Discrete Abstraction and Supervisory Control of Switching Systems*

Rong Su † Sherif Abdelwahed ‡ Gabor Karsai ‡ Gautam Biswas ‡

Abstract – *In this paper we propose a method to create discrete abstraction of state space behavior for continuous-time systems based on gradient analysis of the system dynamics. Then we describe how to use such a discrete model to design a supervisory controller for a given safety spiflication for the system. Finally we provide an entropy measure of nondeterminism, which can be used to evaluate the quality of the result discrete model as the degree of nondeterminism in that model.*

Keywords: Discrete abstraction, supervisory control, hybrid systems.

1 Introduction

In this paper we consider the problem of discrete abstraction of system bahavior and supervisory control for a special class of dynamic systems, referred to as *switching systems*. Switching systems are characterized by a finite set of inputs, and a finite set of operating modes. They represent a special class of hybrid systems.

Considerable research work has been dedicated recently to the study of hybrid systems. See for example [1, 2] and the references therein. In developing supervisory control of hybrid systems, discrete abstractions are usually employed to approximate the continuous dynamics of the system. Building approximate discrete finite state models of hybrid systems has been an active area of research for more than a decade. Several conservative abstraction techniques have been proposed in literature, e.g. in [3] methods for hybrid systems with discrete valued input and outputs; in [6, 7] for linear systems are presented; in [8, 11, 4] for nonlinear continuous-time systems are given; and in [9, 10] methods for nonlinear discretizable systems are proposed.

The proposed technique is aimed at reducing a switching system into a finite state structure that preserves the important dynamics of the original system. It addresses the task of discrete abstraction by performing gradient analysis at the boundaries of different cells in a chosen partition on the state space of the continuous-time system. Since computation is restricted to boundary regime, this method is more computationally efficient than trajectory-based abstraction techniques in the literature, e.g., [6, 9, 10]. Abstraction based on gradient analysis has been proposed in [8, 4]. However, these approaches target a general setting with no specific assumptions about the target system, and, therefore, did not yield a concrete computation procedure for gradient analysis. The abstraction problem is addressed and solved in detail in this paper. A supervisory control scheme for the abstracted system is then proposed. Finally, we also propose a method to measure nondeterminism of a discrete model, which offers us a way to evaluate the quality of the given partition.

This paper is organized as follows. In section 2, we describe how to apply gradient analysis in discrete abstraction for a general class of switching systems. In section 3 we describe the application of supervisory control technique on the discrete model. In section 4 we discuss how to measure nondeterminism of a discrete-event mode. Conclusions and further research directions are presented in section 5.

2 Discrete abstraction

Given an n-dimensional continuous-time dynamic system, which consists of p discrete modes, the dynamic model in a mode M_i takes on the following form,

$$\dot{x} \quad = \quad f_{M_i}(x,u) \qquad \text{with } x \in X_i, \tag{1}$$

where $x = [x_1, \cdots, x_n]$, $f_{M_i} = [f_1, \cdots, f_n]$ and $u = [u_1, \cdots, u_m]$ are vectors. As a simple example, we use the two-tank system shown in Figure 1. It consists of two tanks (T_1 and T_2) and two valves (V_1, V_2). Valve operations are binary, i.e., they can be fully open (=1) or fully close (=0). When V_1 is open, there is a steady water flow into the system. Water level in each tank is

Funded, in part, by the DARPA IXO Software-Enabled Control Program under contract F33615-99-C-3611.

†Corresponding author, email: surong@control.utoronto.ca, Department of Electrical and Computer Engineering, University of Toronto, Toronto, Ontario, Canada

‡emails: {sherif,gabor,gautam}@isis-server.isis.vanderbilt.edu Institute for Software Integrated Systems, Vanderbilt University, Nashville, TN, USA

monitored by a level-crossing sensor, which only reports whether water level is above or below some predefined values, e.g. above 0.3m or below 0.3m. The state-space

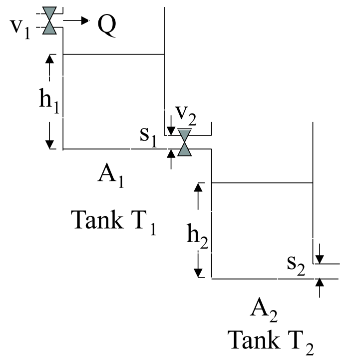

Figure 1: Two-Tank System

equations for the system, shown below cover all operating modes:

$$\dot{x} = \begin{bmatrix} \dot{h}_1 \\ \dot{h}_2 \end{bmatrix} = \begin{bmatrix} \frac{u_1 Q - u_2 s_1 \sqrt{2gh_1}}{A_1} \\ \frac{u_2 s_1 \sqrt{2gh_1} - s_2 \sqrt{2gh_2}}{A_2} \end{bmatrix} = f(x, u) \quad (2)$$

where $Q = 5m^3/s$ is the steady water flow rate, $h_i \in [0, 9)(m)$, $A_i = 2m^2$, $s_i = 0.3m^2$ is the intersection area of Tank i, $u_i \in \{0 : \text{off}, 1 : \text{on}\}$ is the control action of V_i with $i \in [1, 2]$, and $g = 9.8Nm/s^2$ is the gravity constant

Given a set X, we use the notation ∂X to mean the boundary of X (i.e., the set of limit points of X). An n-dimensional polyhedron $X \subseteq \mathcal{R}^n$ is *regular* if there is a set of lower and upper bounds $\{a_{i,l}, a_{i,h} \in \mathcal{R} | a_{i,l} < a_{i,h} \wedge 1 \leq i \leq n\}$ such that

$$X \cup \partial X = \{(x_1, \cdots, x_n) \in \mathcal{R}^n | a_{i,l} \leq x_i \leq a_{i,h}, i \in [1, n]\}.$$

We make the following assumptions about the continuous-time dynamic system:

A$_1$: u is from a finite set U.

A$_2$: For each i, X_i is a regular polyhedron, and so is $X = \bigcup_{i=1}^m X_i$.

A$_3$: $(\forall i, j)\, X_i \cap X_j = \varnothing$.

A$_4$: For each control action $u \in U$, $\dot{x}$ is continuous in $X \cup \partial X$.

With those assumptions our objective is to build a discrete-event model

$$\mathbf{G} = (Z, U, \xi),$$

where Z is the *state set*, U is the *transition event set*, and $\xi : Z \times U \rightarrow Z$ is the *(partial) transition function.* **G** is supposed to be an *abstraction* (which will be defined latter) of the continuous-time system.

Assume that for each i, X_i is partitioned into a set Z_i of regular n-dimensional polyhedrons. Then $Z = \bigcup_{i=1}^m Z_i$. In the two-tank system, $X = \{(h_1, h_2) | h_1 \in [0, 9) \wedge h_2 \in [0, 9)\}$, hence it is a regular polyhedron. We divide the height of each tank into three equal intervals $[0, 3)$, $[3, 6)$, $[6, 9)$ and let W be the set of these intervals. Then $Z = W \times W$ and each element $z \in Z$ is a regular polyhedron. Figure 2 depicts the partition on X.

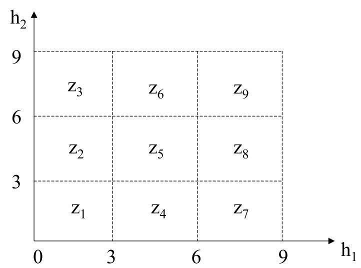

Figure 2: Partition on X in Two-Tank System

It is easy to check that the two-tank system model satisfies all above assumptions. Let $T_{u,x} : [0, \infty) \rightarrow X$ be a trajectory of the continuous-time system under control action u with $T_{u,x}(0) = x$. Let $T = \{T_{u,x} | u \in U \wedge x \in X\}$ be the set of all possible trajectories.

Definition 2.1 A discrete-event model **G** is an *abstraction* of the continuous-time model $\dot{x} = f(x, u)$, if for each pair of states $z_1, z_2 \in Z$ and each control action $u \in U$, $z_2 \in \xi(z_1, u)$ if and only if there exists a trajectory $T_{u,x} \in T$ such that

1. $T_{u,x}(0) \in z_1$ and $T_{u,x}(t) \in z_2$ for some $t < \infty$,

2. $\{t' | 0 \leq t' \leq t \wedge T_{u,x}(t') \in z_1 \cup z_2\}$ is dense in $[0, t]$.

□

The second condition means that "almost all" parts of $T_{u,x}$ is in $z_1 \cup z_2$; if it does include points which are not in Such "almost all" requirement instead of "all" has practical reasons which will be made clear later. Notice that Def. 2.1 implies that there exists a transition between z_i and z_j only if they are neighbored to each other. Otherwise by assumption of continuity A$_\mathbf{4}$, condition 2 in Def. 2.1 won't hold. Thus if such trajectory $T_{u,x}$ does exist then it must cross the *boundary* of z_1, z_2, namely $\overline{z_1} \cap \overline{z_2}$ where the symbol $\overline{z_1}$ represents the closure of z_1 (i.e. $z_1 \cup \partial z_1$). So in order to build the (partial) transition function ξ we only need to analyze those boundary crossings. Given two states:

$$\begin{aligned} z_i &= \{(x_1, \cdots, x_n) \in X | (\forall k)\, x_k \in [b^i_{k,l}, b^i_{k,h})\} \\ z_j &= \{(x_1, \cdots, x_n) \in X | (\forall k)\, x_k \in [c^j_{k,l}, c^j_{k,h})\} \end{aligned}$$

Let $d^{ij}_{k,l} = \max\{b^i_{k,l}, c^j_{k,l}\}$ and $d^{ij}_{k,h} = \min\{b^i_{k,h}, c^j_{k,h}\}$. Then their boundary is defined by

$$B(z_i, z_j) = \Big\{(x_1, \cdots, x_n) \in X | \\ (\forall k)\, x_k \in [d^{ij}_{k,l}, d^{ij}_{k,h}] \wedge d^{ij}_{k,l} \le d^{ij}_{k,h}\Big\}$$

If $B(z_i, z_j) = \varnothing$ then states z_i and z_j are not adjacent to each other, hence no transition exists between them. Now suppose $B(z_i, z_j) \neq \varnothing$ and z_i in mode M_i, z_j in mode M_j. We discuss how to determine boundary crossings on $B(z_i, z_j)$.

By assumption $\mathbf{A}_3$ we get that $dim(B(z_i, z_j)) < n$, namely $(\exists k)\, d^{ij}_{k,l} = d^{ij}_{k,h}$. We call $x_k = d^{ij}_{k,l} = d^{ij}_{k,h}$ a *border*. Let $\mathcal{I}(z_i, z_j) = \{k | d^{ij}_{k,l} = d^{ij}_{k,h}\}$ and

$$\phi(z_i, z_j) := \{(x_1, \cdots, x_n)_k \mid k \in \mathcal{I}(z_i, z_j) \wedge \\ x_k = \frac{c^j_{k,l} + c^j_{k,h} - b^i_{k,l} - b^i_{k,h}}{|c^j_{k,l} + c^j_{k,h} - b^i_{k,l} - b^i_{k,h}|} \wedge (\forall q) q \neq k \Rightarrow x_q = 0\}$$

By assumptions $\mathrm{A}_\mathbf{2}$ both z_i and z_j are regular polyhedrons, so $b^i_{k,l} < b^i_{k,h}$ and $c^j_{k,l} < c^i_{k,h}$. Considering that $\max\{b^i_{k,l}, c^j_{k,l}\} = \min\{b^i_{k,h}, c^j_{k,h}\}$, we get either $b^i_{k,h} > b^i_{k,l} = c^j_{k,h} > c^j_{k,l}$ or $c^j_{k,h} > c^j_{k,l} = b^i_{k,h} > b^i_{k,l}$. In either case, $\phi(z_i, z_j)$ is well defined as a set of unit vectors. We use the notation "$\centerdot$" to represent the dot product of two vectors.

Proposition 2.1 Given two neighboring states $z_i, z_j \in Z$ and control action $u \in U$, if

$$(\exists \hat{x} \in B(z_i, z_j))(\forall \vec{r} \in \phi(z_i, z_j))\, f_{M_i}(\hat{x}, u) \centerdot \vec{r} > 0 \quad (3)$$

then there exists a trajectory $T_{u,x} \in T$ with $T_{u,x}(0) \in z_i$ and $T_{u,x}(t) \in z_j$ for some $t < \infty$, and the set $\{t' | 0 \le t' \le t \wedge T_{u,x}(t') \in z_i \cup z_j\}$ is dense in $[0, t]$.

The existence of such $\hat{x}$ is a sufficient condition for the transition $z_j \in \xi(z_i, u)$, but not necessary in general. Its necessary condition depends on higher order (left-hand side and right-hand side) derivatives of $x(t)$ on t. If we interpret $x(t)$ as a displacement, then $f_{M_i}(x, u) \centerdot \vec{r}$ is the velocity on the direction $\vec{r}$. Even if $f_{M_i}(x, u) \centerdot \vec{r} = 0$, namely the velocity projected onto $\vec{r}$ (i.e., the normal direction of a border) is zero, the trajectory can still cross the border from the current point instantaneously as long as the acceleration ($\ddot{x}$) or an even higher but finite order of derivative projected onto $\vec{r}$ is nonzero at the current point. If the high derivatives don't exist, then we need to consider left-hand side and right-hand side derivatives at the point where the velocity is zero. Considering the complexity involved in computing high order derivatives, we will not discuss it in this paper. Now the problem is to determine whether or not such $\hat{x}$ in Condition (3) exists. This can be converted to the following optimization problem.

$$\begin{array}{ll} \text{minimize} & \mathbf{J}_{z_i, z_j, u} = \sum_{\vec{r} \in \phi(z_i, z_j)} (f_{M_i}(x, u) \centerdot \vec{r} - y_{\vec{r}})^2 \\ \text{subject to} & x \in B(z_i, z_j) \quad \text{and} \\ & y = \{y_{\vec{r}} \ge 0 | \vec{r} \in \phi(z_i, z_j)\} \end{array}$$

We say $y > 0$ if $(\forall \vec{r} \in \phi(z_i, z_j))\, y_{\vec{r}} > 0$.

Proposition 2.2 Condition (3) in Proposition 2.1 holds iff $y > 0$ and $\mathbf{J}(z_i, z_j, u) = 0$.

Solvers for the above nonlinear constraint minimization problem can be found in several softwares, e.g. Optimization Toolbox in MATLAB, MathOptimizer in Mathematica. A discrete abstraction procedure can be described as follows. Suppose the resultant discrete model is stored in a matrix $DiscreteModel(ControlAction, TargetState, ExitState)$, where $DiscreteModel(u, z_j, z_i) = 1$ means there is a transition u from z_i to z_j, namely $z_j \in \xi(z_i, u)$. Initially all entries in $DiscreteModel$ are set to be zero.

Discrete abstraction procedure: Suppose $Z = \{z_1, \cdots, z_k\}$ and $U = \{u_1, \cdots, u_r\}$.

1. **for** $i = 1$ to $k - 1$ {
2.   **for** $j = i + 1$ to k {
3.   **for** $l = 1$ to r {
4.   **if** $\mathbf{J}_{z_i, z_j, u_l} = 0$ and $y_{z_i, z_j, u} > 0$,
5. $DiscreteModel(u, z_j, z_i) := 1$;
6.   **if** $\mathbf{J}_{z_j, z_i, u_l} = 0$ and $y_{z_i, z_j, u} > 0$,
7. $DiscreteModel(u, z_i, z_j) := 1$;}}}

In the two-tank system, after we put in all values of parameters, the system dynamic model is as follows,

$$\dot{x} = \begin{bmatrix} \dot{h}_1 \\ \dot{h}_2 \end{bmatrix} = \begin{bmatrix} 2.5u_1 - 0.89u_2\sqrt{h_1} \\ 0.89u_2\sqrt{h_1} - 0.49\sqrt{h_2} \end{bmatrix} = f(x, u) \quad (4)$$

Suppose $z_j = [3, 6) \times [3, 6)$. Then $B(z_i, z_j) = [3, 3] \times [6, 6] = \{(3, 6)\}$. This gives that $\phi(z_i, z_j) = \{r_1 = (1, 0), r_2 = (0, -1)\}$. With the control action $u = (1, 0)$ we have

$$\begin{aligned} \mathbf{J}(z_i, z_j, u) &= \min \sum_{\vec{r} \in \phi(z_i, z_j)} (f_{M_j}(x, u) \centerdot \vec{r} - y_{\vec{r}})^2 \\ &= \min[(2.5 - y_{(1,0)})^2 + (1.2 - y_{(0,-1)})^2] \end{aligned}$$

Clearly $\mathbf{J}(z_i, z_j, u) = 0$ which, by Propositions 2.1 and 2.2, implies that $z_j \in \xi(z_i, u)$. Notice that, the boundary point $x = (3, 6)$ is not in $z_i \cup z_j$. But there is a trajectory $T_{u,x}$ with $T_{u,x}(0) = (2.8, (\sqrt{6} + 0.0784)^2) \in z_i$ and $T_{u,x}(0.1) = (3.05, 5.904) \in z_j$ such that $\{t' | 0 \le t' \le 0.1 \wedge T_{u,x}(t') \in z_i \cup z_j\} = [0, t] \backslash \{0.08\}$, which is obviously dense in $[0, t]$. This trajectory passes the point $x = (3, 6)$ at $t = 0.08$ instantaneously. It is interesting to notice that, at each boundary point in the two tank system, higher order derivatives projected to a boarder will be zero if the first order derivative projected to the same boarder is zero. This fact makes the condition (3) in Prop. 2.1 a sufficient and necessary condition for the existence of a transition between two states, namely a discrete abstraction of the two-tank system constructed by the proposed procedure won't miss any transition. Figure 3 displays the result discrete-event model. In

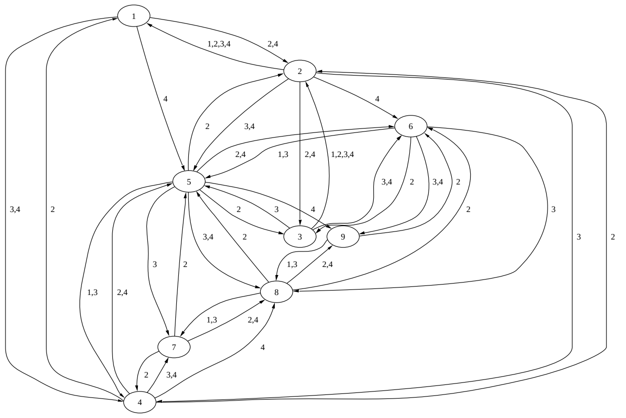

Figure 3: Discrete-Event Model for Two-Tank System

this picture the value in each state is the state number and the value on each edge is the control action. Control actions are defined in Table 1 and state numbers are defined in Table 2.

Table 1: Control action for the Two tanks System

Control Action	u_1	u_2
1	0 (V_1 close)	0 (V_2 close)
2	0 (V_1 close)	1 (V_2 open)
3	1 (V_1 open)	0 (V_2 close)
4	1 (V_1 open)	1 (V_2 open)

Table 2: State numbers for the Two tanks System

State Number	h_1	h_2
1	[0,3)	[0,3)
2	[0,3)	[3,6)
3	[0,3)	[6,9)
4	[3,6)	[0,3)
5	[3,6)	[3,6)
6	[3,6)	[6,9)
7	[6,9)	[0,3)
8	[6,9)	[3,6)
9	[6,9)	[6,9)

The above discrete abstraction procedure has also been applied to the well-known three-tank system. Once we have a discrete model $\mathbf{G} = (Z, U, \xi)$, we can impose supervisory control which is described in the next section.

3 Application of supervisory control on discrete model

Given the continuous model

$$\dot{x} = f_{M_i}(x, u) \qquad\qquad x \in X_i$$

by the proposed abstraction method, we obtain a discrete model $\mathbf{G} = (Z, U, \Sigma_O, \xi, \theta, z_0)$, where Z is the state set, U the event set which consists entirely of controllable events, $\xi : Z \times U \to \mathrm{Pwr}(Z)$ the nondeterministic transition function, $z_0 \in Z$ is the initial state, Σ_O the output event set, $\theta : Z \to \Sigma_O$ the output map. For convenience purposes, assume $\Sigma_O \subseteq Z \cup \{\epsilon_o\}$, where ϵ_o is the silence output event. Let

$$(\forall z \in Z)\, \theta(z) := \begin{cases} z & \text{if output is observable} \\ \epsilon_o & \text{if output is unobservable} \end{cases}$$

In other words, if a state z has observable output, then its output is distinguishable from outputs of all other states. Assume that $\theta(z_0) = z_0 \neq \epsilon_o$, namely the initial state has observable output. Let $Z_m \subseteq Z$ be the desirable state set. For example, in that 2-tank system, we wish that the water levels in both Tanks should be maintained between 3 - 6m. Then any state z that is entirely contained in $[3, 6] \times [3, 6]$ is a desirable state. The control objective is to make sure that once the system's state is within the desirable set, then it should remain in it from then on. If the initial state z_0 is not in Z_m at the beginning, which is common for processing control, then we have to move the system into the desirable zone first. This problem has been discussed in the literature about optimal supervisory control, e.g. [5]. Here we only consider the situation that the initial state z_0 is within the desirable set Z_m. We use state feedback control [12] with partial observation.

Let $\{\epsilon_o\}^*$ be the set of all finite strings which consists entirely of silence event ϵ_o. Let $\Gamma : Z \times U^* \to \mathrm{Pwr}(Z)$ such that for each $z \in Z$ and each $u_1 \cdots u_q \in U^*$ ($q \geq 1$),

$$\begin{aligned} &\Gamma(z, u_1 \cdots u_q) := \{z' \in Z| \\ &\quad (\exists z_1, \cdots, z_q \in Z_m)\;\; z_1 \in \xi(z, u_1) \,\&\, z' = z_q \\ &\quad \& \,[(\forall i \in [1, q-1])\; z_{i+1} \in \xi(z_i, u_{i+1}) \,\&\, \theta(z_i) = \epsilon_o]\} \end{aligned}$$

Let $V : (\theta(Z_m) \cap Z_m) \to \mathrm{Pwr}(U)$ be an enabling map,

$$\begin{aligned} &(\forall z \in (\theta(Z_m) \cap Z_m))\; V(z) := \{u \in U| \\ &\qquad (\forall s \in \{u\}^*)\, s \neq \epsilon \Rightarrow \Gamma(z, s) \cap (Z \backslash Z_m) = \varnothing\} \end{aligned}$$

An enabling map V is applied on $z \in Z_m$ if and only if z has observable output $\theta(z) \neq \epsilon_o$. If no new output is obtained then we don't know whether or not a new state has been reached. So the only thing we can do is to keep the last control action u. In other words, once a control action is taken, it won't change until the next observable output is obtained. Once a control action u is taken at state z, even though the discrete model is non-deterministic, the underlying continuous trajectory starting from a given initial value in z_0 can only pass one state at a time. So the transition behavior at the abstract level is,

$$z(k+1) = \xi(z(k), u(k)) \qquad (k = 0, 1, 2, \cdots)$$

where k represents the kth state in the sequence of states generated by $\mathbf{G}$, $z(0) = z_0$ and $u(0) \in V(z_0)$ because $\theta(z_0) \neq \epsilon_o$. Let $C : Z \times U \to U$ be the control map such that $(\forall z \in Z)(\forall u \in U)$

$$C(z, u) := \begin{cases} u & \theta(z) = \epsilon_o \\ \text{any } u' \in V(z) & \theta(z) \neq \epsilon_o \end{cases}$$

The close loop transition behavior $\mathbf{G}/C$ is as follows,

$$\begin{aligned} z(k+1) &= \xi(z(k), u(k)) \\ u(k+1) &= C(z(k+1), u(k)) \end{aligned}$$

Proposition 3.1 $(k \in \mathbb{N})\, z(k) \in Z_m$.

In the two-tank system we partition the desirable zone $[3, 6] \times [3, 6]$ as shown in Figure 4. By the proposed

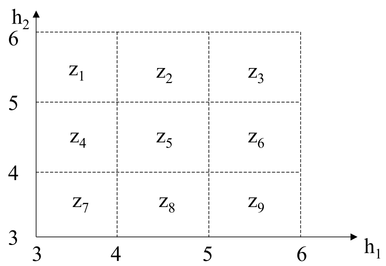

Figure 4: Desirable States for Two-Tank System

abstraction method we obtain the following discrete model as in Figure 5, where $a_1 = [u_1 = 0, u_2 = 0]$, $a_2 = [u_1 = 1, u_2 = 0]$, $a_3 = [u_1 = 0, u_2 = 1]$ and $a_4 = [u_1 = 1, u_2 = 1]$. Circles with dark colors represents states in $Z \backslash Z_m$. Suppose the output event set $\Sigma_O = Z$, namely each state generates an observable output. If the initial state is in $Z_m = \{z_1, \cdots, z_9\}$, then the close loop controlled behavior is shown in Figure 6. So far we have seen that, even though the proposed gradient-based method can only produce a rough model, it may still be able to fulfil control task as shown in the two-tank system.

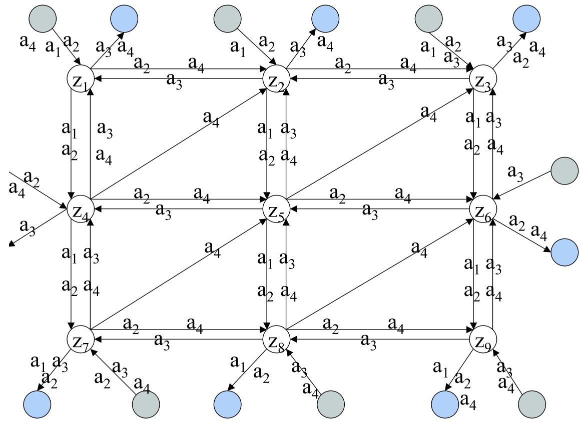

Figure 5: Discrete Model for Two-Tank System in Desirable Zone

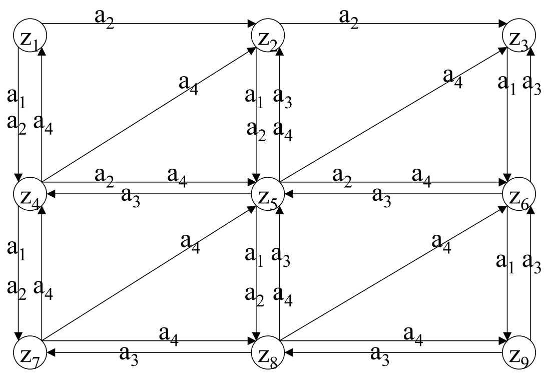

Figure 6: Close Loop Model for Two-Tank System

4 Measure of nondeterminism in discrete-event models

Usually a discrete-event model generated from discrete abstraction is nondeterministic (see Figure 3), namely at some state one transition could end up in a set of states instead of one state. Nondeterminism in a discrete model is caused by the partition on the state space. In this paper we propose a measurement of nondeterminism. If the discrete model is deterministic then given the initial state z_0, a specific input sequence always generates the same sequence of states no matter how many times we run that input sequence. However, if the model is nondeterministic then with the same input sequence, we may get difference sequences of states in different runs. When the total number of test runs is very large, the frequency of each specific sequence of states will approach a fixed number, which is the probability of the occurrence of that specific sequence of states under the same input sequence.

Intuitively we can see that if some sequence of states has very high probability and the rest sequences have relatively low probability, then we may say that the model is very close to a deterministic model. In other words, the higher the difference among probabilities of different sequence of states under the same input sequence, the closer the model to a deterministic one. This intuition suggests us that we should consider each possible input sequence and compute the occurrence probability of each possible output sequence. But clearly this way is not easy to follow since we may have an infinite number of input sequences. So here we take a "weak" version of the above intuition. Under the same input sequence $u_1 \cdots u_n \in U^*$, a sequence of states $z_1, \cdots, z_n \in Z$ has very high probability of occurrence if for each i $(1 \leq i \leq n)$, $\xi(z_{i-1}, u_i)$ has very high probability to reach z_i, namely the prior probability $p(u_i, z_{i-1}|z_i)$ should be very high. Although the product of locally high probabilities may not lead to globally high probability, it is almost true when at each state the difference of local prior probabilities is very high. So we can say that, if at each state z and for each input event u, transition $\xi(z, u)$ is more prone to a specific state z' than to any other state $z'' \in \xi(z, u)$, then the model is more close to a deterministic model. As we know, entropy is a very good measure about such a bias among transitions with the same exit state z and the same event u. So we apply it as follows.

First, we measure the nondeterminism at each state. Assume that we have a prior probability distribution $p : U \times Z \times Z \to [0, 1]$ on each transition, namely $p(u, z'|z)$ is defined if $z' \in \xi(z, u)$. Hence

$$(\forall z \in Z) \quad \sum_{\substack{u, z', \\ z' \in \xi(z,u)}} p(u, z'|z) = 1$$

Let $p(u|z) := \sum_{z' : z' \in \xi(z,u)} p(u, z'|z)$. In practical application, each prior probability $p(u, z'|z)$ can be approximated by the ratio of the number of trajectories from z to z' over the number of all trajectories from z to its neighbor states under input u, which can be obtained by a sufficient number of simulations. Based on this prior probability distribution we can define a local measure of nondeterminism as follows, $(\forall z \in Z)$

$$H_z := - \sum_{\substack{u, \\ \xi(z,u) \neq \varnothing}} p(u|z) \Big[\sum_{\substack{u, z', \\ z' \in \xi(z,u)}} \frac{p(u, z'|z)}{p(u|z)} \log \frac{p(u, z'|z)}{p(u|z)} \Big] \tag{5}$$

We can check that H_z becomes zero when transitions at z are deterministic. Figure 7 depicts three different local transition structures with different measures of nondeterminism, where clearly $0 = H_z^a < H_z^c < H_z^b$. So we can conclude that at state z, transition structure (c) is much closer to the deterministic structure (a) than (b) does.

The global measure of nondeterminism in a discrete-event model can be simply taken as the average over all

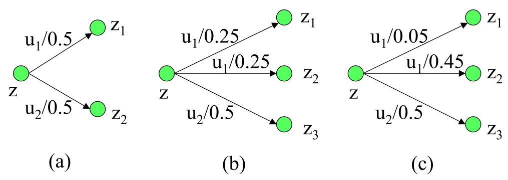

Figure 7: Transition Models with Different Degrees of Nondeterminism

local measures, namely

$$H_{ND} := \frac{\sum_{z \in Z} H_z}{|Z|}$$

where $|Z|$ is the cardinality of Z. Based on this measure we can show that a model DM is deterministic if and only if $H_{ND}^{DM} = 0$, which is equivalent to that $(\forall z \in Z)\, H_z = 0$. We say *a model* DM_1 *is more nondeterministic than a model* DM_2 if $H_{ND}^{DM_1} > H_{ND}^{DM_2}$. In other words. H_{ND} over the degree of nondeterminism of discrete models is nonnegative monotonically increasing.

In the two-tank system let us assume for the sake of example that the prior probability is uniform over *facet-crossing* transitions ($z_j \in \xi(z_i, u) \Rightarrow dim(B(z_i, z_j)) = n-1$) and zero over *non-facet-crossing* transitions ($z_j \in \xi(z_i, u) \Rightarrow dim(B(z_i, z_j)) < n-1$). The reason for zero-probability assignment on non-facet-crossing transitions is that for each state z, the ratio of total number of trajectories starting from z and crossing non-facet boundary over total number of trajectories starting from z is zero. With such probability assignment, the measure of nondeterminism is $H_{ND} = 0.414721$. If we increase the total number of states from 9 to 16 by simply increasing the number of divisions over each tank height from 3 to 4, then $H_{ND} = 0.461403$, which indicates that such partition refinement won't reduce the nondeterminism.

This result can be intuitively explained as follows. In the two-tank discrete model, most states usually have eight neighboring states, except boundary states which have fewer neighbors. The measure of nondeterminism of a state with eight neighbors is higher than that of a boundary state. When we increase the number of states by the proposed way, each non-boundary state still has eight neighbors and the transition structure of such a non-boundary state is the same as in the fewer-state model. Since a finer partition means a larger portion of states with eight neighbors, the global measure of nondeterminism increases. On the other hand, when the total number of states become larger and larger, the portion of states with eight neighbors become dominant. So we expect that the global measure of nondeterminism will asymptotically approach to a fixed number. Figure 8 depicts another example where increasing the number of states from (A) to (B) cannot reduce the measure of nondeterminism.

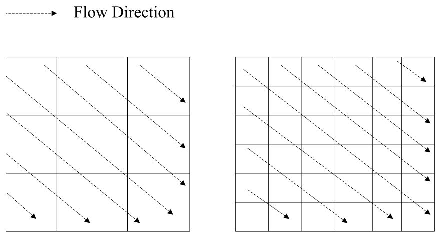

Figure 8: Effect of Partition on Nondeterminism

Although using the proposed way to build a model with more states won't reduce the degree of nondeterminism, it may make it possible for us to do supervisory control if the original fewer cell model cannot. This is nothing to do with nondeterminism, but to do with how much details an abstract model can reveal. For example, if we want to maintain the trajectories within the area of $[3, 9) \times [3, 9)$, then we cannot model this area with one single state because this will leave no place for us to take control actions. So we have to refine our partition and model this area with more states, which will make it possible for us to take control actions to maintain trajectories within these states, hence within the desirable area. On the other hand, when we increase the number of states, if the global measure of nondeterminism increases drastically, then it strongly suggests that the current partition method is not good for control purpose. So this entropy measure offers us a way to evaluate the quality of the current partition method. But it cannot suggest us how to improve a partition if the partition is found not good enough. As for how to efficiently refine a partition to reduce nondeterminism, it is still under investigation.

5 Conclusions

In this paper we have shown that, with a partition consisting of regular polyhedra, we can generate discrete abstraction of a continuous-time system by using gradient analysis on boundaries of each cell in the partition. It is more computationally efficient than other trajectory-based discrete abstraction methods in the literature as long as a continuous-time system with a regular polyhedron state space is concerned. After a discrete model is obtained, we illustrate how to apply supervisory control on such a model. Then considering that the resulting discrete model is usually nondeterministic, we propose an entropy measure on nondeterminism, which can be used as a quality index of the current partition. There are still several problems left unsolved, e.g. how to expand the proposed method to a system whose domain is not a regular polyhedron, and how to reduce nondeterminism after the entropy index indicates that the current partition is not satisfactory. These problems will be discussed in our future work.

References

[1] P. Antsaklis, editor. *Special Issue on Hybrid Systems.* Proceedings of the IEEE. July 2000.

[2] P. Antsaklis, X. Koutsoukos, and J. Zaytoon. On hybrid control of complex systems: a survey. *European Journal of Automation*, 32:1023–1045, 1998.

[3] J. Cury, B. Krogh, and T. Niinomi. Synthesis of supervisory controllers for hybrid systems based on approximating automata. *IEEE Trans. Autom. Control*, 43(4):564568, 1998.

[4] X. D. Koutsoukos, P. J. Antsaklis, J. A. Stiver, and M. D. Lemmon. Supervisory control of hybrid systems. *Proc. IEEE*, 88(7):1026–1049, July 2000.

[5] R. Kumar and V.K. Garg. Optimal supervisory control of discrete event dynamic systems. *SIAM Journal on Control and Optimization*, 33(2):419–439, 1995.

[6] J. Lunze. Qualitative modeling of linear dynamical systems with quantised state measurements. *Automatica*, 30(3):417–431, 1994.

[7] J. Lunze, B. Nixdorf, and J. Schroder. Deterministic discrete-event representations of linear continuous-variable systems. *Automatica*, 35(3):396–406, 1999.

[8] A. Nerode and W. Kohn. Models for hybrid systems: Automata, topologies, controllability, observability. In R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, editors, *Hybrid Systems*, pages 317–356, Berlin, Germany, 1993. Springer-Verlag.

[9] J. Raisch and S. O'Young. A totally ordered set of discrete abstractions for a given hybrid system. In P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry, editors, *Hybrid Systems IV*, pages 342–360, Germany, 1997. Springer-Verlag.

[10] J. Raisch and S. O'Young. Discrete approximation and supervisory control of continuous systems. *IEEE Trans. Autom. Control*, 43(4):568–573, 1998.

[11] J. Stiver, P. Antsaklis, and M. Lemmon. A logical DES approach to the design of hybrid control systems. *Mathl. Comput. Modelling*, 23(11/12):55–76, 1996.

[12] W. M. Wonham. *Notes on Control of Discrete-Event Systems.* ECE Department, University of Toronto, revised 1 July 2002. `http://www.control.utoronto.ca/DES`.