used as a modeling language. The operational semantics of a model in the abstract format is defined formally in a SOS style [13]. It defines the *mathematical meaning* of a hybrid model in terms of an hybrid transition system. The semantics of a model in the concrete format is formally defined by means of a mapping to the abstract format. The advantage of having two formats is that each can be tailored to its specific purpose. In general, the abstract format has fewer concepts in order to simplify the semantics, while the concrete format has 'syntactic sugar' and more emphasis on backward compatibility in order to facilitate modeling. In [14], the concepts of the (concrete) CIF are illustrated by means of a hybrid model of a supermarket refrigeration system that exhibits both, nonlinear DAE dynamics as well as significant discrete dynamics, and serves as a challenging case study for hybrid control techniques in several European research projects. More information about CIF and CIF tools allowing, e.g., simulation and visualization, can be found in [12].

2 Model transformations

The CIF serves as the basis of the European research project MULTIFORM, see [15]. The main objective of this project is to develop interoperability of tools and methods based on different modeling formalisms to provide integrated coherent tool support for the design of large complex controlled systems. Within MULTIFORM, algorithms and tools for the translation to/from the CIF will be defined for a large variety of modeling languages, including Chi, gPROMS, Matlab/Simulink, Modelica, MUSCOD-II, PHAVer, and UPPAAL.

Depending on the availability of a formal definition of the behavioral semantics of a language, two different categories of transformations can be distinguished:

- Transformations from formalisms that have formal semantics to the CIF (vice verse).
- Translations from formalisms that do not have formal semantics to the CIF (vice verse).

2.1 Transformations between formalisms with formal semantics

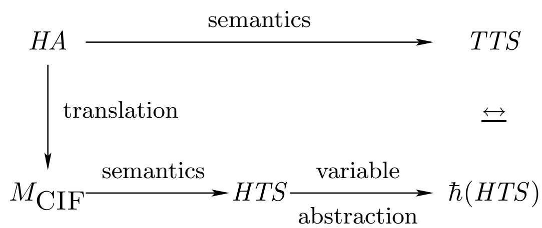
In case of a translation where the source formalism as well as the target formalisms have formal semantics, one can define an equivalence relation [1] between the semantics of the two formalisms. By means of mathematical proof, it can be shown that the behavior of an input model and the behavior of the output model of the translation are equivalent.

To illustrate this approach, consider the translation of hybrid automata [16] to the CIF. The semantics of a hybrid automaton is a timed transition system with two types of transitions: action transitions (corresponding to control switches) and time transitions (corresponding to continuous behavior in a control mode). On the other hand, the semantics of a CIF automaton is a hybrid transition system which also has these two types of transitions. The main difference between these semantics is in the labeling of the action and time transitions. In timed transition systems the labels of action transitions are simply the events of the hybrid automaton, whereas the labels of the action transitions of a hybrid transition system also contain the valuations of the model variables prior to and after the action. For time transitions, the labels in a timed transition system contain only the duration of the time transition whereas time transitions in hybrid transition systems also have the trajectory of the model variables as a label. Finally, a timed transition system can have many initial states whereas a hybrid

[1] In fact multiple equivalence relations can be defined depending on the properties to be preserved.

transition system has only one initial state. This one initial state captures the behavior of all the initial states of the timed transition system.

Let $\hbar$ be a mapping that maps a hybrid transition system onto a timed transition system by removing valuations from action transitions and trajectories from time transitions. Furthermore, let HA be a hybrid automaton and let M_{CIF} be the CIF specification associated to it by its translation. Furthermore, let TTS and HTS be the semantics of HA and M_{CIF}, respectively.



Then, there exists a (strong-)bisimulation relation [17, 18], denoted by $\underline{\leftrightarrow}$, between the states of TTS and the states of $\hbar(HTS)$ such that any transition from an initial state of TTS can be simulated by the initial state of $\hbar(HTS)$ and each transition from the initial state of $\hbar(HTS)$ is simulated by some initial state of TTS.

Examples of similar translations including their correctness proofs can be found in [19, 20] (Translation of χ to 1) piecewise affine systems and 2) hybrid automata), and [21, 22] (Translation of χ to UPPAAL).

2.2 Transformations between formalisms with formal semantics

In case of a translation where the source formalism does not have a formal semantics and the target formalism has a formal semantics, then, by means of a (formally defined) translation, formal semantics is given to the source formalism. An example of this approach can be found in [23] that defines bidirectional transformations between gPROMS and Modelica via the CIF. The correctness of the translations has been validated by means of comparing the simulation results of several input models and their respective output models.

References

[1] D.E. Nadales Agut, D.A. van Beek, R.R.H. Schiffelers, D. Hendriks, and J.E. Rooda. Revision and extension of the CIF including data types and transfer format. Technical Report D 1.1.1, MULTIFORM, 2009.

[2] HYCON Network of Excellence. http://www.ist-hycon.org/, 2005.

[3] D. A. van Beek, M. A. Reniers, R. R. H. Schiffelers, and J. E. Rooda. Foundations of an interchange format for hybrid systems. In Alberto Bemporad, Antonio Bicchi, and Giorgio Butazzo, editors, *Hybrid Systems: Computation and Control, 10th International Workshop*, volume 4416 of *Lecture Notes in Computer Science*, pages 587–600, Pisa, 2007. Springer-Verlag.

[4] D. A. van Beek, M. A. Reniers, J. E. Rooda, and R. R. H. Schiffelers. Revised hybrid system interchange format. Technical Report HYCON Deliverable D3.6.3, HYCON NoE, 2007.

[5] D. A. van Beek, M. A. Reniers, J. E. Rooda, and R. R. H. Schiffelers. Concrete syntax and semantics of the compositional interchange format for hybrid systems. In *17th Triennial World Congress of the International Federation of Automatic Control*, pages 7979–7986, Seoul, Korea, 2008.

[6] D. A. van Beek, P. Collins, D. E. Nadales, J.E. Rooda, and R. R. H. Schiffelers. New concepts in the abstract format of the compositional interchange format. In A. Giua, C. Mahuela, M. Silva, and J. Zaytoon, editors, *3rd IFAC Conference on Analysis and Design of Hybrid Systems*, pages 250–255, Zaragoza, Spain, 2009.

[7] MoBIES team. HSIF semantics. Technical report, University of Pennsylvania, 2002. internal document.

[8] Alessandro Pinto, Luca P. Carloni, Roberto Passerone, and Alberto L. Sangiovanni-Vincentelli. Interchange format for hybrid systems: Abstract semantics. In João P. Hespanha and Ashish Tiwari, editors, *Hybrid Systems: Computation and Control, 9th International Workshop*, volume 3927 of *Lecture Notes in Computer Science*, pages 491–506, Santa Barbara, 2006. Springer-Verlag.

[9] Stefano Di Cairano, Alberto Bemporad, and Michal Kvasnica. An architecture for data interchange of switched linear systems. Technical Report D 3.3.1, HYCON NoE, 2006.

[10] Dave Steinberg, Frank Budinsky, Marcelo Paternostro, and Ed Merks. *EMF Eclipse Modeling Framework*. Addison-Wesley, 2009.

[11] Henry S. Thompson, David Beech, Murray Maloney, and Noah Mendelsohn. XML schema part 1: Structures second edition. http://www.w3.org/TR/xmlschema-1/, 2004.

[12] Systems Engineering Group TU/e. CIF toolset. http://se.wtb.tue.nl/sewiki/cif, 2008.

[13] Gordon D. Plotkin. A structural approach to operational semantics. *Journal of Logic and Algebraic Programming*, 60-61:17–139, 2004.

[14] C. Sonntag, R. R. H. Schiffelers, D. A. van Beek, J. E. Rooda, and S. Engell. Modeling and simulation using the Compositional Interchange Format for hybrid systems. In I. Troch and F. Breitenecker, editors, *6th International Conference on Mathematical Modelling*, Vienna, Austria, 2009.

[15] MULTIFORM consortium. Integrated multi-formalism tool support for the design of networked embedded control systems MULTIFORM. http://www.multiform.bci.tu-dortmund.de, 2008.

[16] T. A. Henzinger. The theory of hybrid automata. In M. K. Inan and R. P. Kurshan, editors, *Verification of Digital and Hybrid Systems*, volume 170 of *NATO ASI Series F: Computer and Systems Science*, pages 265–292. Springer-Verlag, New York, 2000.

[17] R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer-Verlag, 1980.

[18] D. M. R. Park. Concurrency and automata on infinite sequences. In P. Deussen, editor, *Proceedings 5^{th} GI Conference*, volume 104 of *Lecture Notes in Computer Science*, pages 167–183. Springer-Verlag, 1981.

[19] K. L. Man and R. R. H. Schiffelers. *Formal Specification and Analysis of Hybrid Systems*. PhD thesis, Eindhoven University of Technology, 2006.

[20] D. A. van Beek, J. E. Rooda, R. R. H. Schiffelers, K. L. Man, and M. A. Reniers. Relating hybrid Chi to other formalisms. *Electronic Notes in Theoretical Computer Science*, 191:85–113, 2007.

[21] E. M. Bortnik, D. A. van Beek, J. M. van de Mortel-Fronczak, and J. E. Rooda. Verification of timed Chi models using Uppaal. In J. Filipe, J. A. Cetto, and J.-L. Ferrier, editors, *ICINCO 2005, Second International Conference on Informatics in Control, Automation and Robotics*, pages 486–492, Barcelona, 2005. INSTICC Press.

[22] E. M. Bortnik. *Formal methods in support of SMC design*. PhD thesis, Eindhoven University of Technology, 2008.

[23] Christian Sonntag, Martin Hfner, and Adalat Jabrayilov. Realization of the model exchange with Modelica and gPROMS. Technical Report D 1.4.1, MULTIFORM, 2009.