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

Federico Mari Igor Melatti **Ivano Salvo** Enrico Tronci

SAPIENZA
UNIVERSITÀ DI ROMA

Model Checking Group `http://mclab.di.uniroma1.it/`
Computer Science Department – Sapienza University of Rome

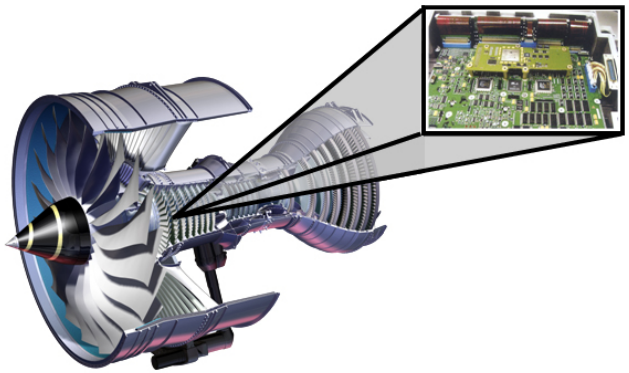
September 21, 2012
ICTCS 2012 – Varese, Italy

Outline

Embedded Systems

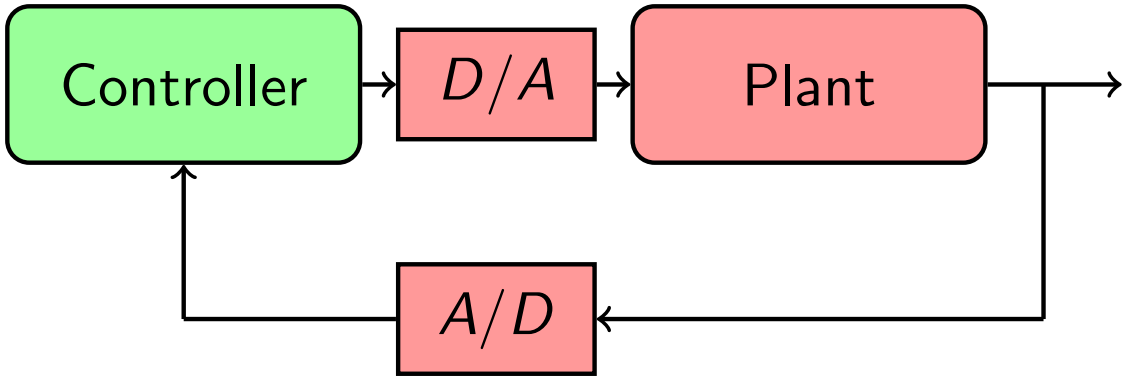
Examples

Wikipedia: An *embedded system* is a computer system designed to do a few dedicated functions with real-time computing constraints



Embedded Systems

modelled as Hybrid Systems



every T seconds **do** // *sampling time*
$\hat{x} = AnalogToDigital(read(\text{plantState}))$
try {
$\hat{u}$=$ctrLaw(\hat{x})$
$send(DigitalToAnalog(\hat{u}))$
} **catch** $(notInCtrReg(\hat{x}))$
{ $FDIR(\hat{x})$ } // *fault isolation and recovery*

Control Software Synthesis (Ideally)

Model Based Design [Henzinger, Sifakis, 2006]

Specifications are easier to define than control software

Input:

- Plant modelled as a (discrete time) **Hybrid System**
- Closed Loop System Level Specifications (**Safety** + **Liveness**)
- Implementation Specifications (**WCET**, **quantization**, etc.)

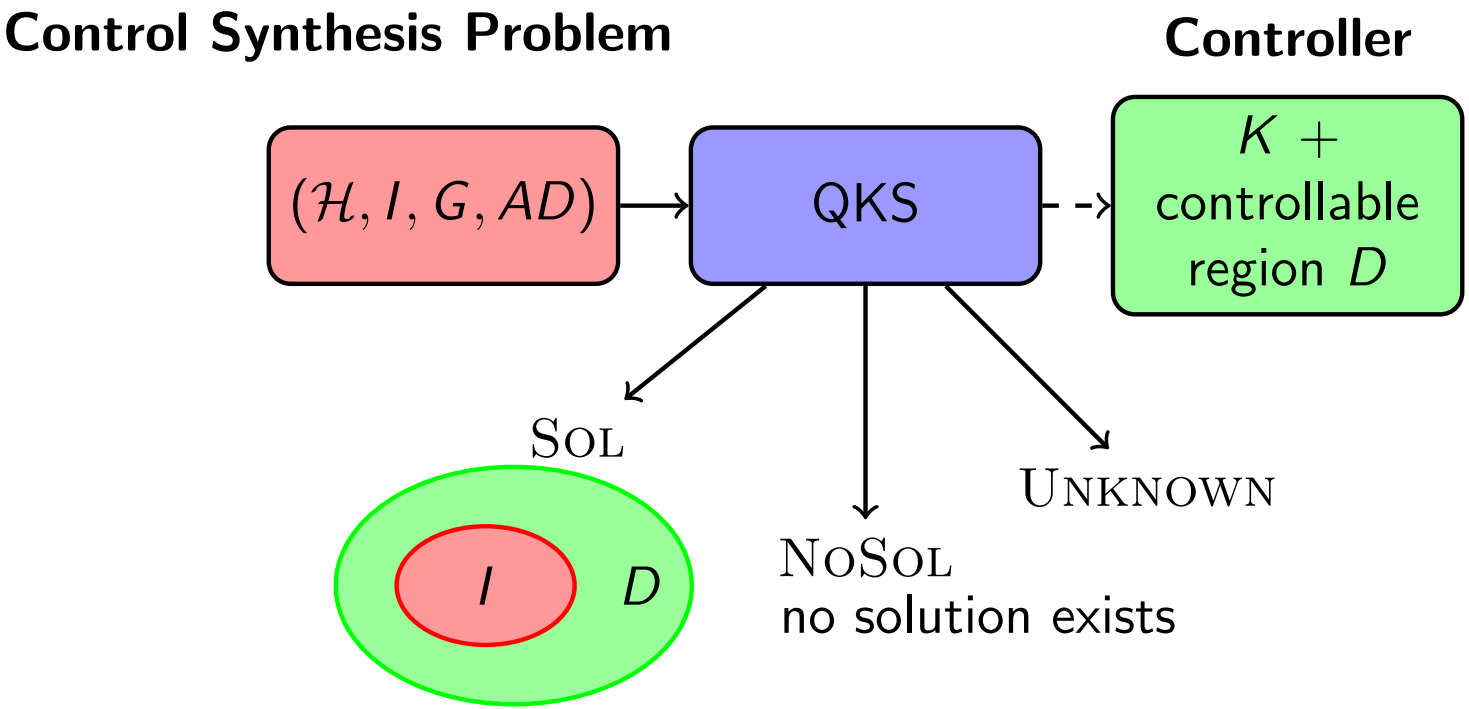
Output:

- **Correct-by-construction** automatically generated control software
- Guaranteed non functional requirements (WCET)
- **Robustness** (wrt plant parameter variations or disturbances)

SAPIENZA
UNIVERSITÀ DI ROMA

Controller Synthesis for DTLHSs

The tool QKS [Mari, Melatti, Salvo, Tronci, CAV 2010, EMSOFT 2012, CDC 2012]



UNKNOWN stems from undecidability of the problem

Outline

Discrete Time Linear Hybrid Systems (DTLHSs)

A **Discrete Time Linear Hybrid System (DTLHS)** $\mathcal{H}$ is a tuple (X, U, Y, N) where:

- X is a finite sequence of **present state variables**. *Next state* variables X' are obtained by decorating with $'$ all variables in X.
- U is a finite sequence of **input variables**, that models *controllable inputs*.
- Y is a finite sequence of **auxiliary variables** that models *modes* or *uncontrollable inputs* (e.g., *disturbances*).
- $N(X, U, Y, X')$ is a linear predicate over $X \cup U \cup Y \cup X'$ defining the **transition relation** (*next state*) of the system.

Each variable $w \in W$ range over a bounded or unbounded integer or real interval $\mathcal{D}_w$. $D_W = \prod_{w \in W} D_w$.

SAPIENZA
UNIVERSITÀ DI ROMA

DTLHS Semantics

as Labeled Transition Systems (LTS)

A **Labeled Tranisition System (LTS)** $\mathcal{S}$ is a tuple (S, A, T)

- S is a possibly infinite set of **states**,
- A is a possibly infinite set of **actions**
- $T : S \times A \times S \to \mathbb{B}$ is the **transition relation** of $\mathcal{S}$.

A **run** for $\mathcal{S}$ is a sequence $\pi = s_0, a_0, s_1, a_1, s_2, a_2, \ldots$ of states s_t and actions a_t s. t. $\forall t \geq 0 \; T(s_t, a_t, s_{t+1})$.

The dynamics of $\mathcal{H}$ is defined by $\mathrm{LTS}(\mathcal{H}) = (\mathcal{D}_X, \mathcal{D}_U, \bar{N})$ where: $\bar{N} : \mathcal{D}_X \times \mathcal{D}_U \times \mathcal{D}_X \to \mathbb{B}$ is a function s.t. $\bar{N}(x, u, x') = \exists\, y \in \mathcal{D}_Y \; N(x, u, y, x')$.

Controller and Closed Loop System

A formal definition

A **controller** restricts the dynamics of an LTS $\mathcal{S}$ so that all states an *initial region I* will reach in one or more steps a givern *goal region G* (**Liveness Specifications**).

A *controller* for $\mathcal{S}$ is a function $K : S \times A \to \mathbb{B}$ such that $\forall s \in S$, $\forall a \in A$, if $K(s, a)$ then $\exists s'\ T(s, a, s')$.

$\mathcal{S}^{(K)}$ denotes the **closed loop system**, that is the LTS $(S, A, T^{(K)})$, where $T^{(K)}(s, a, s') = T(s, a, s') \wedge K(s, a)$. (i.e. the plant in parallel with the controller)

LTS Reachability and Control Problem

A **reachability problem** and a **control problem** are 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 π of $\mathcal{S}$ such that $\pi^{(S)}(0) \in I$ and $\pi^{(S)}(t) \in G$ for some $t \in \mathbb{N}$.

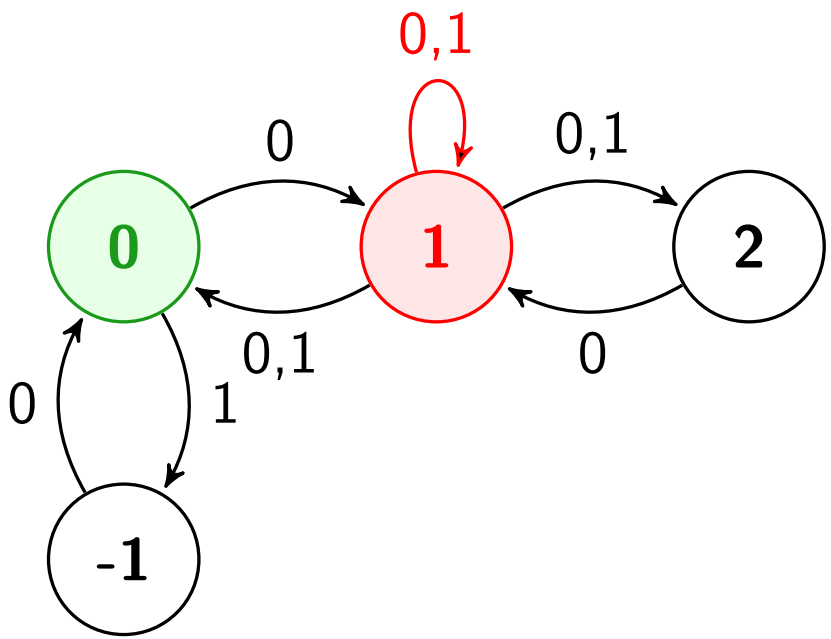
The control problem $(\mathcal{S}, I, G)$ has a **solution** if there exists a controller K such that **all runs** starting in I reach G in a **finite number of steps** in the closed loop system $\mathcal{S}^{(K)}$.

SAPIENZA
UNIVERSITÀ DI ROMA

LTS Reachability Control Problem

Example (I)

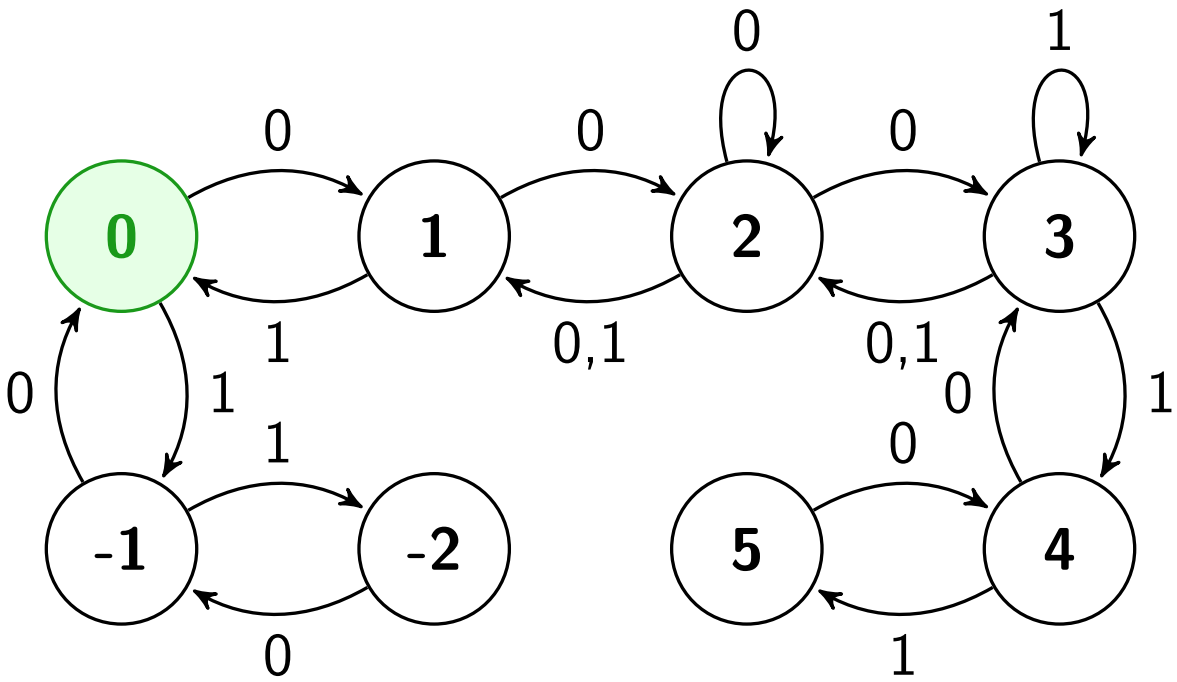
There is no quantized solution because of self–loops in state 1. The worst case distance of 0 from 1 is **infinite**.



LTS Reachability Control Problem

Example (II)

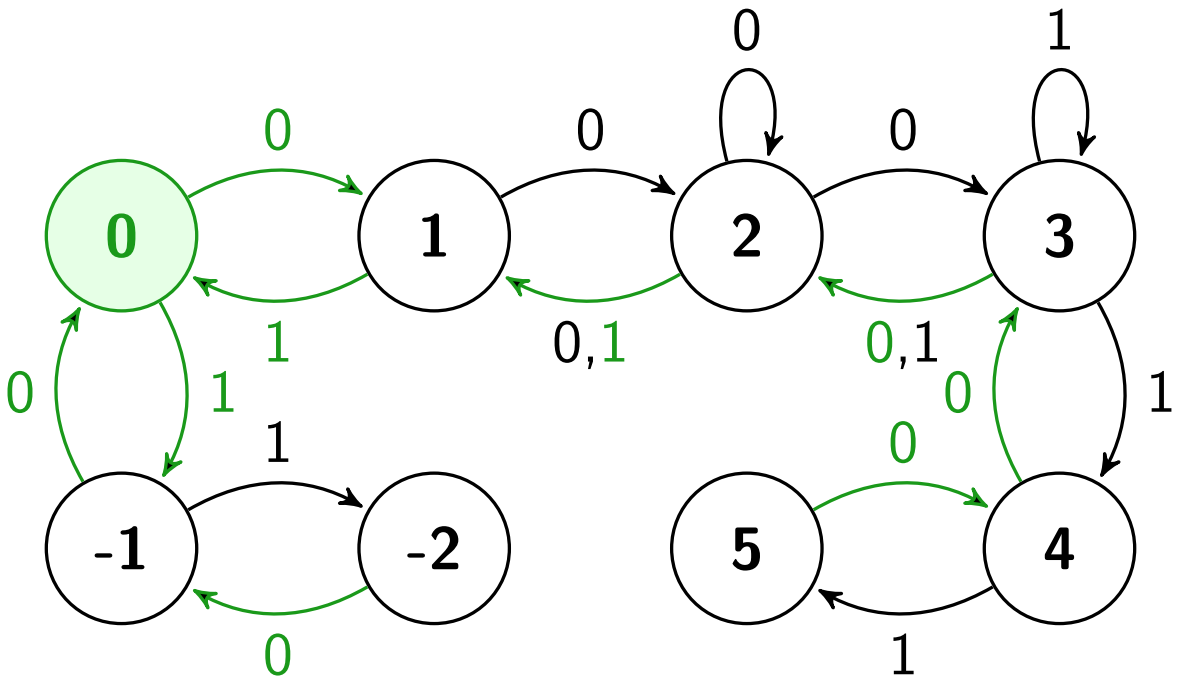
Quantized solutions exist.



Quantized Control Problem

Example (II)

The controller that enables green actions is a solution.



DTLHS Reachability and Control Problem

A **DTLHS control problem** (**reachability problem**) $(\mathcal{H}, I, G)$ is defined as the LTS control problem (reachability) $(LTS(\mathcal{H}), I, G)$.

Example. Let T be $1/10$ (sampling time). Let $\mathcal{H}$ be $(\{x\}, \{u\}, \varnothing, N)$ where: x is a continuous variable, u is a boolean variable, and

$$N(x, u, x') \equiv [\overline{u} \rightarrow x' = x + (5/4 - x)\,T] \wedge [u \rightarrow x' = x + (x - 7/4)\,T].$$

Let us consider the control problem $\mathcal{P} = (\mathcal{H}, I, G)$, where:

$$I(x) \equiv -1 \leq x \leq 5/2 \qquad \text{and} \qquad G(x) \equiv 0 \leq x \leq 1/2.$$

A **solution** K to $\mathcal{P}$ is:

$$K(x, u) = (-1 \leq x < 0 \wedge \overline{u}) \vee (0 \leq x < 3/2 \wedge u) \vee (3/2 \leq x \leq 5/2 \wedge \overline{u}).$$

Observe that $N(5/4, 0, 5/4)$ and $N(7/4, 1, 7/4)$ hold, hence no solution can enable action 0 in $5/4$ and action 1 in $7/4$.

Quantized Control Problem for DTLHSs

A **quantization function** is a non-decreasing function $\gamma_x : \mathcal{D}_x \subseteq \mathbb{R} \mapsto [a, b] \subseteq \mathbb{Z}$

The quantization of a sequence of variables is a sequence of quantization functions: $\Gamma = \{\gamma_{x_1}, \ldots, \gamma_{x_n}\} \quad \Gamma(s) = \langle \gamma_{x_1}(s_1), \ldots, \gamma_{x_n}(s_n) \rangle$

K is a **quantized controller** if there exists $\hat{K} : \Gamma(\mathcal{D}_X) \times \Gamma(\mathcal{D}_U) \to \mathbb{B}$, such that $K(s, a) = \hat{K}(\Gamma(s), \Gamma(a))$.

This enables a **software implementation** of the controller.

Quantized Control Problem

Control Abstraction

We build a finite LTS, the **control abstraction** of a DTLHS $\mathcal{H}$

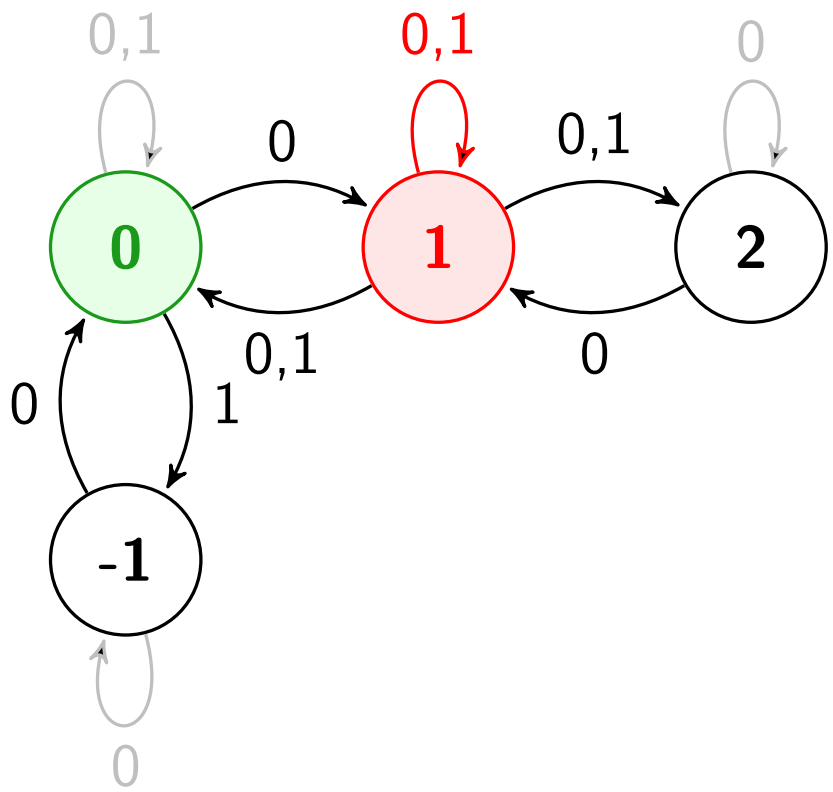
- The set of states is $\Gamma(\mathcal{D}_X)$
- The set of action is $\Gamma(\mathcal{D}_U)$
- $T(s, a, s')$ iff there exists $x \in \Gamma^{-1}(s)$, $x' \in \Gamma^{-1}(s')$, $u \in \Gamma^{-1}(a)$, $y \in \mathcal{D}_y$ such that $N(x, u, y, x')$

A self–loop $T(s, a, s)$ is **non-eliminable** if there exists an infinite run $\pi = x_0 u_0 x_1 u_1 x_2 \ldots$ in $\mathcal{H}$ such that $\forall t \in \mathbb{N}\ x_t \in \Gamma^{-1}(\hat{s})$ and $a_t \in \Gamma^{-1}(\hat{a})$.

Quantized Control Problem

Example (I)

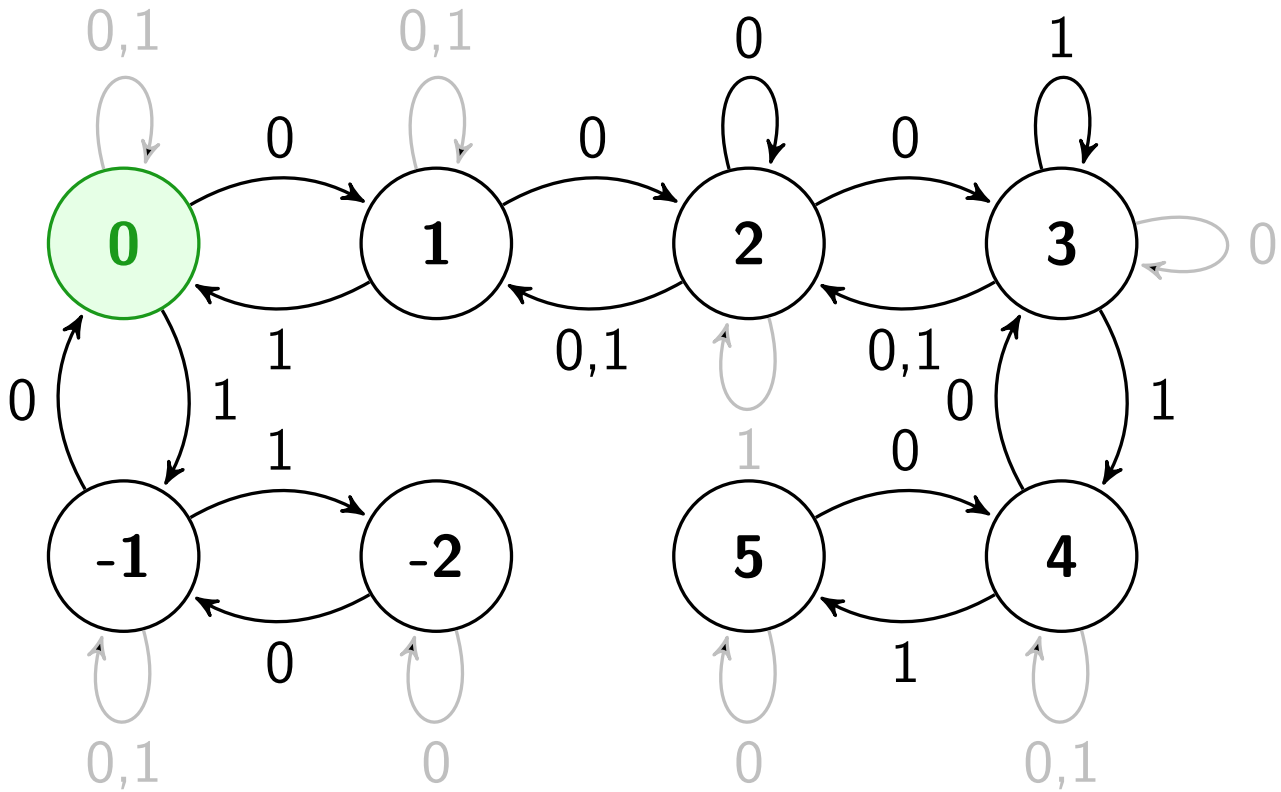
Let $\gamma_x(x) = \lfloor x \rfloor$. There is no quantized solution because of self–loops in state 1. The worst case distance of 0 from 1 is **infinite**.



Quantized Control Problem

Example (II)

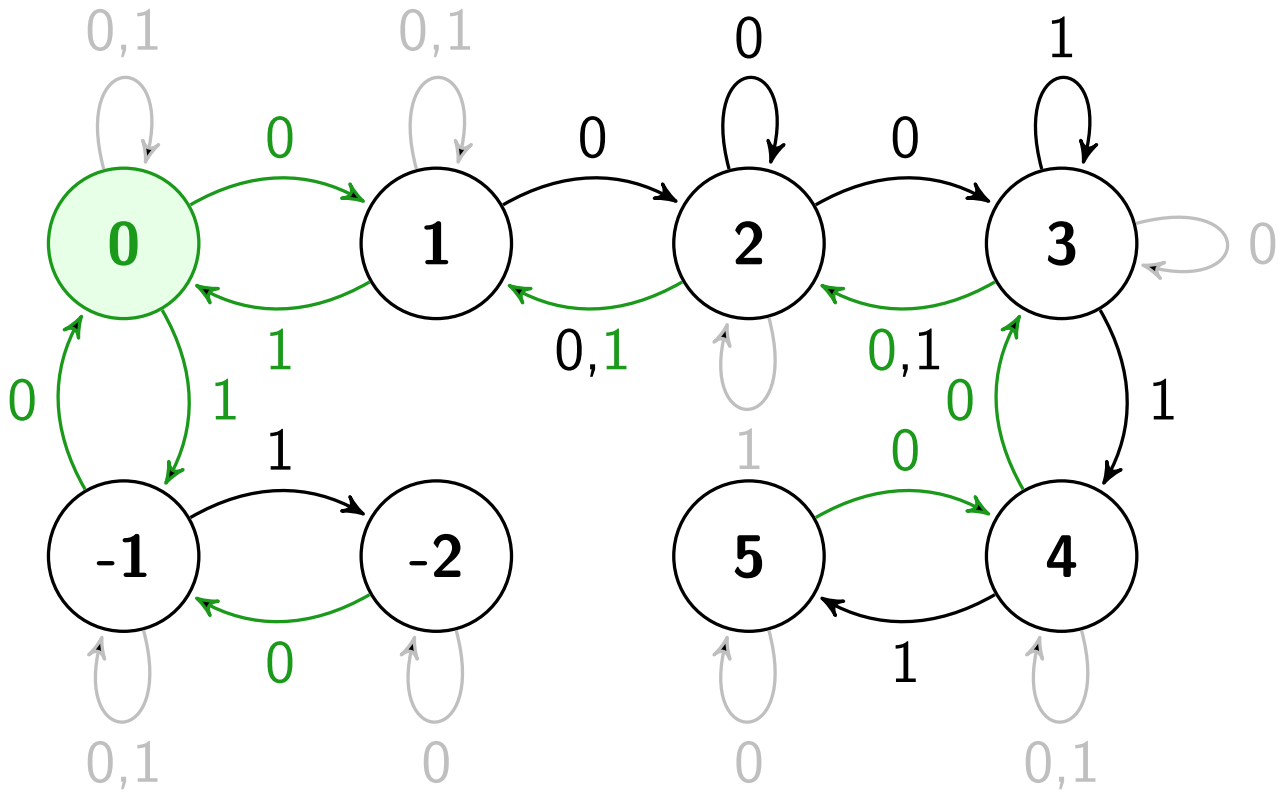
Let $\gamma_x(x) = \lfloor 2x \rfloor$. Quantized solutions exist. (Light gray edges are eliminable self–loops)



Quantized Control Problem

Example (II)

Let $\gamma_x(x) = \lfloor 2x \rfloor$. Quantized solutions exist. (Light gray edges are eliminable self–loops)



Outline

A First Attempt

inspired by [Henzinger, Kopke, Puri, Varaiya - 1996]

Reduce a known undecidable problem about Hybrid Systems to (quantized) DTLHS control problem.

Rectangular Hybrid Automata reachability can be encoded into DTLHS reachability and control problems (see **[Mari, Melatti, Salvo, Tronci, ICTAC 2012]**). This **proves undecidability of DTLHS reachability and control problem**.

This is an interesting result (enconding of a dense time model into a discrete time model). However, the **undecidability** of the DTLHS **quantized control problem** does not follow immediately.

Given a quantization schema, **the number of quantized controllers is finite**. This may lead to think that the problem might be decidable.

SAPIENZA UNIVERSITÀ DI ROMA

Basic Idea

Reduce halting problem for Two Counter Turing Machines to DTLHS (quantized) control problem.

For any *two counter Turing Machine M* we find a DTLHS $\mathcal{H}_M$ and control problem $\mathcal{P} = (\mathcal{H}_M, I, G)$ such that M **halts if and only if** $\mathcal{P}$ **has a solution**.

In our encoding, $\mathcal{H}_M$ **has no controllable actions**, and the undecidability of the quantized control problem for DTLHSs trivially follows.

Two Counter Turing Machines

[Minsky 1961]

Two-counter Turing Machines are a *minimal* **Turing-complete** model of computation. They consist of:

- two counters that store unbounded natural numbers
- finite control program $\langle 1 : stmt_1, \ldots, n : stmt_n \rangle$ where:

$$stmt ::= \text{inc } i \; k \mid \text{dec } i \; k \mid \text{beq } i \; k \mid \text{halt} \qquad (i \in \{0, 1\})$$

- Example of computation
  - $j : \text{beq } i \; k \rightarrow \begin{cases} \text{goto statemet labeled } k \text{ if counter } i \text{ stores } 0 \\ \text{goto statemet labeled } k+1 \text{ otherwise} \end{cases}$
  - $j : \text{inc } i \; k \rightarrow$ add 1 to counter i and goto statemet labeled k

Encoding a Two Counter TM into a DTLHS (I)

Given a two-counter machine M let $\mathcal{H}_M$ be the DTLHS (X, U, Y, N), where:

- $X^r = \{x_0, x_1\}$, $\mathcal{D}_{x_i} = [0, 1]$
  (**idea:** if counter i stores the natural n, x_i assumes value $1/2^n$)
- $X^d = \{l, g\}$
  (**idea**: l stores labels of statements, g is 1 iff the computation halts)
- $U = Y = \varnothing$.

Encoding a Two Counter TM into a DTLHS (II)

Transition Relation

A program $\langle 1 : stmt_1, \ldots, n : stmt_n \rangle$ is encoded by the predicate $N = \bigwedge_{j=1}^{n} [\![j : stmt_j]\!]$, where:

$$
\begin{aligned}
[\![j : \text{dec } i \; k]\!] &\equiv (l \neq j) \vee (((x_i = 1) \vee (x_i' = 2x_i)) \wedge \\
&\quad \wedge ((x_i \neq 1) \vee (x_i' = 1)) \wedge (l' = k) \wedge U(x_{1-i}, g)) \\
[\![j : \text{inc } i \; k]\!] &\equiv (l \neq j) \vee ((x_i' = {}^{x_i}/_2) \wedge (l' = k) \wedge U(x_{1-i}, g)) \\
[\![j : \text{beq } i \; k]\!] &\equiv (l \neq j) \vee (((x_i \neq 1) \vee (l' = k)) \wedge \\
&\quad \wedge ((x_i = 1) \vee (l' = l + 1)) \wedge U(x_{1-i}, g)) \\
[\![j : \text{halt}]\!] &\equiv (l \neq j) \vee ((l' = j) \wedge (g' = 1) \wedge U(x_0, x_1))
\end{aligned}
$$

Main Results

Lemma. For any two-counter machine M, there exists a *bounded*, *conjunctive*, 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 $\mathcal{H}_M$.

Theorem. The reachability problem for bounded and conjunctive DTLHSs is undecidable.

Theorem. Existence of solutions to a control problem for a bounded and conjunctive DTLHS is undecidable.

Theorem. Existence of quantized solutions to a DTLHS control problem is undecidable.

SAPIENZA
UNIVERSITÀ DI ROMA

Outline

Conclusion and Future Work

Conclusions

- We have shown that, for DTLHSs, existence of a quantized sampling controller meeting given (safety and liveness) system level specifications is undecidable.
- we have shown that *Rectangular Automata* (RA), and thus *Timed Automata* (TA), can be modelled as DTLHSs.

Future Work

- Investigating interesting classes of **discrete time** hybrid systems for which quantizing sampling control is decidable: however this can lead to consider **not enough expressive models**.
- Find **easy to compute sufficient conditions** (resp. **necessary conditions**) for the existence (resp. non existence) of a quantized controller.

SAPIENZA UNIVERSITÀ DI ROMA

Good News!

for smart "young" people

- **Post-Doc positions available in the Model Checking Group in Rome** within 2 FP7 european projects.
- **Topics**: application of control synthesis to Energy distribution (SmartHG) and Human Fertility (PAEON).
- **Not exactly TCS**, but people involved in Formal Methods are welcome to apply (Remember: practice can inspire excellent theorectical work!).

SAPIENZA
UNIVERSITÀ DI ROMA

Any questions?

SAPIENZA
UNIVERSITÀ DI ROMA