From: AAAI-98 Proceedings. Copyright © 1998, AAAI (www.aaai.org). All rights reserved.

An Ontology for Transitions in Physical Dynamic Systems*

Pieter J. Mosterman
Inst. of Robotics and System Dynamics
DLR Oberpfaffenhofen
P.O. Box 1116
D-82230 Wessling
Pieter.J.Mosterman@dlr.de

Feng Zhao
Department of Computer and Information Science
The Ohio State University
Columbus, OH 43210
fz@cis.ohio-state.edu

Gautam Biswas
Department of Computer Science
Box 1679, Sta B
Vanderbilt University
Nashville, TN 37235
biswas@vuse.vanderbilt.edu

Abstract

Physical systems often exhibit complex nonlinear behaviors in continuous time at multiple temporal and spatial scales. Abstractions simplify behavioral analysis and help focus on dominant system behaviors by defining sets of equivalent behavior types called *modes*. System behavior evolves in continuous modes with discrete transitions between modes. Subtle interactions between the continuous behaviors and discrete transitions need to be captured by well-defined hybrid modeling and analysis semantics. This paper presents a taxonomy of transition modes, and develops a formal semantics for transition conditions that lead to efficient and physically consistent simulation algorithms for physical systems.

Introduction

Physical system behaviors, governed by the principles of *conservation of energy* and *continuity of power* (Mosterman & Biswas 1998), are continuous but can operate at multiple temporal and spatial scales. When analyzing gross behavior the details of fast nonlinear changes are often insignificant. Consider the bouncing clutch in Fig. 1. With the clutch, Sw_1, in the open position, the mass m_1 applies a gravitational force causing the latch and inertia I_1 to rotate. At a predetermined angle $\theta_{contact}$, the latch collides with the fixed pinions, but the torsional elasticity in the connecting rod, I_1, results in further movement after collision before the angular velocity ω_I of the inertia reverses. For the modeler interested in overall behaviors, the collision process can be abstracted to generate an instantaneous reversal in velocity.

Hybrid modeling techniques simplify complex continuous nonlinear behaviors to piecewise continuous behaviors interspersed with discrete transitions. Fast nonlinear behavior effects are replaced by discrete transitions to alleviate numerical problems caused by the steep gradients. Model generation is simplified by eliminating parasitic parameters that are abstracted away. Discrete transitions are linked to *configuration changes* in the system model, and result in the system operating in a number of different *modes*. *Hybrid systems* are becoming increasingly popular in analyzing embedded systems (physical systems with discrete controllers) and complex physical systems that exhibit fast nonlinear behaviors (Alur *et al.* 1994; Guckenheimer & Johnson 1995; Mosterman & Biswas 1996; 1997). Hybrid system models combine continuous behaviors governed by ordinary differential equations (ODEs) or differential and algebraic equations (DAEs) with discrete transitions defined by finite state machines or Petri nets.

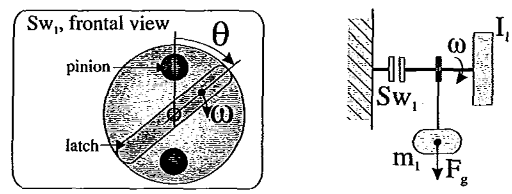

Figure 1: **Elastic collision of a braking clutch.**

Compositional modeling approaches are adopted to model discrete changes as local switching functions defined by system variables crossing threshold values. A local transition can trigger additional changes which continue till no further local transition functions are active, and the system behavior resumes continuous evolution in time. Sometimes system variables in the new continuous mode are at their threshold values, and the mode is departed in an infinitesimally small time interval resulting in a new sequence of discrete changes (Mosterman & Biswas 1997). In other situations, the system *chatters*, i.e., it exhibits quick oscillations between two modes of operation (Mosterman, Zhao, & Biswas 1997; Zhao & Utkin 1996).

We build on two existing strands of work that stud-

* Copyright ©1998, American Association of Artificial Intelligence (www.aaai.org). All rights reserved.

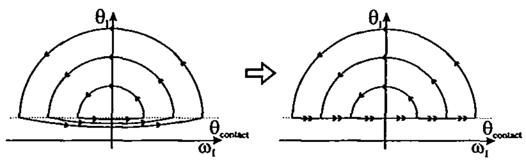

Figure 2: **Bouncing clutch phase space.**

ied individual mode transitions: (i) analysis of hybrid systems models with instantaneous mode and state vector changes (Mosterman & Biswas 1996), and (ii) analysis of hybrid system models which exhibit *chattering* (Mosterman, Zhao, & Biswas 1997; Zhao 1995; Zhao & Utkin 1996). This paper develops a taxonomy of transitions in hybrid models of complex physical systems, and a unified semantics that combines the continuous interior and boundary modes with mythical, pinnacle, and sliding behavior modes. These semantics are translated into a behavior generation algorithm, whose effectiveness is demonstrated by simulation results. The principal contribution of this work is a systematic treatment of transition behavior in hybrid systems.

Phase Space Description Formalism

The state vector of a continuous system defines an n-dimensional space called the *phase space*. Individual behaviors of the system can be described as trajectories in the phase space.

Discontinuities in Phase Space

A common approach to simplify behaviors at multiple scales is to replace a complex trajectory by piecewise continuous segments, where the system of equations describing the behavior in each segment is simpler than the original nonlinear equations. However, this introduces switching functions into the system model, and discontinuities may occur in the system variables at switching points. The phase space behavior representation of the bouncing clutch, illustrated in Fig. 2 shows the nonlinear behavior of I_1's velocity upon collision (left) being replaced by an instantaneous transition (right), indicated by the double-headed arrows.

Hybrid System Definition

Hybrid systems combine continuous and discrete behaviors (Mosterman & Biswas 1997). A hybrid model can be formally defined in terms of I, a discrete indexing set with $\alpha \in I$, defining the modes of the system. Piecewise behavior trajectories $\mathcal{F}_\alpha$ are a continuous, C^2, flow on a possibly open subset V_α of $\Re^n$, called a *chart* (Fig. 3) (Guckenheimer & Johnson 1995). The

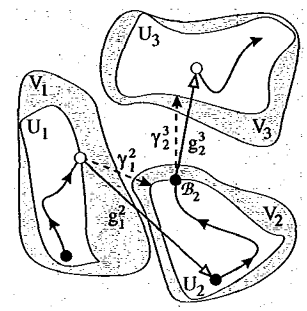

Figure 3: **A planar hybrid system.**

sub-domain of V_α where a continuous flow in time occurs is called a *patch*, $U_\alpha \subset V_\alpha$. Behavior at time t is specified by the state vector $x_\alpha(t)$, a location in chart V_α in mode α. Mode change is specified by the discrete switching function γ_α^β, a *threshold function* on V_α. If $\gamma_\alpha^\beta \leq 0$ then the system transitions from mode α to β. The change in state is defined by the mapping $g_\alpha^\beta : V_\alpha \rightarrow V_\beta$. The piecewise continuous level curves $\gamma_\alpha^\beta = 0$ define patch boundaries. If a flow $\mathcal{F}_\alpha$ includes the level curve, it contains the *boundary point*, $\mathcal{B}_\alpha$ (see patch 2 in Fig 3).

Modes of Hybrid System Behavior

Behavior discontinuities in hybrid models of physical systems have been attributed to two general abstraction techniques: (i) *time scale* and (ii) *parameter* abstraction (Mosterman & Biswas 1997; Mosterman, Zhao, & Biswas 1997). These discontinuities may manifest as jumps in system variable values and discrete switches in the fields that govern behaviors in individual modes. A formal semantics governs mode and state vector changes associated with transitions.

Hybrid Modeling of Physical Systems

Time scale abstractions model complex behaviors over small time intervals by discontinuous changes at points in time. An example is a bouncing rubber ball, where the ball velocity is modeled to reverse instantaneously upon collision with the floor. In reality, the initial kinetic energy of the ball is stored on impact as elastic energy within the ball and the floor for a very small time period, and then returned back as kinetic energy to the ball, which causes it to fly back up. Time scale abstraction reduces the process of energy storage and return to a point in time. Parameter abstractions, on the other hand, eliminate small, parasitic dissipation and storage parameters from the system model. In case of a steel ball, the elasticity coefficient of the ball may be small enough to be ignored. The implication is

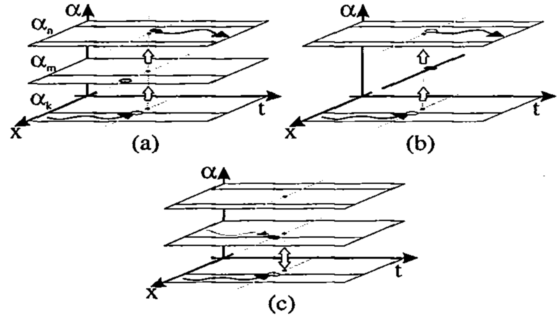

Figure 4: **(a) Mythical, (b) Pinnacle, and (c) Sliding modes.**

that the collision between the floor and ball becomes *non-elastic* (*there is no energy storage at collision*), and the ball comes to rest at the point of contact. A closer study of the two abstraction forms reveals a number of transition behaviors that require particular semantics. Fig. 4 illustrates the different transition mode behaviors (**x** represents the state vector and α represents the mode of operation).

Mythical Modes

Consider the two clutch freewheeling system (Fig. 5) where the bodies and connecting rod are assumed to be rigid (no elasticity) and small component parameters are abstracted away. The dissipation or small deformation effects in the connecting rod that are active upon collision are not modeled. Simulation results, illustrating the torsional force F_I and angular velocity ω_I appear in Fig. 6. Initially, brake Sw_1 is open and Sw_2 is closed. The weight m_1 produces an angular velocity, ω_I in inertia I_1. At 0.2 s, the rotation causes Sw_1 to close. At this point both Sw_1 and Sw_2 are closed, and ω_I is forced to 0. This causes a force in Sw_2 that moves the latch away from the pinion. Sw_2 opens, I_1 is free to rotate, and ω_I stays at a nonzero value.

The intermediate configuration, where the angular velocity was forced to 0, is mythical, i.e., it does not exist in real time. Fig. 4a illustrates a trajectory that transitions from a real mode α_k to an intermediate mythical mode α_m, and then to a real mode α_n where behavior evolves continuously. The instantaneous transitions do not affect the state vector, therefore, the angular velocity of I_1, ω_I, after the configuration changes equals its value before (Fig. 6), which is consistent with real behavior. Note that the intermediate mythical configuration defines the mode change sequence. Mythical modes also occur in hybrid models of a diode-inductor circuit and the collision of a free-

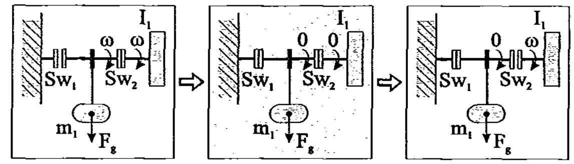

Figure 5: **Mythical mode in analyzing a freewheeling clutch.**

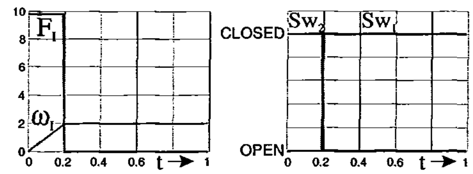

Figure 6: **A mythical configuration has no representation in real time.**

falling thin rod with the ground (Mosterman & Biswas 1997).

Pinnacles

Consider the system in Fig. 7 with significant torsional elasticity in the connecting rod. This causes a perfectly elastic collision between the latch and pinion when Sw_1 closes. The latch and I_1's rotation toward the pinions is a continuous behavior. The collision, governed by the rotational analog of Newton's elastic collision rule, satisfies $\omega_I^+ = -\epsilon\omega_I$, where ϵ represents the coefficient of restitution (= 1 for a perfectly elastic collision). This collision rule captures the torsional compression and expansion of the connecting rod on collision into a behavior at a point in time, called a pinnacle. Change in the state vector at that point in time is governed by algebraic equations, which hold only for that point. Simulation results in Fig. 8 illustrate Sw_1 closing at 0.6 s, causing momentum transfer and instantaneous reversal of angular velocity ω_I at a point in time. A pinnacle manifests as a jump in the state vector, after which the system behavior evolves in a continuous trajectory (Fig. 4b).

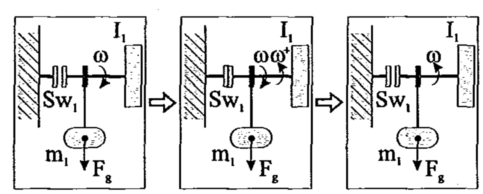

Figure 7: **A pinnacle due to an elastic collision.**

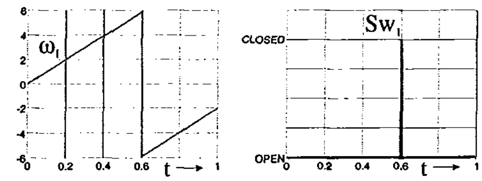

Figure 8: **Pinnacles occur at a point in time.**

Sliding Mode

Sliding mode behavior occurs when a phase space transition chatters between two modes. If a transition leads the system to the boundary region of an adjoining mode, and the direction of the field vector is toward the first mode, the system may switch from the second mode back to the boundary of the first. If the gradient of the field is again toward the second mode, the first transition may repeat. If this phenomenon continues, one observes chattering behavior (i.e., the system goes back and forth between two modes in Fig. 4c). This is best-handled by introducing sliding mode behavior on the surface that defines the boundary of the two modes.

Sliding mode behavior is illustrated for the cam-follower system in Fig. 9. The cam mechanism translates rotational motion into a linear displacement to open and close valves in the engine cylinders. Typically, a spring mechanism ensures contact between the rod and rotating cam but the high velocities of operation (up to several thousands of revolutions per minute) and wear of the spring can cause the rod to bounce on and off the cam.

When the deceleration of the cam causes the rod to disconnect, it may reconnect within an infinitesimal period of time. This chattering behavior, an artifact of the numerical time step, can slow down the simulation process. The sliding mode algorithm replaces chattering by *equivalent dynamics* to derive the nonlinear behavior from the linearized phase space under the assumption of small physical inertial and hysteresis effects (Mosterman, Zhao, & Biswas 1997). Fig. 10 shows the simulation behavior of a cam-follower mechanism. The simulation results on the left do not apply equivalent dynamics. The cam and rod alternately have equal and nonequal velocities. When the rod disconnects from the cam the velocity difference builds up. However, because the cam decelerates, at the next simulation time step a nonelastic collision occurs and the rod and cam velocities are instantaneously forced to equal values. The simulation on the right applies equivalent dynamics to remove this simulation artifact. The system slides on the switching surface $v_{rod} = v_{cam}$ and there is no error due to chattering. This conforms

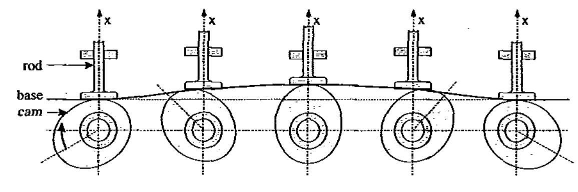

Figure 9: **A cam mechanism opens a valve.**

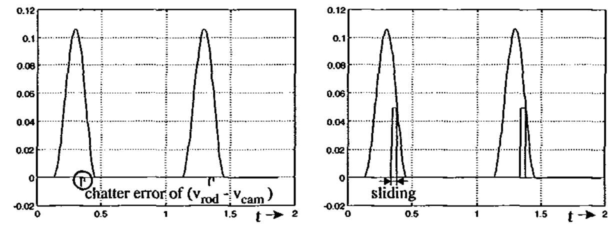

Figure 10: **Sliding mode simulation during an interval of time.**

with true physical behavior, where unmodeled higher order physical phenomena such as adhesive forces between the rod and cam would result in the rod and cam having the same velocity.

An Ontology for Transitions

Starting from the three modes defined above, a formal characterization of mode transitions can be derived by focusing on the mechanisms active during the transition process. This is best derived from a mathematical hybrid system model. An implemented simulation algorithm applies the mode taxonomy to invoke the correct semantics for generating physically consistent behaviors.

The Mathematical Model

The mathematical model defines a switching function, γ_α^β, with parameters the state vector x_α, prior to the jump and x_α^+, the state vector immediately after the jump. The semantics of transitions is specified by the recursive relation between γ_α^β and g_α^β

$$\begin{cases} x^+_{\alpha_k} = g^{\alpha_i}_{\alpha_k}(x_{\alpha_k}) \\ \gamma^{\alpha_{i+1}}_{\alpha_i}(x_{\alpha_k}, x^+_{\alpha_k}) \leq 0 \end{cases} \qquad (1)$$

Note the α_k subscript of x_{α_k} in $g^{\alpha_i}_{\alpha_k}$. In physical systems, continuous behavior is completely specified by the state. Therefore, the state mapping is independent of the departed mode, i.e., g_α^β is independent of α. This results in the general sequence

$$\underbrace{\begin{cases} x^+ = g^{\alpha_1}(x) \\ x = x^+ \\ \dot{x} = f_{\alpha_1}(x,t) \end{cases}}_{\alpha_1} \overset{\gamma^{\alpha_2}_{\alpha_1}(x,x^+)}{\longrightarrow} \underbrace{\begin{cases} x^+ = g^{\alpha_2}(x) \\ x = x^+ \\ \dot{x} = f_{\alpha_2}(x,t) \end{cases}}_{\alpha_2} \overset{\gamma^{\alpha_3}_{\alpha_2}(x,x^+)}{\longrightarrow}$$

$$\cdots \xrightarrow{\gamma^{\alpha_m}_{\alpha_{m-1}}(x,x^+)} \underbrace{\begin{cases} x^+ = g^{\alpha_m}(x) \\ x = x^+ \\ \dot{x} = f_{\alpha_m}(x,t) \end{cases}}_{\alpha_m} \quad (2)$$

In this sequence, each mode, α, may be departed when any of the three assignment statements is executed. The resultant computational model, illustrated in Fig. 11, distinguishes the three cases.

(a) *Transition to mythical mode* (Fig. 11a): This occurs when $x^+ = g^{\alpha_i}(x)$ leads to $\gamma^{\alpha_{i+1}}_{\alpha_i}(x,x^+) \leq 0$. The immediate transition bypasses the *integrator* ($\int$), therefore, the state vector x remains unchanged through the transition (also see Fig. 4a).

(b) *Transition to pinnacle* (Fig. 11b): This occurs when $x = x^+$ results in $\gamma^{\alpha_{i+1}}_{\alpha_i}(x,x^+) \leq 0$. Updating state vector x causes a mode transition. Therefore, mode α_i only exists at a point in time but the state vector can change with the transition (Fig. 4b).

(c) *Transition to continuous mode* (Fig. 11c): In this case after the transition and update $x = x^+$, $\forall_{\alpha_n} \gamma^{\alpha_n}_{\alpha_m}(x,x^+) > 0$. Therefore, f_{α_m} is active. Three situations may occur:

   - *Interior mode:* Behavior evolution is continuously governed by a field, f_{α_m}. In Fig. 12a the system transitions from one continuous mode α_k to a second continuous mode α_m.
   - *Boundary:* A transition occurs after an infinitesimal period of time, which indicates a patch boundary was reached (see Fig. 12b), and the newly established mode switches to another mode within an infinitesimal period of time.
   - *Sliding mode:* A transition occurs after an infinitesimal period of time and the newly established mode switches back to the current one within an infinitesimal period of time (Fig. 4c).

Pinnacles and continuous modes are referred to as *real* modes because they change the state vector x stored in the integrator ($\int$). Interior modes, boundary modes and sliding modes are *continuous* because the gradient of the field vector defines the behavior evolution process. Note that there is a distinct difference between the pinnacle and boundary modes. Pinnacles, caused by time scale abstraction, define behaviors by an algebraic equation that causes a jump in phase space. As soon as the *a priori* state vector is updated, the pinnacle is departed. Boundary behaviors are governed by gradients of continuous flow. After the state vector is updated, the boundary is active. It is departed after behavior evolves over an infinitesimal

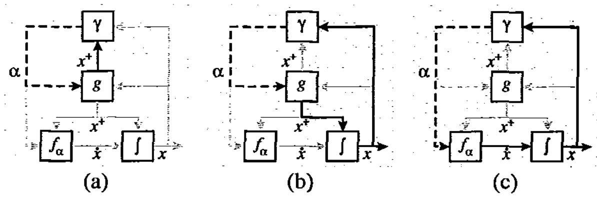

Figure 11: **Classes of modes of operation.**

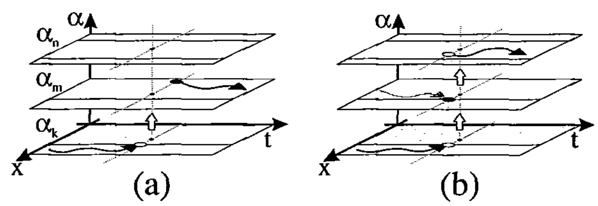

Figure 12: **(a) Interior mode and (b) boundary mode.**

amount of time along the field gradient (Mosterman & Biswas 1997).

The described mode transitions may appear in combination with one another. For example, in the cam-follower system, collision effects occur between the cam and the pushing rod that opens valves. These collisions introduce pinnacles in phase space that are traversed in between sliding modes.

The Simulation Algorithm

Simulation of hybrid models requires special semantics for mythical, pinnacles, and sliding modes. Starting from an interior mode α_k and a transition from α_k to α_m with $x_{\alpha_k}(t)^+ = g^{\alpha_m}(x_{\alpha_k})$, Table 1 specifies the conditions that have to be satisfied for each of these modes.[1] A mythical mode is detected when the new state vector x^+ is beyond the patch of the newly inferred mode. This requires an instantaneous transition governed by invariance of state (see Table 2) (Mosterman & Biswas 1997). For pinnacles, algebraic relations govern system behavior. No continuous behavior evolution occurs. In a new mode for which continuous evolution is specified, an immediate transition may occur when the system is advanced over an infinitesimal time interval. This implies that the transition moved the system onto a boundary point instead of the interior of a patch (Fig. 12). If repeated transitions occur between two modes chattering behavior governed by equivalence dynamics is observed (Mosterman, Zhao, & Biswas 1997).

[1]The function $\gamma^{\beta}_{\alpha}(x,x^+)$ is replaced for clarity reasons by $\gamma^{\beta}_{\alpha}(x^+)$ for sliding modes because $x^+ = x$.

Mode Class	Criteria
mythical mode	$\exists \alpha_n (\gamma_{\alpha_m}^{\alpha_n}(x_{\alpha_k}(t), x_{\alpha_k}(t)^+) \leq 0$
pinnacle	$\exists \alpha_n (\gamma_{\alpha_m}^{\alpha_n}(x_{\alpha_k}(t)^+, x_{\alpha_k}(t)^+) \leq 0$
sliding mode	$\exists \delta t_1 (\delta t_1 < \epsilon)(\gamma_{\alpha_m}^{\alpha_k}(x_{\alpha_m}(t+\delta t)) \leq 0) \wedge$ $\exists \delta t_2 (\delta t_2 < \epsilon)(\gamma_{\alpha_k}^{\alpha_m}(x_{\alpha_k}(t+\delta t)) \leq 0)$

Table 1: **Classification scheme and guards.**

Mode Class	Semantics
mythical mode	invariance of state
pinnacle	no continuous evolution
sliding mode	equivalence of dynamics

Table 2: **Semantics governing particular mode transition behavior.**

A high level description of the simulation algorithm appears as Algorithm 1. The input is the mathematical hybrid system model, and the output a behavior trajectory that includes mode transitions. A forward Euler numerical approximation function, *timeStep*(α, x) evolves behavior along field gradients. When a transition condition occurs ($\gamma_\alpha^\beta \leq 0$), the function *recursion*(α, x), which implements Eq. (1) is invoked. When recursion terminates, the state vector is updated ($x = x^+$). This may cause a further change implying a pinnacle. The pinnacle may be followed by mythical modes. When mode changes terminate in a new continuous mode, the sliding mode condition in Table 1 is checked by the function *slide*(α, x). If satisfied, equivalence dynamics approximates system behavior until behavior moves away from the switching surface. The system continues to evolve until a new transition condition is detected. Applications to the braking clutch and cam-follower system were illustrated earlier.

Algorithm 1 Hybrid Simulation Algorithm

```
Require: α, x, f_α, γ_α^β, g_α^β
  while time < end time do
    x = timeStep(α, x)
    [α+, x+] = recursion(α, x)
    if α+ ≠ α then
      repeat
        α = α+
        x = x+
        [α+, x+] = recursion(α, x)
      until α+ = α
      [α, x] = slide(α, x)
    end if
  end while
```

Conclusions

A systematic study of abstractions provides a formal methodology for hybrid modeling of physical systems. The models operate in multiple piecewise continuous regions represented by patches in phase space. Transitions between patches give rise to modes of behavior classified as: (i) mythical, (ii) pinnacle, (iii) interior, (iv) boundary, and (v) sliding. Formal definitions for transitions between modes are developed that allow the specification of self-consistent simulators for hybrid systems. Our simulator has been applied to generate behaviors for a number of physical examples. It handles the idiosyncrasies of each transition type as well as combinations of transitions well. Future research will focus on applying this framework to verification problems in control. As phase space dimensions increase, verifying the sliding mode guards becomes a more challenging task.

Acknowledgements: Pieter Mosterman is supported by a grant from the DFG Schwerpunktprogramm KONDISK. Feng Zhao is supported in part by ONR YI grant N00014-97-1-0599 and NSF YI Award CCR-9457802, and a grant from Xerox PARC to Ohio State Univ. Gautam Biswas is supported by grants from PNC, Japan and Hewlett-Packard, Co.

References

Alur, R.; et al., 1994. The algorithmic analysis of hybrid systems. In *Proc. of the 11th Intl. Conf. on Analysis and Optimization of Discrete Event Systems*, 331–351.

Guckenheimer, J., and Johnson, S. 1995. Planar hybrid systems. In *Hybrid Systems II*, vol. 999, 202–225.

Mosterman, P. J., and Biswas, G. 1996. A Formal Hybrid Modeling Scheme for Handling Discontinuities in Physical System Models. In *AAAI-96*, 985–990.

Mosterman, P. J., and Biswas, G. Formal Specifications for Hybrid Dynamical Systems. In *IJCAI-97*, 568–573.

Mosterman, P. J., and Biswas, G. 1998. A theory of discontinuities in dynamic physical systems. *Journal of the Franklin Institute* 335B(6):401–439.

Mosterman, P. J.; Zhao, F.; and Biswas, G. 1997. Model semantics and simulation for hybrid systems operating in sliding regimes. In *AAAI Fall Symp. on Model Directed Autonomous Systems*, 48–55.

Zhao, F., and Utkin, V. I. 1996. Adaptive simulation and control of variable-structure control systems in sliding regimes. *Automatica: IFAC Journal* 32(7):1037–1042.

Zhao, F. 1995. Qualitative reasoning about discontinuous control systems. In *Proc. of IJCAI-95 Engineering Problems for Qualitative Reasoning Workshop*, 83–93.