CDC01-REG1422

Modeling and Verification of Hybrid Systems with Clocked and Unclocked Events

B. Izaias Silva and Bruce H. Krogh

Dept. of Electrical and Computer Eng., Carnegie Mellon University,

Pittsburgh, PA 15213 (USA), {izaias,krogh}@cmu.edu

Submitted to

The 40th Conference on Decision and Control

Abstract

We define an extension of hybrid automata, called *sampled-data hybrid automata* (SDHA), to model systems with clocked and unclocked events. Although clocked events can be modeled in standard hybrid automata by introducing additional continuous state variables, we show in this paper how the SDHA formalism provides a basis for more efficient computational methods for verification. Towards this end, we define a transition system semantics for the SDHA and present a method for computing approximate quotient transition systems using an extension of existing tools for computing reachable sets for hybrid systems. The definitions and the elements of a model checking procedure for SDHAs are illustrated with an example.

I. Introduction

It is often observed that many computer-controlled processes can be modeled as hybrid systems, where continuous state variables model the physical process (the *plant*) and discrete state variables model the control logic (the *controller*) [12]. Hybrid automata were introduced as an extension to standard discrete-state automata to model such systems [1]. In a hybrid automaton, continuous state equations are associated with each discrete state and discrete-state transitions occur when the continuous trajectories satisfy associated predicates called *guards*.

In hybrid automata models of computer-controlled processes, the guards represent switching conditions in the control logic based on measurements from the plant. In actual systems, the measurements and subsequent discrete control actions are usually clock-driven events, that is, the computer-controlled system is a *sampled-data system*. If the sampling frequency is fast relative to the process dynamics, it may not be necessary to model the sampling process. On the other hand, as the sampling period approaches the time constants inherent in the process dynamics, a hybrid automaton that neglects sampling may not correctly model the system behavior.

In this paper, we present an extension of hybrid automata called *sampled-data hybrid automata* (SDHA) to model explicitly discrete transitions that are based on clock-driven sampling of the continuous state. SDHA are standard hybrid automata plus a *clock structure* that specifies bounds on the clock phase, variations in the clock period, and sampling jitter at each sampling instant. The discrete-state transitions in SDHA can be either *unclocked events* (standard hybrid

automaton events), or *clocked events*, which means that the associated guards are evaluated only at sampling times that are consistent with the specified clock structure.

The purpose of defining SDHA is threefold. First, SDHA provide a convenient and intuitive formalism for representing sampled-data hybrid systems and the types of variations in sampling events that can occur in real systems (phase, period, and jitter). To model clocked events in a standard hybrid automaton, an additional continuous-state variable (integrator) must be introduced as the clock and its value needs to be included appropriately in the guards where sampling occurs. This increases the order of the continuous-state dynamics. Clock uncertainty due to phase, period, and jitter variations would introduce further complexity in the model. In contrast, only the clock structure is needed to model clocked events in the SDHA.

The second reason for introducing the SDHA is to avoid recording information when nothing occurs except for sampling. In a hybrid system model that includes a clock, all sampling times are recorded as events in the transition system defined for verification and analysis. For the SDHA, we introduce a formal transition semantics in which states are defined at sampling times only if a clocked event occurs. In other words, the clock events (at sampling times) are not represented explicitly in the transition semantics. This results in an significant reduction (possibly orders of magnitude) in the number of states and transitions the model constructed for analysis and verification of the SDHA relative to the equivalent hybrid automaton model.

The third purpose of the SDHA is to provide the basis for more efficient computational methods to verify properties of sampled-data hybrid systems. A model checking procedure is presented for SDHA based on the computation of *approximate quotient transition systems* (AQTS) using existing computational tools for hybrid automata [11]. Efficiency is obtained, relative to model checking for the equivalent hybrid automaton, by: (i) avoiding the computations required at each transition for the insignificant sampling instant (that is, the clock events for which nothing can happen because no guard is enabled); and (ii) realizing a much less conservative AQTS due to the elimination of the approximations that would occur if the sets of reachable states were recorded at every sampling instant.

Sampled-data hybrid automata (SDHA) were first introduced in [2] with only clocked events. In this paper we extend this model to include unclocked events as well. This extension is useful to model applications where some control actions are based on sampling at a rate that cannot be neglected, mixed with some events that are not synchronized with a clock (such as discrete-state transitions in the plant dynamics) or are synchronized with a higher-frequency clock that can be neglected (such as independent safety controllers that monitor the system locally and take immediate action when necessary). The mixture of clocked and unclocked events introduces complexity in the transition system semantics and verification procedure that is not present in the original definition of the SDHA in [2].

We note that another alternative to modeling sampled-data systems is to use discrete-time models of the continuous dynamics (see, e.g., [3],[4],[5],[6]). The difficulties with this approach are: (i) discrete-time state-transition functions cannot be obtained for arbitrary continuous dynamic systems; (ii) the effects of variations in the sampling times are not modeled easily; and (iii) unclocked events cannot be modeled. For these reasons we retain continuous-time differential equations, rather than difference equations, in the SDHA.

The following section introduces notation and presents the definition of SDHA. The definition of the SDHA is illus-

trated with an example in section III, which is used throughout the remainder of the paper to illustrate the concepts. Section IV introduces the transition system semantics for SDHA and section V describes the procedure for computing approximate quotient transition systems to verify properties of SDHA. The concluding section summarizes the contributions of this paper and discusses the status of SDHA extensions to the hybrid system model checker CheckMate [9].

II. Sampled-Data Hybrid Automata (SDHA)

Sampled-data hybrid automata (SDHA) are hybrid automata in which the set of events E (discrete-state transitions) is partitioned into *clocked events*, E_c and unclocked events, E_u. The elements of the hybrid automaton are defined as follows.

Definition 1: A *hybrid automaton with clock and unclocked events* is a tuple $H_c = (X, U, X_0, U_0, F, D, E_c, E_u, L, G, J)$, where

- $X = \Re^n$ is the set of *continuous states*;
- U is a finite set of *discrete states*;
- $X_0 \times U_0 \subseteq X \times U$ is the set of *initial states*;
- $F = \{f_u : X \to X\}_{u \in U}$ is the set of *flows* defining the continuous state equation $\dot{x} = f_u(x)$ for each discrete state $u \in U$;
- $D \subseteq \{U \times U - \{(u,u) | u \in U\}\}$ is the set of *discrete-state transitions*;
- E_c is the finite set of *clocked events*;
- E_u is the finite set *unclocked events*;
- $L : D \to 2^E$, where $E = E_c \cup E_u$, is the *event labeling function* associating events with discrete-state transitions;
- $G : E \to 2^X$ defines the *guard* for each event; and
- $J : E \times X \to 2^X$ is the *jump relation* that associates with each event and continuous state the set of possible new initial state values assigned when the event occurs.

Aside from the introduction of the clocked and unclocked events, the elements of the hybrid automata are standard, with differential equations for the continuous state evolution (rather than differential inclusions). Given a discrete state $u \in U$ with flow f_u, we will denote the continuous state trajectory starting $x_0 \in X$ by $\zeta_{(u,x_0)}(.)$, where $\dot{\zeta}_{u,x_0}(t) = f_u(\zeta_{(u,x_0)}(t)), \forall\ t \geq\ 0$ and $\zeta_{u,x_0}(0) = x_0$. The definition of discrete-state transitions disallows "self-loops", that is, the value of the discrete state changes for any admissible discrete-state transition. This assumption is for technical convenience and introduces no loss of generality.

The missing standard element in above definition of the hybrid automaton is the *invariant*, typically defined for each discrete state as the subset of continuous states that are valid for continuous trajectories when the system is in that discrete state. For sampled-data systems, if the concept of the invariant were introduced, its definition would have to be modified to account for clocked events whose guards are evaluated only at sampling times. The invariant is not needed because in this paper we assume discrete-state transitions occur immediately when the guard for an associated event is satisfied.

In the following, we assume the hybrid automata is defined so that whenever an event occurs with the associated jump in the continuous state, there are no guards enabled at that instant in the new discrete state. This means that upon entering entering a discrete state $u \in U$, the continuous state is assumed to be in the *interior* of u, defined as

$$I_u = \bigcap_{e \in L(D_u)} \overline{G(e)},$$

where D_u denotes the set of out-going transitions from u and $\overline{A}$ denotes the complement of a set A.

Clocked events can occur at *sampling times*, denoted by a strictly monotone increasing sequence $\sigma = \sigma_0 \sigma_1 \cdots$. We assume the sampling process is driven by a clock that generates a sequence of *clock times* denoted by another strictly monotone increasing sequence of positive real numbers $c = c_0 c_1 \cdots$. We refer to the difference between consecutive clock times as the *clock period*, and the time of the initial clock instant as the *clock initial phase*. Clock time c_n is the instant when the n^{th} sample is requested, and the sampling time $\sigma_n \geq c_n$ indicates when the continuous state is actually sampled and the guards for the clocked events are evaluated. The difference between clock time c_n and the corresponding sampling time σ_n is referred to as sampling *jitter*. In applications, jitter represents possible variations in the time it takes to execute a sampling command, due, for example, to task scheduling on a control processor or time variations in the sampling process itself (such as analog-to-digital conversion). Jitter could be particularly significant in network-based data acquisition systems.

Based on the terminology introduced above, admissible clock and sampling sequences are specified by parameters in a *clock structure*, defined as follows.

Definition 2: A *clock structure* is a triple $C = (\Theta, \Delta, B)$ where:

- $\Theta = [\theta_m, \theta_M]$, the clock initial phase interval, with $0 \leq \theta_m \leq \theta_M$;
- $\Delta = [\delta_m, \delta_M]$, the clock period interval, with $0 \leq \delta_m \leq \delta_M$;
- $B = [\beta_m, \beta_M]$, the sampling jitter interval, with $0 \leq \beta_m \leq \beta_M$.

Since the elements of our hybrid automata are all defined independent of time, we assume without loss of generality that all runs begin at $t = 0$. The initial clock phase interval Θ specifies, that $\theta_m \leq c_0 \leq \theta_M$. The clock period interval Δ constrains consecutive clock times to satisfy $\delta_m \leq c_n - c_{n-1} \leq \delta_M$ for all $n = 1, \ldots$. Note that we assume the effects of the clock period variations accumulate from clock instant to clock instant. An alternative assumption might be that the clock period is constant, but can be within the interval Δ, in which case there would be no variation in the differences between consecutive clock times for a given run of the system.

The jitter interval B constrains durations between each clock instant c_n and its corresponding sampling instant σ_n as $\beta_m \leq \sigma_n - c_n \leq \beta_M$ for all $n = 1, \ldots$. In contrast to the clock period variations, the effects of jitter do not accumulate over time.

Combining the constraints on the period and jitter yields the following constraint on the difference between consecutive sampling times:

$$\delta_m - (\beta_M - \beta_m) \leq \sigma_n - \sigma_{n-1}, \leq \delta_M + (\beta_M - \beta_m). \tag{1}$$

Based on the above definitions and observations, the set of valid sampling sequences for a given a clock clock structure is defined as follows.

Definition 3: The set of *valid sampling sequences* for a given clock structure $C = (\Theta, \Delta, B)$, denoted Σ_C, is the set of strictly monotone increasing sequences $\sigma = \sigma_0\sigma_1 \cdots$ satisfying $\sigma_0 \in \Theta + B$ and (1) for all $n = 1, \ldots$.

Having introduced the notion of a clock structure to define the sampling process for a hybrid automaton with clocked events, the sampled-data hybrid automaton is defined as follows.

Definition 4: A *sampled-data hybrid automaton* (SDHA) a pair $S = (H_c, C)$, where H_c is a given hybrid automaton with clocked events and C is a given clock structure.

Given an SDHA $S = (H_c, C)$, the continuous and discrete states evolve in continuous time starting from an initial state in X_0, U_0, respectively, according to the elements defined for H_c and a particular sampling sequence in Σ_C. To define the valid continuous and discrete state trajectories, or *runs*, for an SDHA, we introduce the following notation. Given sequences σ, σ', $\sigma - \sigma'$ denotes the sequence constructed by removing from σ all elements of the sequence σ' that occur in the sequence σ. Given a piecewise continuous function $h : \Re^+ \to A$, where $\Re^+$ denotes the non-negative reals and A is an arbitrary set (with a defined topology), $\tau(h) = \tau_0(h)\tau_1(h) \cdots$ is the sequence of points where h is discontinuous. Note that $\tau(h)$ can be the empty sequence, a finite-length sequence, or an infinite sequence, depending on the number of discontinuities in h. For $t \in \Re^+$, $h(t^-)$ and $h(t^+)$ denote the values of the limits of h at t from the left and right, respectively.

Definition 5: A *run* for a SDHA $S = (H_c, C)$ is a triple $r = (x, u, \sigma)$, where $x : \Re^+ \to X$, $u : \Re^+ \to U$, and $\sigma \in \Sigma_C$, satisfying

1. $(x(0), u(0)) \in X_0 \times U_0$;
2. $\forall\, t \in \tau(u), (u(t^-), u(t^+)) \in D$ and $\exists\, e \in L((u(t^-), u(t^+))$ such that
   (a) if $t \notin \sigma$, $e \in E_u$
   (b) $x(t^-) \in G(e)$
   (c) $x(t^+) \in J(e, x(t^-))$
3. $\forall\ t \notin \tau(u)$
   (a) $\dot{x}(t) = f_{u(t)}(x(t))$
   (b) $\forall\ e \in L(D_{u(t^-)}) \cap E_u,\ x(t) \notin G(e)$
   (c) if $t \in \sigma$, $\forall\ e \in L(D_{(u(t^-)}) \cap E_c,\ x(t) \notin G(e)$

In words, the points in definition 5 can be interpreted as follows. Point (1) says the run starts at a valid initial state; (2) states that the discontinuity points in u are valid discrete-state transitions, where (2a) implies that transitions that do not occur at sampling times must be unclocked events; (3a) characterizes the continuous-state trajectories as satisfying the flows for each discrete state, with the added constraints that when there are no discrete-state transitions, (3b) none of the guards for unclocked events can be enabled, and (3c) at sampling times none the guards for the clocked events can be enabled. These latter constraints reflect the assumption that discrete transitions are forced whenever the guards are enabled.

III. An Example

Throughout the paper, we illustrate the definitions and results using a simple example of an ON-OFF concentration control system if fig. 1. Three tanks compose the system: tank 01 stores an alkaline ingredient, tank 02 stores an

acid ingredient, and tank 03 receives material from both tanks to form a product. the two continuous state variables of interest for this system are the level and concentration of the product in tank 3. For this system we would like to guarantee that when the batch finishes (tank3 is full, x_1=1m), the concentration in tank 3 is $6.8 \leq x_2 \leq 7.2$. The initial condition is $7.1 \leq x_2 \leq 7.2$ and $0 \leq x_1 \leq 0.1$, with a clock structure of $(\Theta, \Delta, B) = ([0, 0.1], [0.4, 0.8], [0, 0.1])$.

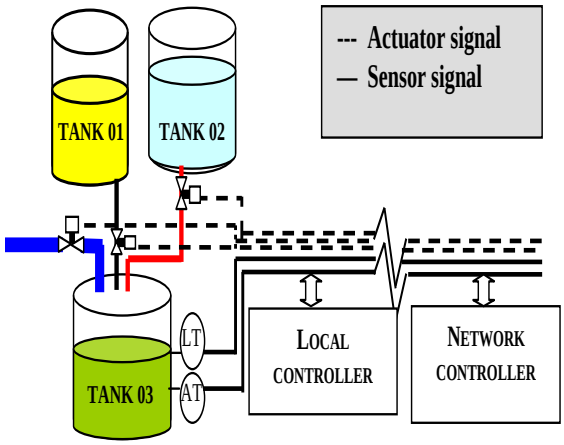

Fig. 1. pH control plant with remote and local controller

There are two controllers for the system. One is a remote controller that acts using a network to exchange data and control the behavior of the system. Since the delay through the network is non-deterministic (the jitter for our modeling) a local controller monitors instantaneously the level of concentration and takes action if the concentration levels are too high or too low (an emergency action). The controllers introduce three possible continuous dynamics $\dot{X} = (0.1, 0.1)^T$ (no correction), $\dot{X} = (0.12, -0.7)^T$ (correcting high pH) and $\dot{X} = (0.12, +0.9)^T$ (correcting low pH). This system was modeled using the hybrid system verification tool *CheckMate*. In figure 4 we show a representation of the SDHA *batch*. Figure 2 shows the trajectories starting from the vertices of the initial set, where the actual values of the clock structure are (θ, δ, β) are (0,0.4,0). The horizontal lines represent the thresholds for the clocked events $pH = 7.0$ ($normal\_pH$),$pH < 6.8$ (low_ph),$pH > 7.2$ ($high\_ph$) and for the unclocked events $pH \leq 6.6$ ($very\_low\_ph$) and $pH \geq 7.4$ ($very\_high\_ph$). The clocked events represent actions performed at sampling times. The unclocked events represent the actions of the emergency local controller. The vertical line for tank level x_1=1m, represents the stopping criterion. After filling the tank, a sample is tested. If the result is $pH > 7.2$ or $pH < 6.8$, it is considered out of specification.

In the simulation shown in fig. 3 the actual clock structure value is ([0,0.1],0.8,[0,0.1]). It shows clocked and unclocked events.

IV. Transition System Semantics for SDHA

Following the approach of Henzinger for developing computational methods to verify properties of hybrid automata [1], we now define a transition-system semantics for SDHA. For standard hybrid automata, a *discrete-trace transition system* (DTTS) can be defined as an abstraction that records the behavior of runs only when discrete-state transitions occur [11]. Similarly, for SDHA we define the *sampled-trace transition system* (STTS), which records the values of runs

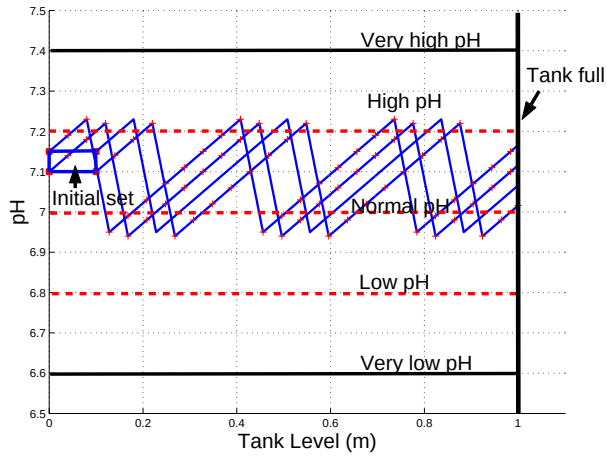

Fig. 2. simulation of initial set vertices for clock structure values (0,0.4,0)

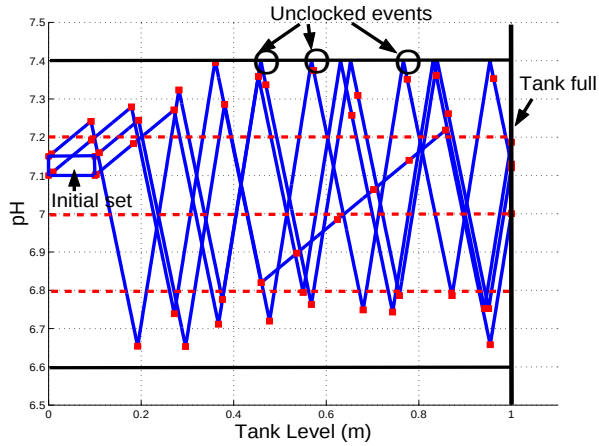

Fig. 3. simulation of initial set vertices, ([0,0.1],0.8,[0,0.1])

when clocked or unclocked events occur. The principle difference between the DTS and the STTS is that the entry states for the discrete states in the SDHA are the states that are reached either at sampling times (for the clocked transitions) or when the dynamics crosses some threshold (for the unclocked transitions), which are not necessarily on the boundaries of the guard conditions. We also include general jump conditions that were not considered in the hybrid automata model in [11]. Therefore, the entry states for discrete state transitions in SDHA models cannot be restricted to these boundaries as they are in the DTS.

The state of the STTS is defined below is a tuple of the form (x, u, ρ, w), where $x \in X$ and $u \in U$. $\rho \in \{-1, 0, 1\}$, and $w \in [0, max(\theta_M + \beta_M, \delta_M + \beta_M)]$. An initial state is of the form $(x_0, u_0, -1, 0)$, where $x_0 \in X_0$ and $u_0 \in U_0$. Subsequent states record the value of the run for the SDHA at the clocked and unclocked events. The parameter ρ indicates if the event occurs before the first sampling time ($\rho = -1$), at the first sampling time ($\rho = 0$) or after the first sampling time ($\rho = 1$). Up to the first sampling time, w records the time of the event. After the first sampling time, w records the time since the last sample for unclocked events and $w = 0$ for clocked events.

Definition 6: Given an SDHA $S = (H_c, C)$, the associated *sampled-trace transition system* (STTS) is given by $T_S =$

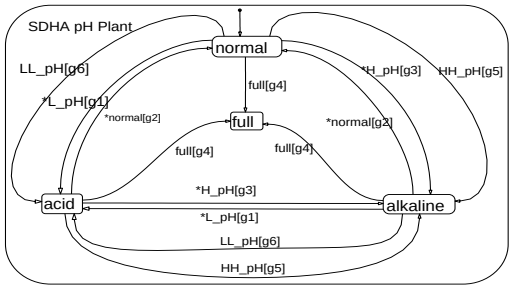

Fig. 4. SDHA pH plant ('$*$' indicates a clocked event).

$(Q_S, \rightarrow_S, Q_{S,0})$, where $Q_S = X \times U \times \{-1, 0, 1\} \times [0, max(\theta_M + \beta_M, \delta_M + \beta_M)]$, $Q_{S,0} = X_0 \times U_0 \times \{-1\} \times \{0\}$, and the transition relation $\rightarrow_S$ is defined as follows. Given two states $q = (x, u, \rho, w), q' = (x', u', \rho', w') \in Q_S$, $q \rightarrow q'$ if and only if one of the following conditions is satisfied:

1. Unclocked events before the first sample time:

$\rho = -1, \rho' = -1, 0 \leq w < w' < \theta_M + \beta_M$, and

(a) $\forall\ e \in L(D_u) \cap E_u$, $\zeta_{(u,x)}(t - w) \notin G(e)$ for $0 < t < w'$

(b) $\exists\ e \in L((u, u')) \cap E_u$ such that

i. $\zeta_{(u,x)}(w' - w) \in G(e)$

ii. $x' \in J(e, \zeta_{(u,x)}(w' - w))$

2. First sample time:

$\rho = -1, \rho' = 0, 0 \leq w < w' < \theta_M + \beta_M$, $w' \geq \theta_m + \beta_m$ and

(a) same as (1.a)

(b) if $u \neq u'$ (an event occurs at the first sampling time)

$\exists\ e \in L((u, u')) \cap E$ such that

i. same as 1.b.i

ii. same as 1.b.ii

if $u = u'$ (no event occurs at first sample)

$\forall\ e \in L(D_u), \zeta_{(u,x)}(w' - w) \notin G(e)$

3. (unclocked events after first sample)

$\rho = \{0, 1\}, \rho' = 1, 0 < w' \leq \delta_M + \beta_M$,

$\exists\ e \in L((u, u')) \cap E_u$, $\exists \sigma \in \Sigma_C$, and $N \geq 0$ such that

(a) $\zeta_{(u,x)}(w' + \sigma_N - \rho w - \sigma_0) \in G(e)$

(b) $x' \in J(e, \zeta_{(u,x)}(w' + \sigma_N - \rho w - \sigma_0))$

(c) $\forall\ \hat{u} \in U, \forall\ e \in L((u, \hat{u}) \cap E_c$, $\zeta_{(u,x)}(\sigma_n - \rho w - \sigma_0) \notin G(e)$, $n = 0, \ldots, N$

(d) $\forall\ \hat{u} \in U, \forall\ e \in L((u, \hat{u}) \cap E_u, \zeta_{(u,x)}(t - \rho w - \sigma_0) \notin G(e)$, $\sigma_0 < t < \sigma_N + w'$

4. (clocked events after first sample)

$\rho = \{0, 1\}, \rho' = 1, w' = 0$,

$\exists\ e \in L((u, u')) \cap E_c$, $\exists \sigma \in \Sigma_C$, and $N \geq 0$ such that

(a) $\zeta_{(u,x)}(\sigma_N - \rho w - \sigma_0) \in G(e)$

(b) $x' \in J(e, \zeta_{(u,x)}(\sigma_N - \rho w - \sigma_0))$

(c) $\forall\, e \in L(D_u) \cap E_c,\ \zeta_{(u,x)}(\sigma_n - \rho w - \sigma_0) \notin G(e),\ n = 0, \ldots, N-1$

(d) $\forall\, e \in L(D_u) \cap E_u, \zeta_{(u,x)}(t - \rho w) \notin G(e),\ \sigma_0 < t < \sigma_N$

The definition of the STTS above extends the transition system semantics defined [2], since it captures the clocked and unclocked behaviors. The pre-clock transitions describe the behavior of the system before the first sample takes place (this happens in the time interval $[0, \sigma_0)$. The initial-clock transition records the continuous state at the time of the first sample, $\sigma_0 = c_0 + \beta_0$, where $c_0 = \theta_0 \in \Theta$, the initial clock phase, and $\beta_0 \in B$, the initial variation in the sampling time due to jitter. Note that the initial transition can be preceded only by unclocked events. For the pH control plant, the states recorded in the STTS are illustrated for a representative trajectory in figure 5. Note that there are several sampling point (marked as '•') that are not considered states in the trajectory of the STTS. These states would be included in the DTS for a regular hybrid automata.

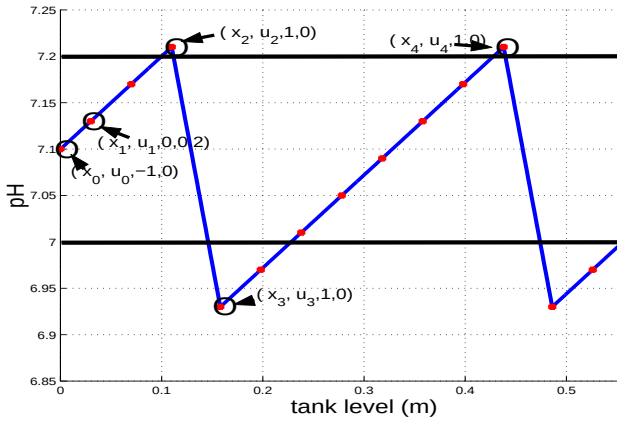

Fig. 5. A trajectory for the STTS based on the ph control plant SDHA.

The following two propositions demonstrate that the STTS provides a transition-system semantics for the SDHA in the sense that all runs of an SDHA can be mapped into state trajectories for the associated STTS (Prop. 1) and for each state trajectory for the STTS there is a corresponding run of the SDHA (Prop. 2). Proofs of the proposition are given in the appendix.

Proposition 1: Given an SDHA $S = (H_c, C)$ and its associated STTS $T_S = (Q_S, \rightarrow_S, Q_{S,0})$, if $r = (x, u, \sigma)$ is a run for S, there exists a corresponding sequence of states $q_0^r q_1^r \cdots$ that is a state trajectory for T_S.

Proposition 2: Given an SDHA $S = (H_c, C)$ and its associated STTS $T_S = (Q_S, \rightarrow_S, Q_{S,0})$, if $q = q_0 q_1 \cdots$ is a state trajectory for T_S, there exists a corresponding run $r^q = (x^q, u^q, \sigma^q)$ for S.

V. Verification of SDHA

To verify properties of SDHA a finite-state approximation of the STTS can be constructed based on a finite partition of the STTS state space. This results in the quotient transition system defined as follows.

Definition 7: **[quotient transition system]** Given a transition system $T = (Q, \rightarrow, Q_0)$, and a partition $\mathbf{P}$ of Q, the quotient transition system of T is defined as $T/\mathbf{P} = (\mathbf{P}, \rightarrow_{\mathbf{P}}, Q_0/\mathbf{P})$ where

1. $\forall P, P' \in \mathbf{P}, P \rightarrow_{\mathbf{P}} P'$ iff there exists $q \in P$ and $q' \in P'$ such that $q \rightarrow q'$;
2. $Q_0/\mathbf{P} = \{P \in \mathbf{P} | P \subseteq Q_0\}$

To create a QTS for an STTS, a partition $\mathbf{P}$ must be created. We will consider partitions with subsets of the form $P_u = (\Delta x, u, \rho, \Delta w)$, i.e, the partition of X is orthogonal to the partition of the time stamp set Ω. Separate partitions of X and Ω are required for each discrete state, $u \in U$. We denote the partition of X and Ω with respect to a discrete state (u) by P_u. Letting $\mathbf{P}$ denote the complete partition, the QTS $T_H/\mathbf{P}$ for an STTS T_H can be constructed, provided the reachability mapping could be computed for the continuous flows of the SDHA.

In general, it is not possible to represent, much less compute, the sets of reachable states required to construct the QTS for a transition system defined for a hybrid system. Nevertheless, if one can compute a conservative approximation to the reachable state mapping, an approximate QTS (AQTS) can be created that simulates the transition system and the AQTS can be used for effective verification in many cases [6]. If the specification is not satisfied for the AQTS, the approximation can be improved by refining the partition of the STTS state space. The authors presented in [2] a procedure to compute an AQTS for the case when there are only clocked events. Now we extend this procedure for the mixed case (clocked and unclocked events). In the following, we introduce an expression to compute the set of reachable states from a set $X' \subseteq X$ at a time interval $\Delta T = [T_1, T_2]$ under f_u. This set can be given by:

$$Reach^T(u, X') = \bigcup_{t \in T} \bigcup_{v \in X'} \{x \in X \mid \zeta_{(u,v)}(t) = x\}$$

$\widehat{Reach}^T(u, X')$ is called a conservative approximation procedure to compute $Reach^T(u, X')$ iff $Reach^T(u, X') \subseteq \widehat{Reach}^T(u, X')$. To assure that the reachability analysis will end, we assume that for any pair (x, u), $Reach^{[0,t]}(u, x) \cap G(e) \neq \emptyset$ for some $e \in L((u, u'))$, for some $(u, u') \in E, 0 \leq t \leq T_{max}$. In other words, T_{max} is the *time limit* for a transition to happen.

For the purpose of approximating a transition system, we define the post-condition of a set of states P, denoted $Post(P)$, as follows.

Definition 8: Given a transition system $T = (Q, \rightarrow, Q_0)$, and a set $P \subseteq Q$, the *post-condition* of P, denoted Post(P), is defined as $Post(P) = q \in Q | \exists p \in P, p \rightarrow q$

Any method M used to compute and represent post-condition sets for a transition system wil be referred as a *reachability approximation method.* We will denote the post-condition sets for a set of states P computed using method M by $Post^M(P)$. In the following definition we present a particular method M^* that will be used to construct an approximation to the STTS T_S.

Definition 9: Reachability method M^* Given a SDHA $S = (H_c, C)$ and its associated STTS $T_S = (Q_S, \rightarrow_S, Q_{S,0})$, a partition $\mathbf{P}$ of Q_S, and any given conservative procedure $\widehat{Reach}^T(.)$ to compute $Reach^T(.)$, we define the reachability method M^* as a predicate $\mathsf{M}^* : \mathbf{P} \times \mathbf{P} \rightarrow \{TRUE, FALSE\}$ as follows:

Given $P_1 = (\Delta x, u, \rho, \Delta w), P_2 = (\Delta x', u', \rho', \Delta w') \in \mathbf{P}, \mathsf{M}^*(P_1, P_2)$ is TRUE iff one of the following is true:

C1 If $\rho = \rho' = -1$ (pre-sampling case),

(a) Compute $\Delta T = [(w'_L - w_H), (w'_H - w_L)], \Delta w = [w_L, w_H], \Delta w' = [w'_L, w'_H]$ (the broadest time interval to test either the reachable set of P overlaps with P').

(b) $\exists T \in \Delta T$ such that:

i. $\exists\, e \in L((u,u')) \cap E_u$ (valid transition)

ii. $\widehat{Reach}^{[0,T+w_H-w_I]}(u,\Delta x) \bigcap G(e) = R \neq \emptyset$

iii. $J(e,R) \bigcap \Delta x' \neq \emptyset$

C2 If $\rho = -1, \rho' = 0$, (initial sampling case)

(a) Compute $\Delta T = [(w'_I - w_S), (w'_S - w_I)], \Delta w = [w_I, w_S], w' = [w'_I, w'_S]$ and $(\Delta w + \Delta T) \bigcap (\Theta + B) \neq \emptyset$. (same as in C1. The time interval has to be also valid for the first sampling time)

(b) $\exists T \in \Delta T$ such that:

i. If $u \neq u'$:

A. $\exists\, e \in L((u,u')) \cap E$

B. $\widehat{Reach}^{[0,T]}(u,\Delta x) \bigcap G(e) = R \neq \emptyset$

C. $J(e,R) \bigcap \Delta x' \neq \emptyset$

ii. If $u = u'$: (No transition happened. Special Partition because of the initial sampling time). for this case $\widehat{Reach}^{[0,T]}(u,\Delta x) \bigcap \Delta x' = R \neq \emptyset$

C3 If $\rho \neq -1, \rho' = 1$:

Compute a sequence of $N \geq 0$ time intervals $\Delta t_0\ \Delta t_1\ \ldots\ \Delta t_N$ such that $\Delta t_n = [n(\delta_m + (\beta_M - \beta_m) - \rho\, w_H, n(\delta_M + (\beta_M + \beta_m) - \rho\, w_L] = [t_n^L, t_n^H], 0 \leq\ n\ \leq\ N$ and $t_N^H \leq T_{max} \leq t_{N+1}^H$ (these are the possible sequences of sampling times before the time limit)

(a) Either $\exists\, e \in L((u,u')) \cap E_c$ and $\Delta w' = 0$ or $\exists\, e \in L((u,u')) \cap E_u$ and $\Delta w' \subseteq [0, max(\delta_M, \theta_M)]$

(b) $\widehat{Reach}^{[0,T]}(u,\Delta x) \bigcap G(e) = R \neq \emptyset$ for some $T \in (\Delta t_n + \Delta w'), 0 \leq\ n\ \leq\ N$

(c) $J(e,R) \bigcap \Delta x' \neq \emptyset$

The method M^* can be used to compute the set of reachable states from one state in a transition system where the states are elements of a partition. In figure 6, we have an illustration about the possible sequence of reachable states starting from the initial set X_0. For this part of the computation, the initial set reaches first one state when t=θ (initial sample time), even thought there is not change in the dynamics of the system.

To build the approximate quotient transition system, a reachability method needs to be *conservative*, which means that starting from a set $P \subseteq Q$, if a set $P' \subseteq Q$ is reachable in the original transition system, it will be in a AQTS. It may be the case that other states not reached in the original transition system will be in the AQTS. In def. 10 we state this concept formally.

Definition 10: [**conservative approximation**] A reachability approximation method M for a transition system $T = (Q, \rightarrow, Q_0)$, is called conservative if $\forall P \subseteq Q\ Post(P) \subseteq Post^M(P)$.

It is proved in [11] that given a transition system $T = (Q, \rightarrow, Q_0)$, a partition P of Q and a conservative method M, an AQTS T^{M}/P can be created to simulate T. In proposition 3 we prove that the reachability method M^* is conservative, therefore, one could use it to construct a AQTS for T_S provided that $\mathbf{P}$ is consistent partition of Q. See [11] for details. A procedure to compute $Post^{\mathsf{M}^*}$ is presented in figure 7.

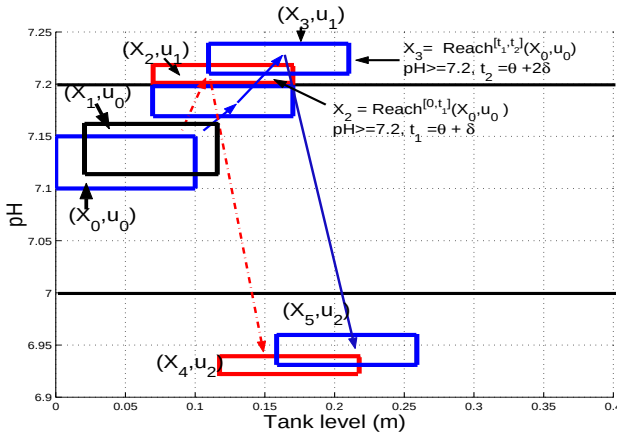

Fig. 6. Reachable sets from the initial condition X_0

Begin procedure

$Post^{\mathsf{M}^*}(P_u) := \emptyset$ % initialize $Post^{\mathsf{M}^*}$ as an empty set

For each $P'_{u'} \in \mathbf{P}$, such that $(u, u') \in D$

If $\mathsf{M}^*(P_u, P'_{u'}) == TRUE$

Add $P'_{u'}$ to $Post^{\mathsf{M}^*}(P)$

End

End

End procedure

Fig. 7. Procedure for computing $Post^{\mathsf{M}^*}$.

Proposition 3: Given an transition system $T_S = (Q_S, \rightarrow_S, Q_{S,0})$ and a partition $\mathbf{P}$ of Q_S, the reachability method function M^* presented in def. 9 is conservative, in the sense of def. 10.

VI. Discussion

This paper presents an new approach to the modeling and verification of sampled-data hybrid systems. Tools for hybrid system verification are based on methods for representing and approximating reachable sets for continuous dynamic systems. The procedure in Sec. V for constructing approximate quotient transition systems is valid for any methods for computing and representing conservative approximations to these sets.

Currently, we are implementing the verification procedure for SDHA using the computational routines in the CheckMate verification tool [9]. Future research will be directed towards efficient and effective methods for building the approximate quotient transition system using polyhedral approximations, and applying the methods presented in this paper to problems of practical interest.

Acknowledgments

The first author is supported by CNPq/Brazil under grant 200079/92. The second author is partially supported by DARPA, General Electric and Ford.

References

[1] T. Henzinger, The theory of hybrid automata, *in Proc. 11th Annual IEEE Symposium on Logic in Computer Science* (LICS 1996), pp. 278-292, 1996.

[2] B. I. Silva and B. H. Krogh, Modeling and Verification of Sampled-Data Hybrid Systems, ADPM 2000, Dortmund, Germany, September 2000.

[3] A. Bemporad and M. Morari, Verification of hybrid systems via mathematical programming, in Hybrid Systems: Computation and Control, Lecture Notes of Computer Science, vol. 1569, pp. 30- 45, Springer-Verlag, 1999.

[4] Y. Huiqum, P. Pandya, and S. Yongqiang, A calculus for hybrid sampled data systems, Formal Techniques in Real-Time and Fault-Tolerant Systems. Third International Symposium Proceedings, ProCoS, pp. 716-37, Springer-Verlag, 1994.

[5] J. Raisch and S.D. O'Young, Discrete approximation and supervisory control of continuous systems, IEEE Trans. on Automatic Control, vol. 43, no. 4, April 1998.

[6] J. Lunze, Qualitative modelling of linear dynamical systems with quantized state measurements, Automatica, vol. 30, no. 3, pp. 417-432, March 1994.

[7] A. Chutinan and B.H. Krogh, Approximate quotient transition systems for hybrid systems, *in Proc. 2000 American Control Conference*, Chicago, June 2000.

[8] T. A. Henzinger, Hybrid automata with finite bisimulations, in ICALP 95: Automata, Languages, and Programming, pp. 324-335, Springer-Verlag, 1995.

[9] B. Silva, H. Krogh, K. Richeson, A. Chutinan, Modeling and verifying hybrid dynamic systems using CheckMate, ADPM 2000, Dortmund, Germany, September 2000.

[10] B. I. Silva and B. H. Krogh, Formal verification of hybrid systems using CheckMate: a case study, *in Proc. 2000 American Control Conference*, Chicago, June 2000.

[11] A. Chutinan and B.H. Krogh, Verification of Infinite-State Dynamic Systems Using Approximate quotient transition systems, *IEEE Transactions on Automatic Control*, to appear.

[12] X.D. Koutsoukos,P.J. Antsaklis, J.A. Stiver,M.D.Lemmon, Supervisory Control of Hybrid Systems,*Proceedings of the IEEE, special Issue on Hybrid Systems: Theory and Applications*, pp 1026 - 1049, July 200 Vol. 88 N$^{\underline{0}}$ 07.

Appendix: Proofs of Propositions

Proof of proposition 1

Proof: The sequence of states $q_0^r q_1^r \cdots$ is constructed as follows. For the sequence points of discontinuities in u, $\tau(u) = \tau_0 \tau_1 \cdots$, define n_{-1} as the index of the largest τ_n less than the first sampling time σ_0, and define n_1 as the index of the smallest τ_n greater than σ_0. From these definitions construct $\{t_i\}$ as the following sequence of times: $t_n = \tau_n$ for $n = 0, \ldots, n_{-1}$, $t_{n_{-1}+1} = \sigma_0$, and $t_{n_{-1}+2+k} = \tau_{n_1+k}$ for $k = 0, 1, \ldots$. From this sequence of times, construct the following sequence of states in Q_S:

$q_0^r = (x(0), u(0), -1, 0)$

$q_{n+1}^r = (x(t_n^+), u(t_n^+), -1, t_n)$ for $n = 0, \ldots, n_{-1}$

$q_n^r = (x(t_n^+), u(t_n^+), 0, t_n)$ for $n = n_1 + 1$

for $n > n_1 + 1$:

if $t_n \in \sigma$, $q_n^r = (x(t_n^+), u(t_n^+), 1, 0)$

if $t_n \notin \sigma$, $q_n^r = (x(t_n^+), u(t_n^+), 1, t_n - \hat{\sigma}_n)$, where $\hat{\sigma}_n$ is the largest element of *sigma* less than t_n.

We now prove this sequence is a trajectory for T_S.

Initial state: By def. $(x(0), u(0)) \in X_0 \times U_0$. Therefore, $q_0^r \in X_0 \times U_0 \times [-1] \times [0]$.

Transition relation $\rightarrow_s$: Select any two consecutive states of Q_s $q_i^r = (x, u, \rho, w)$ and $q_{i+1}^r = (x', u', \rho', w')$. We have the following possibilities:

1. If $\rho = \rho' = -1$

(a) by def. $w, w' < \sigma_0$ and $w, w' \in \{t_i\}$ are consecutives, therefore $0 \leq w < w' < \sigma_0 \leq \theta_M + \beta_M$;(Item 1 of def. 6)

(b) $\forall\, t, w < t < w', t \notin \{t_i\}$($w$ and w' are consecutives); (Item 1.a)

(c) Since $t = w' \in \tau(u)$ (there is a discontinuity at time w', item 1.b) $\exists\ e \in L((u, u')) \cap E_u$ such that

i. $\zeta_{(u,x)}(w' - w) \in G(e)$

ii. $x' \in J(e, \zeta_{(u,x)}(w' - w))$

2. If $\rho = -1, \rho' = 0$

(a) Since $w = t_{n_{-1}}$ and $w' = t_{n_{-1}+1}$ (def. of $\{t_i\}$),$0 \leq w < w' = \sigma_0 \leq \theta_M + \beta_M, w' = \sigma_0 \geq \theta_m + \beta_m$; (Item 2 of def. 6).

(b) $\forall\, t, w < t < w', t \notin \{t_i\}$($w$ and w' are consecutives); (Item 2.a)

(c) Either $u \neq u'$: In this case there was a discontinuity in $w' = \sigma_0$ and since (s, σ) is a valid run $\exists\ e \in L((u, u')) \cap E$ such that (Item 2.b.i and 2.b.ii):

- $\zeta_{(u,x)}(w' - w) \in G(e)$
- $x' \in J(e, \zeta_{(u,x)}(w' - w))$

(d) or $u = u'$: Since (s, σ) is a valid run, $\forall\ e \in L(D_u)\zeta_{(u,x)}(w' - w) \notin G(e)$, otherwise there would be a discontinuity in $u(\sigma_0)$, and then $u \neq u'$ - contradiction. (Item 2 for the case $u = u'$)

3. If $\rho \neq -1, \rho' = 1$

(a) If $\rho = 0, w = \sigma_0 \in \Theta + B$; if $\rho = 1$, w was computed as an offset with respect to a valid sampling time and then $0 \leq w \leq \beta_M + \delta_M$;(Item 3 or item 4)

(b) $u \neq u'$, since by def of $\{t_i\}$ there is a discontinuity at t_k, $t_k = w' + \sigma_j$ for some t_k and σ_j

i. If $0 \leq w' \leq \delta_M + \beta_M$, $\exists\ e \in L((u, u')) \cap E_u$ (from the algorithm construction of q^r.

ii. If $w' = 0$, $\exists\ e \in L((u, u')) \cap E_c$ (from the algorithm construction of q^r.

iii. for $\sigma \in \Sigma_C$ in the def. of run r, and $N > 0$ such that:

A. By def. of valid run items 3.a, 3.b are true (4.a, 4.b for the case when $w' = 0$).

B. Since $t_{k^*} = w + \sigma_{j_1}$ and $t_{k^*+1} = w' + \sigma_{j_2}$ for some k_1, k_2, j_1, j_2 and $\sigma_{j_1}, \sigma_{j_2} \in \sigma$, there is no possible transition between t_{k^*} and t_{k^*+1} (consecutive elements in $\{t_i\}$ and items 3.c, 3.d are true (4.a, 4.b for the case when $w' = 0$).

■

Proof of proposition 2

Proof:

1. **construction of** $r^q = (x^q, u^q, \sigma^q)$

(a) Let n_θ be such that $q_{n_\theta} = (x_{n_\theta}, u_{n_\theta}, 0, w_{n_\theta})$. Construct $\tau_t = \tau_0\ \tau_1 \ldots$ a monotonic sequence as follows:

i. $\tau_i = w_i$ for $0 \leq i \leq n_\theta$

ii. τ_i is such that $x_i \in J(e, \zeta_{(u_{i-1}, x_{i-1})}(t_i - \tau_{i-1}))$, $i > n_\theta$, some $e \in L((u_{i-1}, u_i)) \cap E$

(b) Construct x^q and u^q as follows:

i. $(x^q(0), u^q(0)) = (x_0, u_0)$ where $q_0 = (x_0, u_0, -1, 0)$

ii. For $\tau_i \leq t < \tau_{i+1}$

- $x^q(t) = \zeta_{(x^q(\tau_i),u^q(\tau_i))}(t - \tau_i)$
- $x^q(\tau_{i+1}^+) = J(e, \tau_{i+1}^-)$ some $e \in L((u_{i-1}, u_i)) \cap E$
- $u^q(t) = u_i$

(c) Construct σ^r as follows:

- let $\varsigma_0 \varsigma_1 \ldots \varsigma_{N_i}$ be a subsequence of some valid sampling sequence satisfying the transition relation $q_i \rightarrow_H q_{i+1}$ and $N_i \geq 0$. Create the sequence $\alpha^i = \varsigma_1' \varsigma_2' \ldots \varsigma_{N_i}'$ such that $\varsigma_j' = t_i - \rho_i w_i + \varsigma_j - \varsigma_0$, $1 \leq 0 \leq N_i$, and $q_i = (x_i, u_i, \rho_i, w_i)$. σ^r is as follows :

- $\sigma^r_{[N_{-1},N_0]} = w_{n_0}\ \alpha^0$;
- $\sigma^r_{[N_{i-1},N_i]} = \sigma^r_{[N_{i-2},N_{i-1}]}\ \alpha^i, i \geq 0$;

2. Proof that r^q is a valid run:

(a) Proof that σ^r is a valid clock sequence (by induction):

i. $\sigma^r_0 = w_{n_\theta}$. By def. $w_{n_\theta} \in \Theta + B$

ii. Suppose $\sigma^r_{[N_{i-1},N_i]}$ is a subsequence of valid clock sequence for some $i \geq 0$. We have to prove that $\sigma^r_{[N_i,N_{i+1}]} = \sigma^r_{[N_{i-1},N_i]}\ \alpha^{i+1}$ is also a subsequence of valid clock sequence. Since α^{i+1} is a subsequence of a valid clock sequence, we only have to prove that the time interval between the last element of $\sigma^r_{[N_{i-1},N_i]}(\varsigma_{N_i}')$ and the first element of $\alpha^{i+1}(\varsigma_{1_{i+1}}')$ satisfies $\delta_m - (\beta_M - \beta_m) \leq \varsigma_{N_i}' - \varsigma_{1_{i+1}}' \leq \delta_M + (\beta_M - \beta_m)$. The next expressions are obtained from the definitions of $\sigma^r_{[N_{i-1},N_i]}, \alpha^{i+1}$ and x_{i+1}:

- $\varsigma_{N_i}' = t_i - \rho_i w_i + \varsigma_{N_i} - \varsigma_{0_i}$.
- $\varsigma_{1_{i+1}}' = t_{i+1} - w_{i+1} + \varsigma_{1_{i+1}} - \varsigma_{0_{i+1}}$.($\rho_{i+1}' = 1$, since q_{i+1} is after the first sampling time)
- $x_{i+1} \in J(e, \zeta_{(u_i,x_i)}(\varsigma_{N_i} + w_{i+1} - \rho_i w_i - \varsigma_{0_i}))$ for some $e \in L((u_i, u_{i+1})) \cap E$.
- Also $x_{i+1} = J(e, \zeta_{(u_i,x_i)}(t_{i+1} - t_i))$. Then a sufficient condition is that $\varsigma_{N_i} + w_{i+1} - \rho_i w_i - \varsigma_{0_i} = t_{i+1} - t_i$.
- $\varsigma_{1_{i+1}}' - \varsigma_{N_i}' = (t_{i+1} - w_{i+1} + \varsigma_{1_{i+1}} - \varsigma_{0_{i+1}}) - (t_i - \rho_i w_i + \varsigma_{N_i} - \varsigma_{0_i}) = t_{i+1} - t_i - w_{i+1} + \varsigma_{1_{i+1}} + \rho_i w_i - \varsigma_{N_i} - \varsigma_{0_{i+1}} + \varsigma_{0_i}$
- Finally $\varsigma_{1_{i+1}}' - \varsigma_{N_i}' = \varsigma_{1_{i+1}} - \varsigma_{0_{i+1}}$
- Since ς is valid sampling sequence: $\delta_m - (\beta_M - \beta_m) \leq \varsigma_{1_{i+1}} - \varsigma_{0_{i+1}} \leq \delta_M + (\beta_M - \beta_m)$

(b) Proof that x^q and u^q follow the requirements for a run

i. By def. of STTS $x^q(0) = x_0 \in X_0, u^q(0) = u_0 \in U_0$ (Item 1 of def. 5);

ii. by def. of STTS, suppose $t_i \in \tau_t$:

- If $0 < i \neq n_\theta$: $(u^q(t_i^-), u^q(t_i^+)) \in D$ and $\exists\ e \in L(u^q(t_i^-), u^q(t_i^+)) \cap E$;(First part of item 2 of def 5)
- If $i = n_\theta$; either $(u^q(w_{n_\theta}^-), u^q(w_{n_\theta}^+)) \in D$ and $t_{n_\theta} \in \tau_t$(and previous conditions also hold) or $u^q(w_{n_\theta}^-) = u^q(w_{n_\theta}^+)$ and t_{n_θ} is not a discontinuity point (First part of item 2 of def 5);
- Suppose $t_i \notin \sigma$, but $e \in E_c$. According to the algorithm to construct x^q and u^q, the state that relates to t_i is $q_i = (x^q(t_i^+), u^q(t_i^+), \rho, 0)$. Since w=0, $t_i \in \sigma$ — **contradiction**. So if $t_i \notin \sigma$, $e \in E_u$. (item 2a)
- By def. of x^q and u^q, $x^q(t_i^-) \in G(e)$ (Item 2b)
- By def. of x^q and u^q, $x^q(t_i^+) \in J(e, x^q(t_i^-))$ (Item 2c)

iii. suppose $t \notin \tau(t)$

- $\dot{x}^q(t) = f_{u^q(t)}(x^q(t))$, otherwise, there would be a discontinuity in $u^q(t)$, and $t \in \tau(t)$ - **contradiction**. (Item 3a)

- By def. of $\tau(t), \forall\ e \in L(D_{u^q(t^-)}) \cap E_u$ we have $x^q(t^-) \notin G(e)$, otherwise, t would belong to $\tau(t)$ (item 3b).
- suppose $t \in \sigma$ and $\exists\, u' \in U$, such that $(u^q(t^-), u') \in D$, and $\exists\ \ e \in L((u^q(t^-), u') \cap E_c$ such that $x^q(t^-) \in G(e)$, it means that should be a discontinuity in $u^q(t)$, and $t \in \tau_t$ - **contradiction**. (item 3c)

■

Proof of proposition 3

Proof:

To prove that M* is a conservative approximation, we have to show that $P_2 \in Post^{M^*}(P_1)$. Assume the following:

- $P_2 \in Post(P1)$, for $P_1 = (\Delta x, u, \rho, \Delta w), P_2 = (\Delta x', u', \rho', \Delta w')$
- Also, suppose that, $p_1 = (x, u, \rho, w) \rightarrow_S\ p_2 = (x', u', \rho', w')$, for $p_1 \in P_1$,$p_2 \in P_2$.

According to the definition of $Post^{M^*}(P_1)$ in figure 7, it is enough to prove that $M^*(P_1, P_2)$ =TRUE, since the algorithm in fig. 7 will add P_2 to $Post^{M^*}(P_1)$. the proof is as follows:

1. If $\rho = \rho' = -1$ (condition C1 of 9)

(a) assume $T = w' - w$. Clearly $T \in [(w'_I - w_S), (w'_S - w_I)]$

(b) $\exists\, e \in L((u, u')) \cap E_u$ (C1.b.i)

(c) since $\zeta_{(u,x)}(T) \in G(u, u')$ and also $\zeta_{(u,x)}(w'-w) \subseteq Reach^{[0,T]}(u, \Delta x) \subseteq \widehat{Reach}^{[0,T]}(u, \Delta x)$ we have that $\widehat{Reach}^{[0,T]}(u, \Delta x) \cap G(e) = R \neq \emptyset$. (C1.iii)

(d) Since $\zeta_{(u,x)}(w' - w) \in R$, $x' \in J(e, \zeta_{(u,x)}(w' - w)$ and $x' \in \Delta x'$ we have that $J(e, R) \cap \Delta x' \neq \emptyset$ (C1.iii)

2. If $\rho = -1 \rho' = 0$ (condition C2 of 9)

(a) assume $T = w' - w$ clearly $T \in [(w'_I - w_S), (w'_S - w_I)]$ and also $\theta_m + \beta_m \leq T + w$

(b) If $u \neq u, \exists\, e \in L((u, u')) \cap E$ (C2.b.i.A)

i. since $\zeta_{(u,x)}(T) \in G(u, u')$ and $\zeta_{(u,x)}(w'-w) \subseteq Reach^{[0,T]}(u, \Delta x) \subseteq \widehat{Reach}^{[0,T]}(u, \Delta x)$ we have that $\widehat{Reach}^{[0,T]}(u, \Delta x) \cap G(e) = R \neq \emptyset$. (C2.b.i.B)

ii. Since $\zeta_{(u,x)}(w' - w) \in R$, $x' \in J(e, \zeta_{(u,x)}(w' - w)$ and $x' \in \Delta x'$ we have that $J(e, R) \cap \Delta x' \neq \emptyset$ (C2.b.i.C)

(c) If $u = u'$

i. since $\zeta_{(u,x)}(T) = x'$ and $\zeta_{(u,x)}(w' - w) \subseteq Reach^{[0,T]}(u, \Delta x) \subseteq \widehat{Reach}^{[0,T]}(u, \Delta x)$ we have that $\widehat{Reach}^{[0,T]}(u, \Delta x) \cap \Delta x' = R \neq \emptyset$. (C2.b.ii)

C3 If $\rho \neq -1, \rho' = 1$:

Consider $\sigma = \sigma_0 \sigma_1 \ldots \sigma_N$ a subsequence of some valid clock sequence satisfying $p_1 \rightarrow p_2$:

(a) By def. of STTS, If $\exists\, e \in L((u, u')) \cap E_c$, $w' = 0$. It implies that $\Delta w' = 0$. In the same way, if $e \in L((u, u')) \cap E_u$, $w' \subseteq [0, max(\delta_M, \theta_M)]$. It implies that $\Delta w' \subseteq [0, max(\delta_M, \theta_M)]$. (C3.a)

(b) Assume $T = \sigma_N - \rho w + w'$

(c) Since $\zeta_{(u,x)}(T) \in G(u, u')$ and $\zeta_{(u,x)}(T) \subseteq Reach^{[0,T]}(u, \Delta x) \subseteq \widehat{Reach}^{[0,T]}(u, \Delta x)$ we have that $\widehat{Reach}^{[0,T]}(u, \Delta x) \cap G(e) = R \neq \emptyset$.(C3.b)

(d) Since $\zeta_{(u,x)}(w' - w) \in R$, $x' \in J(e, \zeta_{(u,x)}(w' - w)$ and $x' \in \Delta x'$ we have that $J(e, R) \cap \Delta x' \neq \emptyset$ (C3.c)

■