

Specification and Analysis of Discrete Behavior of Hybrid Systems in the Workbench ISMA

Yury V. Shornikov, Dmitry N. Dostovalov, Maria S. Myssak, Artem N. Komarichev, Andrey M. Tolokonnikov

Department of Automated Control Systems, Novosibirsk State Technical University, Novosibirsk, Russia Email: shornikov@inbox.ru, dostovalov.dmitr@mail.ru, maria_myssak@mail.ru, calmnessart@gmail.com, wrt-tolok@yandex.ru

Received 2013

ABSTRACT

Hybrid systems are important in applications in CAD, real-time software, robotics and automation, mechatronics, aeronautics, air and ground transportation systems, process control, and have recently been at the center of intense research activity in the control theory, computer-aided verification, and artificial intelligence communities. In the past several years, methodologies have been developed to model hybrid systems, to analyze their behavior, and to synthesize controllers that guarantee closed-loop safety and performance specifications. These advances have been complemented by computational tools for the automatic verification and simulation of hybrid systems. Modern technologies of computer simulation tools include preparing, debugging, analysis and calculation of effective program models, meaningful interpretation of research results.

Keywords: Hybrid System; Statechart; Event Detection; Differential-algebraic Equations; Simulation Tools

1. Introduction

There are many systems (mechanical, electrical, chemical, biological, etc.), the behavior of which can be conveniently described as a sequential change of continuous modes. These systems are referred to as hybrid (HS) or event-continuous. Each mode is given by a set of differential-algebraic equations with the following constraints:

$$y' = f(x, y, t), x = \varphi(x, y, t),$$

$$pr : g(x, y, t) < 0,$$

$$t \in [t_0, t_k], x(t_0) = x_0, y(t_0) = y_0,$$

$$x \in R^{N_x}, y \in R^{N_y}, t \in R,$$

$$f : R^{N_x} \times R^{N_y} \times R \to R^{N_y},$$

$$\varphi : R^{N_x} \times R^{N_y} \times R \to R^{N_x},$$

$$g : R^{N_x} \times R^{N_y} \times R \to R^S.$$

(1)

The vector-function g(x, y, t) is referred to as event function or guard. A predicate pr determines the conditions of existence in the corresponding mode or state. Inequality g(x, y, t) < 0 means that the phase trajectory in the current mode should not cross the border g(x, y, t) = 0. Events occurring in violation of this condition and leading to transition into another mode without crossing the border are referred to as one-sided. Many practical problems are characterized by stiff modes, and the surface of boundary g(x, y, t) = 0 has sharp angles or solution has several roots at the boundary [1]. Numerical analysis of such models by traditional methods is difficult or impossible, as it gives incorrect results. Therefore it is necessary to use special methods to detect events accurately.

Computer analysis of these systems is typically performed in simulation tools, best of which are Charon (USA), AnyLogic (Russia), Scicos (France), MVS (Russia), Hybrid Toolbox and HyVisual (USA), DYMOLA (Sweden) and etc.

The specification of the discrete behavior of HS reflects the instantaneous discrete transitions from one state to another one. State diagrams allow to easily describe HS models of any formalism and to represent semantics of HS modes and mechanism of discrete transitions in intuitive manner.

This paper describes the features of design and HS models analysis in the ISMA instrumental environment [2].

2. Discreet Behavior of Hybrid Systems and Its Specification

Statecharts have been widely used since a variant has become a part of the Unified Modeling Language (UML). This way of describing hybrid systems commonly used in information technology and related fields. This approach was proposed by D. Harrel [3] and is a description of complex dynamic systems. Statecharts represent directed graph whose nodes correspond to states of continuous object, arcs – to discrete transitions (changing of states). System can be represented by statecharts if it is characterized by a finite number of continuous states. The method allows to describe the operating logic of the object in visual form in order to represent its behavior at the continuous state changes conditions. For example, **Figure 1** shows a Harel statechart, where A,B,C corresponds to the continuous states; α , β , γ – to the predicates (conditions).

Graphics Editor

Graphics editor was designed by our team for a clearer representation of statecharts (**Figure 2**).

The editor is developed on the object-oriented programming language Java. The main reason for choosing this language is that programs written in Java are crossplatform.



Figure 1. Example of statetchart.



Figure 2. Graphics editor.

The right panel is for editing of the base statechart. On the left panel there are elements that can be dragged to the right panel for further work with them (drag-anddrop). In addition, there was developed menu that is standard for applications running on the Windows.

The editor commands are controlled by hot keys. These keys are responsible for copy, paste, delete, and edit the selected state or arc. Thus the editing of states and transitions is a convenient process.

Designed application allows you to convert a graphical representation of simulation model. The transformation algorithm includes the steps of the analysis and the selection of all the states and transition, relationships definition and generation in computational model taking into account the defined relations.

3. Event Detection in Hybrid Systems

The correct analysis of hybrid models is significantly depends on the accuracy of detection of the change of the local states of the HS. Therefore, the numerical analysis is necessary to control not only the accuracy and stability of the calculation, but also the dynamics of the eventfunction. The degree of approximation by the time the event occurred is defined by the behavior of event driven function.

Consider the mode of one-sided HS as a Cauchy problem with constraints (1).

Any non-linear guard g(x, y, t) can be reduced to linear form by adding the phase variable z = g(x, y, t). As a result, problem (1) can be rewritten as follows (for simplicity we omit the algebraic equations and proceed to the autonomous Cauchy problem:

$$y' = f(y), z' = \frac{\partial g}{\partial y} \cdot f(y) + \frac{\partial g}{\partial t}, z < 0.$$

In solving these problems by using explicit methods [4], we obtain $y_{n+1} = y_n + h_{n+1}\varphi_n$, n = 0, 1, 2, ... Then the event dynamic is described as

$$g_{n+1} = g(y_n + h_{n+1}\varphi_n, t_n + h_{n+1})$$

Decomposing the g_{n+1} g_{n+1} in a Taylor series and taking into account the linearity of g(y,t), we obtain the dependence of g_{n+1} of the projected step h_{n+1} :

$$g_{n+1} = g_n + h_{n+1} \left(\frac{\partial g_n}{\partial y} \cdot \varphi_n + \frac{\partial g_n}{\partial t} \right)$$
(2)

Theorem. The choice of the step according to the formula

$$h_{n+1} = (\gamma - 1) \frac{g_n}{\frac{\partial g_n}{\partial \nu} \cdot \varphi_n + \frac{\partial g_n}{\partial t}}$$
(3)

where $y \in (0,1)$, provides the event-dynamics behavior

as a stable linear system, the solution of which is asymptotically approaching to the surface g(y,t) = 0.

Proof. Substituting (3) in (2), we have $g_{n+1} = \gamma g_n$, n = 0, 1, 2, ... Converting recurrently this expression we get $g_{n+1} = \gamma^{n+1} g_0$. Given that $\gamma < 1$, then $g_n \to 0$ takes place when $n \to \infty$. In addition, condition $\gamma > 0$ implies that function g_n does not change sign. Therefore, when $g_0 < 0$, $g_n < 0$ will be valid for all *n*. Then the guard condition will potentially never cross potentially the dangerous area $g(\gamma_n, t_n) = 0$, which completes the proof.

Let us formulate integration algorithm, taking into account the forecast of step by an event function. Let the solution y_n in the point t_n is calculated with step h_n . Then the approximate solution at the point t_{n+1} is calculated as follows.

Step 1. Calculate the function $f_n = f(y_n)$.

Step 2. Calculate the functions $g_n = g(y_n, t_n)$, $\partial g_n / \partial y = \partial g(y_n, t_n) / \partial y$, $\partial g_n / \partial t = \partial g(y_n, t_n) / \partial y$. **Step 3.** Calculate the step h_{n+1}^p by (3), where

 $\varphi_n = f_n.$

Step 4. A new step h_{n+1} is calculated by the formula $h_{n+1} = min(h_{n+1}^p, h_{n+1}^n)$, where h_{n+1}^n is step by selecting the numerical method of integration.

Step 5. Go to next step of integration.

In the practical implementation of the algorithm it is necessary to consider the following. Near the boundary regime denominator (3) will be positive, and away from the boundary g(y,t)=0 becomes negative. Then, defining the direction of change event-function, we cannot perform step 4 of the algorithm and do not impose any further restrictions on the integration step if the eventfunction is removed from the boundary of state.

Test of the Event-detection Algorithm

To illustrate the event-detection algorithm we consider a hybrid system of two oscillating masses on springs [5], shown in **Figure 3**.

The system can be in one of two local states: when masses move separately or together. Mathematical model is not presented here because of the proximity to description of the computer model. A computer model of system in the ISMA shown in **Figure 4**.

Qualitative simulation results are obtained with enabled event-detection algorithm (Figure 5). Traditional analysis of the system without using the event-detection algorithm does not allow to obtain valid results as shown in (Figure 6).



Figure 3. The system of two oscillating masses.



Figure 4. A computer model of the system in ISMA instrumental environment.



Figure 5. Calculation results (using the event-detection algorithm).



Figure 6. Calculation results (excluding the event-function dynamics).

4. Dry Friction Dynamics Simulation

Simulation object shown in the **Figure 7** represents a system described in [6].



Figure 7. Simulation object.

The motion of body has an oscillation nature, where the motion stage and the resting stage are changing periodically. The motion stage is divided on the motion in range of Shtribeck effect at low speeds and the motion in range of Amonton-Coloumb law.

The Hybrid model in the formalism of hierarchical behavioral maps designed in the ISMA is shown in **Figure 8**.

Figure 8(a) corresponds to the top-level behavioral map, Figure 8(b) illustrates the motion stage.



Figure 8. The hybrid model of dry friction.

Obtained simulation results are shown in the **Figure 9** and **Figure 10**. Time charts of tension of spring x, body velocity V, friction force F, displacement of the body relative to the origin 1 and local state z are presented in **Figure 9. Figure 10** illustrates the phase diagram.



Figure 9. Simulation results. Time characteristics.



Figure 10. Simulation results. Phase diagram.

5. Acknowledgements

This work was supported by the Russian Foundation for

Basic Research under grant 11-01-00106-a.

REFERENCES

- J. Esposito, V. Kumar and G.J. Pappas, "Accurate Event Detection for Simulating Hybrid Systems," In: *Hybrid Systems: Computation and Control (HSCC)*, Vol. LNCS 2034, 1998.
- [2] Y. V. Shornikov, V. S. Druzhinin, N. A. Makarov, K. V. Omelchenko and I. N. Tomilov, "Instrumental Tools of Computerized Analysis (ISMA)," Official registration license for computers No. 2005610126, Moscow, Rospatent, 2005.

- [3] D. Harel, "Statecharts: A Visual Formalism for Complex Systems," *Science of Computer Programming Programming*, Vol. 8, 1987, pp. 231-274.
- [4] E. A. Novikov, "Explicit Methods for Stiff Systems," Novosibirsk: Nauka. Sib. Enterpr. The Russian Academy of Sciences, 1997.
- [5] A. E. Novikov, E. A. Novikov, Y. V. Shornikov and D. N. Dostovalov, "Numerical Simulation of Hybrid Sys-

tems with an Explicit Method of Order in the ISMA Instrumental Environment," *Problems of Informatics*, Vol.3, 2010, pp. 73-80.

[6] K. J. Astrom and Canudas-de-Wit C. "Revisiting the LuGre Model. Stick-slip Motion and Rate Dependence," *IEEE Control Systems Magazine*, June 2008, No. 28, pp. 101-114.