# Undecidability of Quantized State Feedback Control for Discrete Time Linear Hybrid Systems

Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci
Computer Science Department, Sapienza University of Rome, Italy

**Abstract.** We show that existence of a quantized controller for a given *Discrete Time Linear Hybrid System* (DTLHS) is undecidable. This is a relevant class of controllers since *control software* always implements a quantized controller.

**Introduction** Many embedded systems are software based control systems. A software based control system consists of two main subsystems: the *controller* and the *plant*. Typically, the plant is a physical system consisting, for example, of mechanical or electrical devices while the controller consists of *control software* running on a microcontroller. In an endless loop, each $T$ seconds (sampling time), the controller, after an *Analog-to-Digital* (AD) conversion (*quantization*), reads sensor outputs from the plant and, possibly after a *Digital-to-Analog* (DA) conversion, sends commands to plant actuators. The controller selects commands in order to guarantee that the closed loop system (that is, the system consisting of both plant and controller) meets given safety and liveness properties, i.e. system level specifications. Formal verification of system level specifications for software based control systems requires modelling both continuous systems (typically, the plant) as well as discrete systems (the controller). This is typically done using *Hybrid Systems* (e.g., see [2, 1]). In [7], we presented a constructive necessary condition and a constructive sufficient condition for the existence of a (*quantized sampling*) controller for a software based control system when the plant is modelled using a *Discrete Time Linear Hybrid System* (DTLHS), that is a discrete time hybrid system whose dynamics is defined as a linear predicate (i.e., a boolean combination of linear constraints) on its continuous as well as discrete variables. System level safety as well as liveness specifications may be modelled as set of states defined in turn as linear predicates. From [5], we know that existence of a sampling controller, even for relatively simple linear hybrid automata, is undecidable. Considering that, given a *quantization schema* (i.e. number of bits used in AD conversion), the number of quantized sampling controllers is finite and that when using DTLHSs also the plant is modelled using a discrete model of time, one may be led to think that existence of a quantized sampling controller might be decidable. In this paper we show that also for DTLHSs, existence of a quantized sampling controller meeting given specifications is undecidable. We prove such a result by showing that any two-counter machine can be coded as a DTLHS thereby extending to DTLHSs the proof technique in [5]. Undecidability results of the control synthesis problem for dense as well as discrete time linear hybrid systems have been presented in [6, 5, 9, 3]. A more general problem is considered in [4], namely the discrete time control with unknown sampling rate, that is undecidable even for TA. Moreover, we note that none of the above papers addresses the issue of quantized control.

**Labeled Transition Systems** An LTS $\mathcal{S}$ is a tuple $(S, A, T)$ where $S$ is a set of *states*, $A$ is a set of *actions*, and $T : S \times A \times S \to \mathbb{B}$ is the *transition relation* of $\mathcal{S}$. $\mathcal{S}$ is *deterministic* if $\forall s \in S, a, a', a'' \in A$, $T(s, a, s')$ and $T(s, a, s'')$ imply $s' = s''$. A *run* or *path* for an LTS $\mathcal{S}$ is a sequence $\pi = s_0, a_0, s_1, a_1, \ldots$ of states $s_t$ and actions $a_t$ such that $\forall t \geq 0\ T(s_t, a_t, s_{t+1})$. The length $|\pi|$ of a finite run $\pi$ is the number of actions in $\pi$. $\pi^{(S)}(t)$ denotes the $t$-th state element of $\pi$, and $\pi^{(A)}(t)$ the $t$-th action element of $\pi$.

**Definition 1.** *A* reachability problem *is a triple $(\mathcal{S}, I, G)$, where $\mathcal{S}$ is an LTS $(S, A, T)$, and $I, G \subseteq S$. $G$ is* reachable *from $I$ if there exists a run $\pi$ of $\mathcal{S}$ such that $\pi^{(S)}(0) \in I$ and $\pi^{(S)}(t) \in G$ for some $t \in \mathbb{N}$.*

**LTS Control Problem** A *controller* for an LTS $\mathcal{S}$ is used to restrict the dynamics of $\mathcal{S}$ so that all states in the initial region will reach in one or more steps the goal region. In what follows, let $\mathcal{S} = (S, A, T)$ be an LTS, $I$, $G \subseteq S$ be, respectively, the *initial* and *goal* regions of $\mathcal{S}$.

**Definition 2.** *A* controller *for $\mathcal{S}$ is a function $K : S \times A \to \mathbb{B}$ s. t. $\forall s \in S$, $\forall a \in A$. $K(s, a) \to \exists s'\ T(s, a, s')$. The* domain *of $K$ is the set of states for which a control action is enabled, i.e.* $\mathrm{dom}(K) = \{s \in S \mid \exists a\, K(s, a)\}$. *The* closed loop system *$\mathcal{S}^{(K)}$ is the LTS $(S, A, T^{(K)})$, where $T^{(K)}\ (s, a, s') = T(s, a, s') \wedge K(s, a)$.*

A path $\pi$ is a *fullpath* if either it is infinite or its last state has no successors. We denote with $\mathrm{Path}(s, a)$ the set of fullpaths starting in state $s$ with action $a$. Given a path $\pi$ in $\mathcal{S}$, we define $j(\mathcal{S}, \pi, G)$ as follows. If there exists $n > 0$ such that $\pi^{(S)}(n) \in G$, then $j(\mathcal{S}, \pi, G) = \min\{n \mid n > 0 \wedge \pi^{(S)}(n) \in G\}$. Otherwise, $j(\mathcal{S}, \pi, G) = +\infty$. We require $n > 0$ since our systems are nonterminating and each controllable state (including a goal state) must have a path of positive length to a goal state. Taking $\sup \varnothing = +\infty$ and $\inf \varnothing = +\infty$, the *worst case distance* (pessimistic view) of a state $s$ from the goal region $G$ is $J_s(\mathcal{S}, G, s) = \sup\{j_s(\mathcal{S}, G, s, a) \mid a \in \mathrm{Adm}(\mathcal{S}, s)\}$, being $j_s(\mathcal{S}, G, s, a) = \sup\{j(\mathcal{S}, G, \pi) \mid \pi \in \mathrm{Path}(s, a)\}$. The *best case distance* (optimistic view) of a state $s$ from the goal region $G$ is $J_w(\mathcal{S}, G, s) = \sup\{j_w(\mathcal{S}, G, s, a) \mid a \in A\}$, being $j_w(\ \mathcal{S}, G, s, a) = \inf\{j(\mathcal{S}, G, \pi) \mid \pi \in \mathrm{Path}(s, a)\}$.

**Definition 3.** *A* control problem *for $\mathcal{S}$ is a triple $\mathcal{P} = (\mathcal{S}, I, G)$. A* strong *(resp.* weak*) solution to $\mathcal{P}$ is a controller $K$ for $\mathcal{S}$, such that $I \subseteq \mathrm{dom}(K)$ and for all $s \in \mathrm{dom}(K)$, $J_s(\mathcal{S}^{(K)}, G, s)$ (resp. $J_w(\mathcal{S}^{(K)}, G, s)$) is finite.*

**Proposition 1.** *Each strong solution of a control problem $(\mathcal{S}, I, G)$ is also a weak solution. If $\mathcal{S}$ is deterministic, any weak solution is also a strong solution.*

**Discrete Time Linear Hybrid Systems** A *Discrete Time Linear Hybrid System* (DTLHS) $\mathcal{H}$ is a tuple $(X,\ U,\ Y,\ N)$ where: $X = X^r \cup X^d$ is a finite sequence of real $(X^r)$ and discrete $(X^d)$ *present state* variables. We denote with $X'$ the sequence of *next state* variables obtained by decorating with $'$ all variables in $X$. $U = U^r \cup U^d$ is a finite sequence of *input* variables. $Y = Y^r \cup Y^d$ is a finite sequence of *auxiliary* variables. Auxiliary variables are typically used to model *modes* or *uncontrollable inputs*. $N(X, U, Y, X')$ is a linear predicate over $X \cup U \cup Y \cup X'$ defining the *transition relation* (*next state*) of the system. The dynamics of a DTLHS $\mathcal{H}$ is defined by the labeled transition system $\mathrm{LTS}(\mathcal{H}) =$

$(\mathcal{D}_X,\, \mathcal{D}_U,\, \bar{N})$ where: $\bar{N} : \mathcal{D}_X \times \mathcal{D}_U \times \mathcal{D}_X \to \mathbb{B}$ is a function s.t. $\bar{N}(x, u, x') = \exists\, y \in \mathcal{D}_Y\ N(x, u, y, x')$. A *state* $x$ for $\mathcal{H}$ is a state $x$ for LTS($\mathcal{H}$) and a *path* for $\mathcal{H}$ is a path for LTS($\mathcal{H}$).

**Definition 4.** *Let $\mathcal{H} = (X, U, Y, N)$ be a DTLHS and let $I$ and $G$ be linear predicates over $X$. The* DTLHS reachability problem $\mathcal{R} = (\mathcal{H}, I, G)$ *is defined as the LTS reachability problem* (LTS($\mathcal{H}$)*, $I$, $G$*). *Similarly, the* DTLHS control problem $(\mathcal{H}, I, G)$ *is defined as the LTS control problem* (LTS($\mathcal{H}$)*, $I$, $G$*).

**Quantized Control Problem** Quantization is the process of approximating a continuous interval by a set of integer values. A *quantization function* $\gamma : \mathbb{R} \mapsto \mathbb{Z}$ is a non-decreasing function, such that for any bounded interval $I = [a, b] \subset \mathbb{R}$, $\gamma(I)$ is a bounded integer interval. Given a DTLHS $\mathcal{H} = (X, U, Y, N)$ a *quantization* $\mathcal{Q}$ for $\mathcal{H}$ is a pair $(\mathcal{A}, \Gamma)$, where (let $W = X \cup U \cup Y$): $\mathcal{A}$ is a predicate over $W$ that explicitely bounds each variable in $W$. For each $w \in W$, we denote with $\mathcal{A}_w$ its *admissible region* and with $\mathcal{A}_W = \prod_{w \in W} \mathcal{A}_w$ and $\Gamma$ is a set of maps $\{\gamma_w \mid w \in W$ and $\gamma_w$ is a quantization function$\}$. Let $W = [w_1, \ldots, w_k]$ and $v = [v_1, \ldots, v_k] \in \mathcal{A}_W$. We write $\Gamma(v)$ for the tuple $[\gamma_{w_1}(v_1), \ldots, \gamma_{w_k}(v_k)]$.

A control problem admits a *quantized* solution if control decisions can be made by just looking at quantized values. This enables a software implementation for a controller.

**Definition 5.** *Let $\mathcal{H} = (X, U, Y, N)$ be a DTLHS, $\mathcal{Q} = (\mathcal{A}, \Gamma)$ be a quantization for $\mathcal{H}$ and $\mathcal{P} = (\mathcal{H}, I, G)$ be a DTLHS control problem. A $\mathcal{Q}$* Quantized Feedback Control *(QFC) strong (resp. weak) solution to $\mathcal{P}$ is a strong (resp. weak) solution $K(x, u)$ to $\mathcal{P}$ such that $K(x, u) = \hat{K}(\Gamma(x), \Gamma(u))$ where $\hat{K} : \Gamma(\mathcal{A}_X) \times \Gamma(\mathcal{A}_U) \to \mathbb{B}$.*

**Undecidability of Quantized Feedback Control Problem** We prove the undecidability of the DTLHS quantized feedback control problem along the same lines of similar undecidability proofs [6, 5]. We first show that a two-counter machine $M$ can be encoded as a deterministic DTLHS $\mathcal{H}_M$ without controllable actions in such a way that $M$ halts if and only if $\mathcal{H}_M$ reaches a goal region. This immediately implies that DTLHS reachability is undecidable. Since $\mathcal{H}_M$ has no controllable inputs, existence of a weak controller is equivalent to a reachability problem. For the same reason, actions enabled by any controller for $\mathcal{H}_M$ do not depend on state variables. As a consequence, a quantized weak control problem is equivalent to a DTLHS control problem. Finally, by Proposition 1, weak solutions to deterministic LTS control problems are also strong solutions. Therefore, since $\mathcal{H}_M$ is deterministic, the quantized strong control problem for DTLHS is undecidable, too.

**Two-Counter Machines.** A *two-counter machine* [8] $M$ consists of two counters that store unbounded natural numbers and a finite control that is a finite sequence of statements $\langle 1 : \mathit{stmt}_1, \ldots, n : \mathit{stmt}_n \rangle$, where $\mathit{stmt} ::= \mathsf{inc}\ i\ k \mid \mathsf{dec}\ i\ k \mid \mathsf{beq}\ i\ k \mid \mathsf{halt}$, with $i \in \{0, 1\}$. As an example, if the counter $i$ is 0, the execution of $j : \mathsf{beq}\ i\ k$ causes a jump to the statement labeled $k$, otherwise the execution proceed with statement $j + 1$. The halting problem for two-counter machine is undecidable [8].

**Lemma 1.** *For any two-counter machine $M$, there exists a* bounded *and* deterministic *DTLHS $\mathcal{H}_M$, and two predicates $I$ and $G$ such that $M$ halts if and only if $G$ is reachable from $I$ in $\mathcal{H}_M$.*

*Proof.* (Sketch) Let $M$ be a two-counter machine and let $\mathcal{H}_M$ be the DTLHS $(X, U, Y, N)$, where $X^r = \{x_0, x_1\}$, $X^d = \{l, g\}$, and $U = Y = \varnothing$. We use two real variables $x_0$ and $x_1$ to encode values stored in counters. Each natural number $m$ is encoded by the rational number $1/2^m$. A discrete variable $l$ stores the label of the statement currently under execution. Finally, the boolean variable $g$ encodes termination of the computation of $M$. The transition relation $N$ encodes the execution of the control program. A program $\langle 1 : stmt_1, \ldots, n : stmt_n \rangle$ is encoded by the predicate $N = \bigwedge_{j=1}^{n} [\![ j : stmt_j ]\!]$. As an example, here we present the encoding of the `beq` statement: $[\![ j : \texttt{beq}\ i\ k ]\!] \equiv (l \neq j) \vee (((x_i \neq 1) \vee (l' = k)) \wedge ((x_i = 1) \vee (l' = l + 1)) \wedge (x_{1-i} = x'_{1-i}) \wedge (g = g'))$.

An immediate consequence of Lemma 1 is the undecidability of the DTLHS reachability problem.

**Theorem 1.** *The reachability problem for bounded DTLHSs is undecidable. Existence of strong and weak solutions to a control problem for a bounded DTLHS is undecidable. Existence of QFC strong and weak solutions to a DTLHS control problem is undecidable.*

**Conclusions** We have shown that, for DTLHSs, existence of a quantized sampling controller meeting given (safety and liveness) system level specifications is undecidable. The relevance of such a problem stems from the fact that the *control software* implementing the controller in a software based control system always yields a quantized sampling controller. Investigating interesting classes of (discrete time) hybrid systems for which quantized sampling control is decidable appears to be an interesting future work.

## References


1. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P.H., Nicollin, X., Olivero, A., Sifakis, J., Yovine, S.: The algorithmic analysis of hybrid systems. Theoretical Computer Science 138(1), 3 – 34 (1995)
2. Alur, R.: Timed automata. In: CAV. pp. 8–22. LNCS 1633 (1999)
3. Asarin, E., Bouajjani, A.: Perturbed turing machines and hybrid systems. In: LICS. pp. 269–278 (2001)
4. Cassez, F., Henzinger, T.A., Raskin, J.F.: A comparison of control problems for timed and hybrid systems. In: HSCC. pp. 134–148 (2002)
5. Henzinger, T.A., Kopke, P.W.: Discrete-time control for rectangular hybrid automata. In: ICALP. pp. 582–593 (1997)
6. Henzinger, T.A., Kopke, P.W., Puri, A., Varaiya, P.: What's decidable about hybrid automata? J. of Computer and System Sciences 57(1), 94–124 (1998)
7. Mari, F., Melatti, I., Salvo, I., Tronci, E.: Synthesis of quantized feedback control software for discrete time linear hybrid systems. In: CAV. pp. 180–195. LNCS 6174 (2010)
8. Minsky, M.L.: Recursive unsolvability of post's problem of "tag" and other topics in theory of turing machines. The Annals of Mathematics 74(3), pp. 437–455 (1961)
9. Vidal, R., Schaffert, S., Shakernia, O., Lygeros, J., Sastry, S.: Decidable and semi-decidable controller synthesis for classes of discrete time hybrid systems. In: CDC. pp. 1243–1248. IEEE Computer Society (2001)