OPTIMAL CONTROL USING BISIMULATIONS

by

Mireille Broucke, Maria Domenica Di Benedetto,
Stefano Di Gennaro and Alberto Sangiovanni-Vincentelli

Memorandum No. UCB/ERL M00/34

22 June 2000

OPTIMAL CONTROL USING BISIMULATIONS

by

Mireille Broucke, Maria Domenica Di Benedetto,
Stefano Di Gennaro and Alberto Sangiovanni-Vincentelli

Memorandum No. UCB/ERL M00/34

22 June 2000

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Optimal control using bisimulations

Mireille Broucke[1], Maria Domenica Di Benedetto[2], Stefano Di Gennaro[2], and Alberto Sangiovanni-Vincentelli[1]

Department of Electrical Engineering and Computer Sciences,
University of California, Berkeley, CA 94720
mire,marika,alberto@eecs.berkeley.edu,digennar@dis.uniroma1.it

Abstract. We consider the synthesis of optimal controls for continuous feedback systems by recasting the problem to a hybrid optimal control problem, which is to synthesize optimal enabling conditions for switching between locations in which the control is constant. An algorithmic solution is obtained by translating the hybrid automaton to a finite automaton using a bisimulation and formulating a dynamic programming problem with extra conditions to ensure non-Zenoness of trajectories. We show that the discrete value function converges to the viscosity solution of the Hamilton-Jacobi-Bellman equation as a discretization parameter tends to zero.

1 Introduction

The goal of this paper is the development of a computationally appealing technique for synthesizing optimal controls for continuous feedback systems $\dot{x} = f(x, u)$, by reducing substantially the complexity of the problem. This goal is achieved by virtue of recasting the problem to a hybrid optimal control problem. The hybrid problem is obtained by approximating the control set $U \subset \mathbb{R}^m$ by a finite set $\Sigma \subset U$ and defining vector fields for the locations of the hybrid system of the form $f(x, \sigma)$, $\sigma \in \Sigma$; that is, the control is constant in each location. The hybrid control problem is, then, to synthesize an optimal switching rule between locations, or equivalently, optimal enabling conditions, such that a target set $\Omega_f \subset \Omega$ is reached while a hybrid cost function is minimized, for each initial condition in a specified set $\Omega \subset \mathbb{R}^n$.

Casting the problem into the domain of hybrid control is not appealing per se, on the contrary! Algorithmic approaches for solving the controller synthesis problem for specific classes of hybrid systems have appeared [17, 28] but no general, efficient algorithm is yet available. Hence, to be able to solve the (nonlinear) hybrid optimal control problem, we must exploit some additional property. We have a feasible and quite appealing approach if we can translate the problem to an equivalent discrete problem, which abstracts completely the continuous behavior. This translation is possible if we can construct a finite *bisimulation* defined on the hybrid state set. The bisimulation can be constructed using the geometric approach reported in [4], based on the following key assumption: $n-1$ *local (on Ω) first integrals can be expressed analytically for each vector field*

$f(x,\sigma)$, $\sigma \in \Sigma$. This assumption is imposed in the transient phase of a feedback system's response, when the vector field is non-vanishing and local first integrals always exist, though analytical expressions for them may not be readily computable. The assumption breaks down at equilibria, thus restricting the region Ω where the method can apply.

If the assumption is met, then we can transform the hybrid system to a quotient system associated with the bisimulation. If the bisimulation is finite, the quotient system is a finite automaton. The control problem posed on the finite automaton is to synthesize a discrete supervisor, providing a switching rule between automaton locations, that minimizes a discrete cost function approximating the original cost function, for each initial discrete state. We provide a dynamic programming solution to this problem, with extra constraints to ensure non-Zenoness of the closed-loop trajectories. By imposing non-Zeno conditions on the synthesis we obtain piecewise constant controls with a finite number of discontinuities in bounded time.

The discrete value function depends on the discretizations of U and of Ω using the bisimulation. We quantify these discretizations by parameters δ and δ_Q, respectively. The main theoretical contribution is to show that as $\delta, \delta_Q \to 0$, the discrete value function converges to the unique viscosity solution of the Hamilton-Jacobi-Bellman (HJB) Equation.

There is a similarity between our approach to optimal control and *regular synthesis*, introduced in [2], in the sense that both restrict the class of controls to a set that has some desired property and both use a finite partition to define switching behavior. For linear systems, the results on regular synthesis are centered on the Bang-Bang principle [19], stating that a sufficient class of optimal controls is piecewise constant. If U is a convex polyhedron, then the number of discontinuities of the control is bounded. There is no hope that general Bang-Bang results are available due to the following example.

Example 1 (Fuller's problem [11]). Consider the optimal control problem

$$\dot{x}_1 = x_2 \tag{1}$$
$$\dot{x}_2 = u \tag{2}$$

with $|u| \leq 1$ and the cost function $J(x,\mu) = \int_0^{T(x,\mu)} x_1^2(s)ds$. If $y \in \mathbb{R}^2$ is any point except the origin, then there exists a unique optimal control driving y to the origin, and it is bang-bang with infinite number of switchings. In fact Kupka has shown in [14] that this phenomenon is generic at sufficiently high dimensions.

Inspite of Fuller's example, in many applications the optimal control is a piecewise continuous function, and therefore methods of regular synthesis of such controls are worth investigating.

Our paper focuses on piecewise constant controls and provides a constructive approach to obtain a cell decomposition by using a finite bisimulation, which further allows us to formulate the synthesis problem on its quotient system - a finite automaton.

The idea of using a time abstract model formed by partitioning the continuous state space has been pursued in a number of papers recently. Lemmon, Antsaklis, Stiver and coworkers [25], [15] use a partition of the state space to convert a hybrid model to a discrete event system (DES). This enables them to apply controller synthesis for DES's to synthesize a supervisor. While our approach is related to this methodology, it differs in that we have explicit conditions for obtaining the partition. In [21] hybrid systems consisting of a linear time-invariant system and a discrete controller that has access to a quantized version of the linear system's output is considered. The quantization results in a rectangular partition of the state space. This approach suffers from spurious solutions that must be trimmed from the automaton behavior.

Hybrid optimal control problems have been studied in papers by Witsenhausen [27] and Branicky, Borkar, Mitter [3]. These studies concentrate on problems of well-posedness, necessary conditions, and existence of optimal solutions but do not provide algorithmic solutions.

The paper is organized as follows. In section 2 we state the optimal control problem, while in section 3 the associated hybrid optimal control problem is given. In Section 4 we review how the bisimulation is constructed. Section 5 formulates the proposed solution using bisimulation and dynamic programming. In section 6 we prove the main theoretical result. In section 7 we study the implementation of an algorithmic solution of the dynamic programming problem including a formal justification of the algorithm's optimality. In section 9 we study several examples of the proposed method. Section 10 summarizes our findings and indicates future directions of research.

2 Optimal control problem

Notation. $\mathbf{1}(\cdot)$ is the indicator function. $cl(A)$ denotes the closure of set A. $\|\cdot\|$ denotes the Euclidean norm. Let $C^1(\mathbb{R}^n)$ and $\mathcal{X}(\mathbb{R}^n)$ denote the sets of continuously differentiable real-valued functions and smooth vector fields on $\mathbb{R}^n$, respectively. $\phi_t(x_0, \mu)$ denotes the trajectory of $\dot{x} = f(x, \mu)$ starting from x_0 and using control $\mu(\cdot)$.

Let U be a compact subset of $\mathbb{R}^m$, Ω an open, bounded, connected subset of $\mathbb{R}^n$, and Ω_f a compact subset of Ω. Define $\mathcal{U}_m$ to be the set of measurable functions mapping $[0,T]$ to U. We define the minimum hitting time $T : \mathbb{R}^n \times \mathcal{U}_m \to \mathbb{R}^+$ by

$$T(x,\mu) := \begin{cases} \infty & \text{if } \{t \mid \phi_t(x,\mu) \in \Omega_f\ \} = \emptyset \\ \min\{t \mid \phi_t(x,\mu) \in \Omega_f\} & \text{otherwise.} \end{cases} \tag{3}$$

A control $\mu \in \mathcal{U}_m$ specified on $[0,T]$ is *admissible* for $x \in \Omega$ if $\phi_t(x,\mu) \in \Omega$ for all $t \in [0,T]$. The set of admissible controls for x is denoted $\mathcal{U}_x$. Let

$$\mathcal{R} := \{\ x \in \mathbb{R}^n \mid \exists \mu \in \mathcal{U}_x.\ T(x,\mu) < \infty\ \}.$$

We consider the following optimal control problem. Given $y \in \Omega$,

$$\begin{aligned} &\text{minimize} && J(y,\mu) = \int_0^{T(y,\mu)} L(x(t),\mu(t))dt + h(x(T(y,\mu))) && (4)\\ &\text{subject to} && \dot{x} = f(x,\mu), \qquad a.e.\ t \in [0,T(y,\mu)] && (5)\\ & && x(0) = y && (6)\end{aligned}$$

among all admissible controls $\mu \in \mathcal{U}_y$. $J : \mathbb{R}^n \times \mathcal{U}_m \to \mathbb{R}$ is the *cost-to-go* function, $h : \mathbb{R}^n \to \mathbb{R}$ is the *terminal cost*, and $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is the *instantaneous cost*. At $T(y,\mu)$ the terminal cost $h(x(T(y,\mu)))$ is incurred and the dynamics are stopped. The control objective is to reach Ω_f from $y \in \Omega$ with minimum cost.

Assumption 2.1.

(1) $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ satisfies $\|f(x',u') - f(x,u)\| \le L_f\left[\|x'-x\| + \|u'-u\|\right]$ for some $L_f > 0$. Let M_f be the upper bound of $\|f(x,u)\|$ on $\Omega \times U$.
(2) $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ satisfies $|L(x',u') - L(x,u)| \le L_L\left[\|x'-x\| + \|u'-u\|\right]$ and $1 \le |L(x,u)| \le M_L$, $x \in \Omega$, $u \in U$, for some $L_L, M_L > 0$.
(3) $h : \mathbb{R}^n \to \mathbb{R}$ satisfies $|h(x') - h(x)| \le L_h\|x' - x\|$ for some $L_h > 0$, and $h(x) \ge 0$ for all $x \in \Omega$. Let M_h be the upper bound of $|h(x)|$ on Ω.

Remark 2.1. These assumptions ensure existence of solutions and continuity of the value function, defined below. Weaker assumptions are possible but since our goal is to introduce a method rather than obtain the most general setting for it, we are satisfied with these. See [1] for other possibilities.

The *value function* or optimal cost-to-go function $V : \mathbb{R}^n \to \mathbb{R}$ is given by

$$V(y) = \inf_{\mu \in \mathcal{U}_y} J(y,\mu)$$

for $y \in \Omega \setminus \Omega_f$, and by $V(y) = h(y)$ for $y \in \Omega_f$. A control μ is called *ϵ-optimal* for x if $J(x,\mu) \le V(x) + \epsilon$.

It is well-known [10] that V satisfies the *Hamilton-Jacobi-Bellman* (HJB) equation

$$-\inf_{u \in U}\left\{L(x,u) + \frac{\partial V}{\partial x} f(x,u)\right\} = 0 \qquad (7)$$

at each point of $\mathcal{R}$ at which it is differentiable. The HJB equation is an infinitesimal version of the equivalent *Dynamic Programming Principle* (DPP) which says that

$$\begin{aligned} V(x) &= \inf_{\mu \in \mathcal{U}_x}\left\{\int_0^t L(\phi_s(x,\mu),\mu(s))ds \;+\; V(\phi_t(x,\mu))\right\}, && x \in \Omega \setminus \Omega_f\\ V(x) &= h(x) && x \in \Omega_f.\end{aligned}$$

The subject of assiduous effort has been that the HJB equation may not have a C^1 solution. This gap in the theory was closed by the inception of the concept of viscosity solution [16, 6], which can be shown to provide the unique solution of

(7) without any differentiability assumption. In particular, a bounded uniformly continuous function V is called a *viscosity solution* of HJB provided, for each $\psi \in C^1(\mathbb{R}^n)$, the following hold:

(i) if $V - \psi$ attains a local maximum at $x_0 \in \mathbb{R}^n$, then

$$-\inf_{u \in U}\left\{L(x_0,u) + \frac{\partial \psi}{\partial x}(x_0) f(x_0,u)\right\} \leq 0,$$

(ii) if $V - \psi$ attains a local minimum at $x_1 \in \mathbb{R}^n$, then

$$-\inf_{u \in U}\left\{L(x_1,u) + \frac{\partial \psi}{\partial x}(x_1) f(x_1,u)\right\} \geq 0.$$

Assumption 2.2. For every $\epsilon > 0$ and $x \in \mathcal{R}$, there exists $N_\epsilon > 0$ and an admissible piecewise constant ϵ-optimal control μ having at most N_ϵ discontinuities and such that $\phi_t(x,\mu)$ is transverse to $\partial \Omega_f$.

The transversality assumption implies that the viscosity solution is continuous at the boundary of the target set, a result needed in proving uniform continuity of V over a finite horizon. The assumption can be replaced by a small-time controllability condition. For a treatment of small time controllability and compatibility of the terminal cost with respect to continuity of the value function, see [1]. The finite switching assumption holds under mild assumptions such as Lipschitz continuity of the vector field and cost functions, and is based on approximating measurable functions by piecewise constant functions.

3 Hybrid system

The approach we propose for solving the continuous optimal control problem first requires a mapping to a hybrid system and, second, employs a bisimulation of the hybrid system to formulate a dynamic programming problem on the quotient system. In this section we define the hybrid optimal control problem. First, we discretize U by defining a finite set $\Sigma_\delta \subset U$ which has a mesh size

$$\delta := \sup_{u \in U} \min_{\sigma \in \Sigma_\delta} \|u - \sigma\|.$$

We define the hybrid automaton $H := (\Sigma \times \mathbb{R}^n, \Sigma_\delta, D, E_h, G, R)$ with the following components:

State set $\Sigma \times \mathbb{R}^n$ consists of the finite set $\Sigma = \Sigma_\delta \cup \{\sigma_f\}$ of control locations and n continuous variables $x \in \mathbb{R}^n$. σ_f is a terminal location when the continuous dynamics are stopped (in the same sense that the dynamics are "stopped" in the continuous optimal control problem).

Events Σ_δ is a finite set of control event labels.

Vector fields $D : \Sigma \to \mathcal{X}(\mathbb{R}^n)$ is a function assigning an autonomous vector field to each location. We use the notation $D(\sigma) = f_\sigma$.

Control switches $E_h \subset \Sigma \times \Sigma$ is a set of control switches. $e = (\sigma, \sigma')$ is a directed edge between a source location σ and a target location σ'. If $E_h(\sigma)$ denotes the set of edges that can be enabled at $\sigma \in \Sigma$, then $E_h(\sigma) := \{(\sigma, \sigma') \mid \sigma' \in \Sigma \setminus \sigma\}$ for $\sigma \in \Sigma_\delta$ and $E_h(\sigma_f) = \emptyset$. Thus, from a source location not equal to σ_f, there is an edge to every other location (but not itself), while location σ_f has no outgoing edges.

Enabling conditions $G : E_h \to \{g_e\}_{e \in E_h}$ is a function assigning to each edge an enabling (or guard) condition $g \subset \mathbb{R}^n$. We use the notation $G(e) = g_e$.

Reset conditions $R : E_h \to \{r_e\}_{e \in E_h}$ is a function assigning to each edge a reset condition, $r_e : \mathbb{R}^n \to 2^{\mathbb{R}^n}$, where we use the notation $R(e) = r_e$.

3.1 Semantics

A state is a pair (σ, x), $\sigma \in \Sigma$ and $x \in \mathbb{R}^n$. In location $\sigma \in \Sigma_\delta$ the continuous state evolves according to the vector field $f(x, \sigma)$. In location σ_f, the vector field is $\dot{x} = f(x, \mu_f)$ where μ_f is the (not necessarily constant) control of the terminal location. Trajectories of H evolve in *steps* of two types. A *σ-step* is a binary relation $\xrightarrow{\sigma} \subset (\Sigma \times \mathbb{R}^n) \times (\Sigma \times \mathbb{R}^n)$, and we write $(\sigma, x) \xrightarrow{\sigma'} (\sigma', x')$ iff (1) $e = (\sigma, \sigma') \in E_h$, (2) $x \in g_e$, and (3) $x' = r_e(x)$. We assume the transition $(\sigma, x) \xrightarrow{\sigma'} (\sigma', x')$ is taken at the first time in location σ when the control event label is σ' and $x \in g_e$ for $e = (\sigma, \sigma')$. A *t-step* is a binary relation $\xrightarrow{t} \subset (\Sigma \times \mathbb{R}^n) \times (\Sigma \times \mathbb{R}^n)$, and we write $(\sigma, x) \xrightarrow{t} (\sigma', x')$ iff (1) $\sigma = \sigma'$, and (2) for $t \geq 0$, $x' = \phi_t(x, \sigma)$, where $\dot{\phi}_t(x) = f(\phi_t(x, \sigma), \sigma)$. A *hybrid control* is a finite or infinite sequence of labels $\omega = \omega_1 \omega_2 \ldots$, with $\omega_i \in \Sigma \cup \mathbb{R}^+$. $\omega_i \in \mathbb{R}^+$ is the duration of the t-step at step i. The set of hybrid controls is denoted $\mathcal{S}$. A *hybrid trajectory* π over $\omega \in \mathcal{S}$ is a finite or infinite sequence $\pi : (\sigma_0, x_0) \xrightarrow{\omega_1} (\sigma_1, x_1) \xrightarrow{\omega_2} (\sigma_2, x_2) \xrightarrow{\omega_3} \ldots$ where $(\sigma_i, x_i) \in \Sigma \times \mathbb{R}^n$. Trajectory π is *accepted* by H iff $\forall i$, $(\sigma_i, x_i) \xrightarrow{\omega_{i+1}} (\sigma_{i+1}, x_{i+1})$ is either a t-step or σ-step of H. Let π be the trajectory (not necessarily accepted by H) starting at $(\sigma, x) \in \Sigma \times \Omega$ and defined over $\omega \in \mathcal{S}$. We say ω is *admissible* for (σ, x) on interval $[0, T]$ if (1) π remains in $\Sigma \times \Omega$ for $t \in [0, T]$, and (2) corresponding to ω is a piecewise constant control $\mu_\omega(t)$ (with a finite number of discontinuities in finite time). Let $\mathcal{S}_{(\sigma, x)}$ be the set of admissible controls for (σ, x).

Example 2. Consider a time optimal control problem for

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = u.$$

We select $\Omega = (-1, 1) \times (-1, 1)$ and $\Omega_f = \overline{B}_\epsilon(0)$, the closed epsilon ball centered at 0. The cost-to-go function is $J(x, \mu) = \int_0^{T(x,\mu)} dt$ and $U = \{u \ : \ |u| \leq 1\}$. We select $\Sigma_\delta = \{-1, 1\}$, so that $\delta = 1$. The hybrid system is show in Figure 1. The state set is $\{\sigma_{-1} = -1, \sigma_1 = 1, \sigma_f\} \times \mathbb{R}^n$. $g_{e_{-1}}$ and g_{e_1} are unknown and must be synthesized, while $g_{e_2} = g_{e_3} = \Omega_f$.

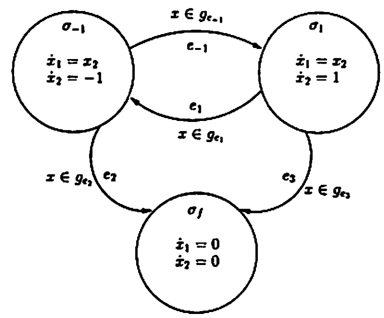

Fig. 1. Hybrid automaton for time optimal control of a double integrator system

3.2 Hybrid optimal synthesis

We want to synthesize enabling conditions so that for each $y \in \mathcal{R}$, the cost-to-go from y well-approximates the viscosity solution at y of HJB. This requires posing a hybrid optimal synthesis problem. We define a *hybrid cost-to-go function* $J_H : \Sigma \times \mathbb{R}^n \times \mathcal{S} \to \mathbb{R}$ as follows. For $\omega \in \mathcal{S}_{(\sigma,x)}$,

$$J_H((\sigma, x), \omega) = J(x, \mu_\omega).$$

The *hybrid value function* $V_H : \Sigma \times \mathbb{R}^n \to \mathbb{R}$ is

$$V_H((\sigma, x)) = \inf_{\omega \in \mathcal{S}_{(\sigma,x)}} J_H((\sigma, x), \omega).$$

Hybrid optimal synthesis problem:
Given H and $0 < \epsilon^1 < \epsilon^2$, synthesize g_e, $e \in E_h$, subject to:

1. *$g_e = \Omega_f$ if $e = (\sigma, \sigma_f)$, $\sigma \in \Sigma_\delta$.*
2. *For each $e \in E_h$, $g_e \subseteq \Omega$.*
3. *For all $\omega \in \mathcal{S}$ and $(\sigma, x) \in \Sigma \times \Omega$ such that $V_H((\sigma, x)) < \infty$, $\pi_{(\sigma,x)}$ is accepted by H if ω is admissible and ϵ^1-optimal for (σ, x).*
4. *For all $\omega \in \mathcal{S}$ and $(\sigma, x) \in \Sigma \times \Omega$, $\pi_{(\sigma,x)}$ is not accepted by H if either ω is not admissible for (σ, x), ω is not ϵ^2-optimal for (σ, x), or $V_H((\sigma, x)) = \infty$.*

Remark 3.1. Condition 1 says that the enabling condition for edges going to the final location is Ω_f. Condition 2 corresponds to trajectories remaining in Ω. Conditions 3 and 4 say the hybrid automaton "does the right thing".

4 Construction of bisimulation

We propose to solve the hybrid optimal control problem using the bisimulation of H. In this section we define bisimulation and the quotient system that is obtained from it.

Let λ represent an arbitrary time interval. corresponding to some $t \in \mathbb{R}^+$. A *bisimulation* of H is an equivalence relation $\simeq \subset (\Sigma_\delta \times \mathbb{R}^n) \times (\Sigma_\delta \times \mathbb{R}^n)$ such that for all states $p_1, p_2 \in \Sigma_\delta \times \mathbb{R}^n$, if $p_1 \simeq p_2$ and $\sigma \in \Sigma_\delta \cup \{\lambda\}$, then if $p_1 \xrightarrow{\sigma} p_1'$, there exists p_2' such that $p_2 \xrightarrow{\sigma} p_2'$ and $p_1' \simeq p_2'$. If $\simeq$ is finite, the quotient system is a finite automaton. The finite automaton can be used to study properties of the reach set of H. For an overview of results on bisimulations for hybrid systems, see [13].

Since the dynamics are restricted to the set Ω, the set of interesting equivalence classes of $\simeq$, denoted Q, are those that intersect $\Sigma_\delta \times cl(\Omega)$. For each $q \in Q$ we define a distinguished point $(\sigma, \xi) \in q$. We associate q with its distinguished point by the notation $q = [(\sigma, \xi)]$. It is now possible to define the enabling and reset conditions of H in terms of Q. In particular, the enabling conditions of H are synthesized as subsets of Q while the reset conditions are defined as follows. For $e = (\sigma, \sigma')$

$$r_e(x) = \{\ y \mid \exists \xi . [(\sigma, x)] = [(\sigma, \xi)] \wedge [(\sigma', \xi)] = [(\sigma', y)]\ \}. \tag{8}$$

That is, $r_e(x)$ is the projection to $\mathbb{R}^n$ of the set of equivalence classes $[(\sigma', y)]$ such that the projection to $\mathbb{R}^n$ of $[(\sigma', y)]$ and $[(\sigma, x)]$ have nonempty intersection. This definition in effect gives an over-approximation of the identity map in terms of the equivalence classes of $\simeq$ and will introduce non-determinacy in the finite automaton. Notice also that (8) encodes information about the bisimulation in H. This sequence of steps is not typical; it is characteristic of our synthesis procedure. We define a mesh size on Q by

$$\delta_Q = \max_{q \in Q} \sup_{(\sigma,x),(\sigma,y) \in q} \{\|x - y\|\}.$$

Finally, for each $q = [(\sigma, \xi)] \in Q$ we associate the duration τ_q, the maximum time to traverse q using constant control σ. That is,

$$\tau_q = \sup_{(\sigma,x),(\sigma,y) \in q} \{\ t \mid y = \phi_t(x, \sigma)\ \}.$$

4.1 Review of geometric construction

We briefly review a method for obtaining bisimulations [4] which relies on the following (related) assumptions on the vector fields on $cl(\Omega)$.

Assumption 4.1.

(1) $n-1$ first integrals can be defined analytically on Ω for each $f(x, \sigma)$, $\sigma \in \Sigma_\delta$.
(2) There exists $m_f > 0$ such that $\|f(x, u)\| \geq m_f$ for all $x \in cl(\Omega)$, $u \in U$.

A bisimulation of $\Sigma_\delta \times \mathbb{R}^n$ is constructed using a set of simple, co-dimension one tangential foliations with associated submersions $\gamma_i^\sigma(x) = y_i^\sigma$, $i = 1, \ldots, n-1$ and a simple co-dimension one transversal foliation with submersion $\gamma_n^\sigma = y_n^\sigma$, such that $(y_1^\sigma, \ldots, y_n^\sigma)$ form a set of Euclidean coordinates $\gamma^\sigma : [-1,1]^n \rightarrow V \subseteq \Omega$

for each $\sigma \in \Sigma_\delta$. We *discretize* the foliations by selecting a finite set of leaves. Fix $k \in \mathbb{Z}^+$ and let $\Delta = \frac{1}{2^k}$. Define

$$C_k = \{0, \pm\Delta, \pm 2\Delta, \dots, \pm 1\}. \tag{9}$$

Each $y_i^\sigma = c$ for $c \in C_k$, $i = 1, \dots, n$ defines a hyperplane in $\mathbb{R}^n$ denoted $\tilde{W}_{i,c}^\sigma$, and a submanifold $W_{i,c}^\sigma = (\gamma^\sigma)^{-1}(\tilde{W}_{i,c}^\sigma)$. The collection of submanifolds for $\sigma \in \Sigma_\delta$ is

$$\mathcal{W}_k^\sigma = \{\ W_{i,c}^\sigma \mid c \in C_k, i \in \{1, \dots, n\}\ \}. \tag{10}$$

$\Omega \setminus \mathcal{W}_k^\sigma$ is the union of $2^{n(k+1)}$ disjoint open sets $\mathcal{V}_k^\sigma = \{V_j^\sigma\}$. We define an equivalence relation $\simeq^e$ on $\mathbb{R}^n$ as follows. $x \simeq^e x'$ iff
(1) $x \notin [-1,1]^n$ iff $x' \notin [-1,1]^n$, and
(2) if $x, x' \in [-1,1]^n$, then for each $i = 1, \dots, n$, $x_i \in (c, c+\Delta)$ iff $x' \in (c, c+\Delta)$, and $x_i = c$ iff $x_i' = c$, for all $c \in C_k$.
We define the equivalence relation $\simeq$ on $\Sigma_\delta \times \mathbb{R}^n$ as follows. $(\sigma, x) \simeq (\sigma', x')$ iff (1) $\sigma = \sigma'$, and (2) $\gamma^\sigma(x) \simeq^e \gamma^\sigma(x')$.

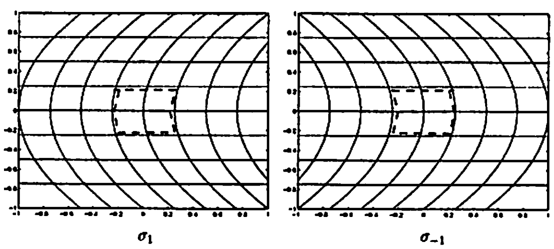

Fig. 2. Partitions for states σ_1 and σ_{-1} of the hybrid automaton of Figure 1

Example 3. Continuing example 2, a first integral for vector field $\dot{x}_1 = x_2, \dot{x}_2 = 1$ is $x_1 - \frac{1}{2}x_2^2 = c_1$, $c_1 \in \mathbb{R}$. For $\dot{x}_1 = x_2$, $\dot{x}_2 = -1$ a first integral is $x_1 + \frac{1}{2}x_2^2 = c_2$, $c_2 \in \mathbb{R}$. We select a transverse foliation for each vector field, given by $x_2 = c_3$. A possible set of partitions for locations σ_1 and σ_{-1} and $\Omega = (-1,1) \times (-1,1)$ are shown in Figure 2. The equivalence classes of $\simeq$ are pairs consisting of a control label in Σ_δ and the interiors of regions, open line segments and curves forming the boundaries of two regions, and the points at the corners of regions. $\tau = 0$ for the segments transverse to the flow and the corner points. $\tau = \Delta$ for the interiors of regions and segments tangential to the flow, where $\Delta = .25$ in Figure 2.

5 Discrete problem

In this section we transform the hybrid optimal control problem to a dynamic programming problem on a non-deterministic finite automaton, for which an algorithmic solution may be found. Consider the class of non-deterministic automata with cost structure represented by the tuple

$$A = (Q, \Sigma_\delta, E, obs, Q_f, \hat{L}, \hat{h}).$$

Q is the state set, as above, and Σ_δ is the set of control labels as before. $obs : E \to \Sigma_\delta$ is a map that assigns a control label to each edge and is given by $obs(e) = \sigma'$, where $e = (q, q')$, $q = [(\sigma, \xi)]$ and $q' = [(\sigma', \xi')]$. Q_f is the target set given by the over-approximation of Ω_f,

$$Q_f = \{q \in Q \mid \exists x \in \Omega_f \,.\, (\sigma, x) \in q\}. \tag{11}$$

$E \subseteq Q \times Q$ is the transition relation encoding t-steps and σ-steps of H. A will be used to synthesize g_e of H, so, in the spirit of [22], E includes all possible edges between locations. The synthesis procedure on A will involve trimming undesirable edges. Thus, $(q, q') \in E$, where $q = [(\sigma, \xi)]$ and $q' = [(\sigma', \xi')]$ if either (a) $\sigma = \sigma'$, there exists $x \in \Omega$ such that $(\sigma, x) \in q$, and there exists $\tau > 0$ such that $\forall t \in [0, \tau]$, $(\sigma, \phi_t(x, \sigma)) \in q$ and $(\sigma, \phi_{\tau+\epsilon}(x, \sigma)) \in q'$ for arbitrarily small $\epsilon > 0$, or (b) $\sigma = \sigma'$, there exists $x \in \Omega$ such that $(\sigma, x) \in q$, and there exists $\tau > 0$ such that $\forall t \in [0, \tau)$, $(\sigma, \phi_t(x, \sigma)) \in q$ and $(\sigma, \phi_\tau(x, \sigma)) \in q'$, or (c) $\sigma \neq \sigma'$ and there exists $x \in \Omega$ such that $(\sigma, x) \in q$ and $(\sigma', x) \in q'$. Cases (a) and (b) say that from a point in q, q' is the first state (different from q) reached after following the flow of $f(x, \sigma)$ for some time. Case (c) says that an edge exists between q and q' if their projections to $\mathbb{R}^n$ have non-empty intersection.

Let $e = (q, q')$ with $q = [(\sigma, \xi)]$ and $q' = [(\sigma', \xi')]$. $\hat{L} : E \to \mathbb{R}$ is the *discrete instantaneous cost* given by

$$\hat{L}(e) := \begin{cases} \tau_q L(\xi, \sigma) & \text{if } \sigma = \sigma' \\ 0 & \text{if } \sigma \neq \sigma'. \end{cases} \tag{12}$$

Thus, no cost is incurred for control switches. $\hat{h} : Q \to \mathbb{R}$ is the *discrete terminal cost* given by

$$\hat{h}(q) := h(\xi).$$

The domain of $\hat{h}$ can be extended to Ω, with a slight abuse of notation, by

$$\hat{h}(x) := \hat{h}(q) \tag{13}$$

where $q = \arg\min_{q'}\{\|x - \xi'\| \mid q' = [(\sigma', \xi')]\}$.

5.1 Semantics

A transition or *step* of A from $q = [(\sigma, \xi)] \in Q$ to $q' = [(\sigma', \xi')] \in Q$ with observation $\sigma' \in \Sigma_\delta$ is denoted $q \xrightarrow{\sigma'} q'$. If $\sigma \neq \sigma'$ the transition is referred to as a *control switch*; otherwise, it is referred to as a *time step*. If $E(q)$ is the set of edges that can be enabled from $q \in Q$, then for $\sigma \in \Sigma_\delta$,

$$E_\sigma(q) = \{e \in E(q) \mid obs(e) = \sigma\}.$$

If $|E_\sigma(q)| > 1$, then we say that $e \in E_\sigma(q)$ is *unobservable* in the sense that when control event σ is issued, it is unknown which edge among $E_\sigma(q)$ is taken. If $\sigma = \sigma'$, then $|E_\sigma(q)| = 1$, by the uniqueness of solutions of ODE's and by the definition of bisimulation.

A *control policy* $c : Q \rightarrow \Sigma_\delta$ is a map assigning a control event to each state; $c(q) = \sigma$ is the control event issued when the state is at q. A *trajectory* π of A over c is a sequence $\pi = q_0 \stackrel{\sigma_1}{\rightarrow} q_1 \stackrel{\sigma_2}{\rightarrow} q_2 \stackrel{\sigma_3}{\rightarrow} \ldots$, $q_i \in Q$. A trajectory is *non-Zeno* if between any two non-zero duration time steps there are a finite number of control switches and zero duration time steps. Let $\Pi_c(q)$ be the set of trajectories starting at q and applying control policy c, and let $\check{\Pi}_c(q)$ be the set of trajectories starting at q, applying control policy c, and eventually reaching Q_f. If for every $q \in Q$, $\pi \in \Pi_c(q)$ is non-Zeno then we say c is an *admissible control policy.* The set of all admissible control policies for A is denoted $\mathcal{C}$.

A control policy c is said to have a *loop* if A has a trajectory $q_0 \stackrel{c(q_0)}{\rightarrow} q_1 \stackrel{c(q_1)}{\rightarrow} \ldots \stackrel{c(q_{m-1})}{\rightarrow} q_m = q_0$, $q_i \in Q$. A control policy has a *Zeno loop* if it has a loop made up of control switches and/or zero duration time steps (i.e. $\tau_q = 0$) only.

Lemma 1. *A control policy c for non-deterministic automaton A is admissible if and only if it has no Zeno loops.*

Proof. First we show that a non-deterministic automaton with non-Zeno trajectories has a control policy without Zeno loops. For suppose not. Then a trajectory starting on a state belonging to the loop can take infinitely many steps around the loop before taking a non-zero duration time step. This trajectory is not non-Zeno, a contradiction. Second, we show that a control policy without Zeno loops implies non-Zeno trajectories. Suppose not. Consider a Zeno trajectory that takes an infinite number of control switches and/or zero duration time steps between two non-zero duration time steps. Because there are a finite number of states in Q, by the Axiom of Choice, one of the states must be repeated in the sequence of states visited during the control switches and/or zero duration time steps. This implies the existence of a loop in the control policy. Either each step of the loop is a control switch, implying a Zeno loop. Or the loop has one or more zero duration time steps. But the bisimulation partition permits zero duration time steps if $\tau_q = 0$, which implies a Zeno loop.

Example 4. Consider the automaton in Figure 3. If we are at q_1 and the control $\sigma'\sigma'\sigma$ is issued, then three possible trajectories are $q_1 \stackrel{\sigma'}{\rightarrow} q_3 \stackrel{\sigma'}{\rightarrow} q_4 \stackrel{\sigma}{\rightarrow} q_2$, $q_1 \stackrel{\sigma'}{\rightarrow} q_4 \stackrel{\sigma'}{\rightarrow} q_5 \stackrel{\sigma}{\rightarrow} q_2$, or $q_1 \stackrel{\sigma'}{\rightarrow} q_3 \stackrel{\sigma'}{\rightarrow} q_4 \stackrel{\sigma}{\rightarrow} q_1$. The first trajectory has a zero duration time step. The control is inadmissible since the last trajectory has a Zeno loop.

5.2 Dynamic programming

In this section we formulate the dynamic programming problem on A. This involves defining a cost-to-go function and a value function that minimizes it over control policies suitable for non-deterministic automata.

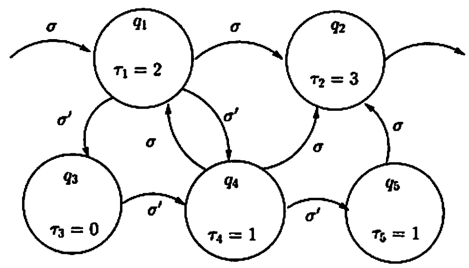

Fig. 3. Fragment of automaton with a zero duration time step.

Let $\pi = q_0 \xrightarrow{\sigma_1} q_1 \rightarrow \ldots \rightarrow q_{N-1} \xrightarrow{\sigma_N} q_N$, where $q_i = [(\sigma_i, \xi_i)]$ and π takes the sequence of edges $e_1 e_2 \ldots e_N$. We define a *discrete cost-to-go* $\hat{J} : Q \times C \rightarrow \mathbb{R}$ by

$$\hat{J}(q,c) = \begin{cases} \max_{\pi \in \tilde{\Pi}_c(q)} \left\{ \sum_{j=1}^{N_\pi} \hat{L}(e_j) + \hat{h}(q_{N_\pi}) \right\} & \text{if } \Pi_c(q) = \tilde{\Pi}_c(q) \\ \infty & \text{otherwise} \end{cases}$$

where $N_\pi = \min\{j \geq 0 \mid q_j \in Q_f\}$. We take the maximum over $\tilde{\Pi}_c(q)$ because of the non-determinacy of A: it is uncertain which among the (multiple) trajectories allowed by c will be taken so we must assume the worst-case situation. The *discrete value function* $\hat{V} : Q \rightarrow \mathbb{R}$ is

$$\hat{V}(q) = \min_{c \in C} \hat{J}(q,c)$$

for $q \in Q \setminus Q_f$ and $\hat{V}(q) = \hat{h}(q)$ for $q \in Q_f$. We show in Proposition 1 that $\hat{V}$ satisfies a DPP that takes into account the non-determinacy of A and ensures that optimal control policies are admissible. This DPP describes the accumulation of cost over one step to be the worst case cost among edges that have the same label. Let $\mathcal{A}_q$ be the set of control assignments $c(q) \in \Sigma_\delta$ at q such that c is admissible.

Proposition 1. *$\hat{V}$ satisfies*

$$\hat{V}(q) = \min_{c(q) \in \mathcal{A}_q} \left\{ \max_{e=(q,q') \in E_{c(q)}(q)} \{\hat{L}(e) + \hat{V}(q')\} \right\}, \quad q \in Q \setminus Q_f \tag{14}$$

$$\hat{V}(q) = \hat{h}(q), \qquad q \in Q_f. \tag{15}$$

Proof. Fix $q \in Q$. By definition of $\hat{J}$

$$\hat{J}(q,c) = \max_{e=(q,q') \in E_{c(q)}(q)} \{\hat{L}(e) + \hat{J}(q',c)\}. \tag{16}$$

By definition of $\hat{V}$

$$\hat{J}(q,c) \geq \max_{e=(q,q') \in E_{c(q)}(q)} \{\hat{L}(e) + \hat{V}(q')\}.$$

Since $c(q) \in \mathcal{A}_q$ is arbitrary

$$\hat{V}(q) \geq \min_{c(q)\in\mathcal{A}_q} \Big\{ \max_{e=(q,q')\in E_{c(q)}(q)} \{\hat{L}(e) + \hat{V}(q')\} \Big\}.$$

To prove the reverse inequality suppose, by way of contradiction, there exists $\sigma' \in \Sigma_\delta$ such that

$$\hat{V}(q) > \max_{e=(q,q')\in E_{\sigma'}(q)} \{\hat{L}(e) + \hat{V}(q')\} := \hat{L}(e) + \hat{V}(\overline{q}). \tag{17}$$

Suppose the optimal admissible policy for $\overline{q}$ is $\overline{c}$. Define $c = \overline{c}$ on $Q \setminus \{q\}$ and $c(q) = \sigma'$. Then $\hat{J}(q,c) = \hat{L}(e) + \hat{V}(\overline{q}) < \hat{V}(q)$. This gives rise to a contradiction if we can show c is admissible. Suppose not. Then there exists a loop of control switches and zero duration time steps containing q and $\overline{q}$, implying $\hat{V}(\overline{q}) \geq \hat{V}(q)$, which contradicts hypothesis (17). □

Remark 5.1. The DPP for $\hat{V}$ is a prescription for synthesizing admissible control policies, but we have not indicated how, in practice, this can be achieved. One possibility is to introduce a fictitious switching cost in the formulation of $\hat{V}$. Capuzzo-Dolcetta and Evans [5] introduce a small switching cost which tends to zero as $\delta \to 0$. Alternatively, admissible controls can be obtained through a device introduced in implementation. For example, a counter of the number of switches could be used. We will propose an algorithmic solution guaranteed not to generate Zeno loops in Section 7.

5.3 Synthesis of g_e

The synthesis of enabling conditions or *controller synthesis* is typically a post-processing step of a backward reachability analysis (see, for example, [28]). This situation prevails here as well: equations (14)-(15) describe a backward analysis to construct an optimal policy $c \in \mathcal{C}$. Once c is known the enabling conditions of H are extracted as follows.

Consider each $e = (\sigma, \sigma') \in E$ of H with $\sigma \neq \sigma'$. There are two cases. If $\sigma' \neq \sigma_f$ then $g_e = \{x \mid (\sigma, x) \in q, q \in Q \ \wedge \ c(q) = \sigma'\}$. That is, if the control policy designates switching from $q \in Q$ with label σ to $q' \in Q$ with label σ', then the corresponding enabling condition in H includes the projection to $\mathbb{R}^n$ of q. The second case when $\sigma' = \sigma_f$ is for edges going to the terminal location of H. Then $g_e = \{x \mid (\sigma, x) \in q, q \in Q_f\}$.

6 Main Result

We will prove that $\hat{V}$ converges to V, the viscosity solution of the HJB equation, as $\delta_Q, \delta \to 0$. The proof will be carried out in three steps. In the first step we consider restricting the set of controls to piecewise constant functions, whose constant intervals are a function of the state. In the second step we introduce the discrete approximations of L and h. In the last step we introduce the discrete states Q and consider the non-determinacy of A.

In the sequel we make use of a filtration of control sets $\Sigma_k \equiv \Sigma_{\delta_k}$ corresponding to a sequence $\delta_k \to 0$ as $k \to \infty$, in such a manner that $\Sigma_k \subset \Sigma_{k+1}$. Considering (10), we define a filtration of families of submanifolds such that $\mathcal{W}_k^\sigma \subset \mathcal{W}_{k+1}^\sigma$, for each $\sigma \in \Sigma_k$.

Step 1: piecewise constant controls.

In the first step we define a class of piecewise constant functions that depend on the state and show that the value function which minimizes the cost-to-go over this class converges to the viscosity solution of HJB as $\delta_k \to 0$. The techniques of this step are based on those in Bardi and Capuzzo-Dolcetta [1] and are related to those in [5].

We consider the optimal control problem (4)-(6) when the set of admissible controls is $\mathcal{U}_k^1$, piecewise constant functions consisting of finite sequences of control labels $\sigma \in \Sigma_k$ and each σ is applied for a time $\tau(\sigma, x)$. Let $(\sigma, x) \in q$ for $q \in Q$ and define $\tau(\sigma, x)$ to be the minimum of the time it takes the trajectory starting at x and using control $\sigma \in \Sigma_k$ to reach (ta) $\partial\Omega_f$, or (tb) some x' such that $(\sigma, x') \notin q$. If a trajectory is at x_i at the start of the $(i+1)$th step, then the control σ_{i+1} is applied for time $\tau_{i+1} := \tau(\sigma_{i+1}, x_i)$ and $x_{i+1} = \phi_{\tau_{i+1}}(x_i, \sigma_{i+1})$.

Let

$$\mathcal{R}_k^1 := \{ x \in \mathbb{R}^n \mid \exists \mu \in \mathcal{U}_k^1 \, . \, T(x, \mu) < \infty \}.$$

We define the cost-to-go function $J_k^1 : \Omega \times \mathcal{U}_k^1 \to \mathbb{R}$ as follows. For $x \in \Omega$ and $\mu = \sigma_1\sigma_2 \ldots \in \mathcal{U}_k^1$, if $T(x, \mu) < \infty$ then

$$J_k^1(x, \mu) = \sum_{j=1}^{N} \int_0^{\tau(\sigma_j, x_{j-1})} L(\phi_s(x_{j-1}, \sigma_j), \sigma_j) ds + h(x_N)$$

where $N = \min\{j \geq 0 \mid x_j \in \partial\Omega_f\}$. $J_k^1(x, \mu) = \infty$, otherwise. We define the value function $V_k^1 : \mathbb{R}^n \to \mathbb{R}$ as follows. For $x \in \Omega \setminus \Omega_f$,

$$V_k^1(x) = \inf_{\mu \in \mathcal{U}_k^1} J_k^1(x, \mu) \tag{18}$$

and for $x \in \Omega_f$, $V_k^1(x) = h(x)$.

Proposition 2. *V_k^1 satisfies, for all $x \in \mathcal{R}_k^1$,*

$$V_k^1(x) = \min_{\sigma \in \Sigma_k} \left\{ \int_0^{\tau(\sigma, x)} L(\phi_s(x, \sigma), \sigma) ds + V_k^1(\phi_{\tau(\sigma, x)}(x, \sigma)) \right\}. \tag{19}$$

Proof. Fix $x \in \mathcal{R}_k^1$ and $\mu = \sigma\sigma_1\sigma_2 \ldots \in \mathcal{U}_k^1$. Using the semigroup property of flows and the definition of J_k^1

$$J_k^1(x, \mu) = \int_0^{\tau(\sigma, x)} L(\phi_s(x, \sigma), \sigma) ds + J_k^1(\phi_{\tau(\sigma, x)}(x, \sigma), \overline{\mu}) \tag{20}$$

where $\overline{\mu} = \sigma_1\sigma_2\ldots \in \mathcal{U}_k^1$. By definition of V_k^1

$$J_k^1(x,\mu) \geq \int_0^{\tau(\sigma,x)} L(\phi_s(x,\sigma),\sigma)ds + V_k^1(\phi_{\tau(\sigma,x)}(x,\sigma)).$$

Hence,

$$V_k^1(x) \geq \min_{\sigma\in\Sigma_k}\left\{\int_0^{\tau(\sigma,x)} L(\phi_s(x,\sigma),\sigma)ds + V_k^1(\phi_{\tau(\sigma,x)}(x,\sigma))\right\}.$$

To prove the reverse inequality fix $\sigma \in \Sigma_k$, set $z = \phi_{\tau(\sigma,x)}(x,\sigma)$, and fix $\epsilon > 0$ and $\mu_z \in \mathcal{U}_k^1$ such that

$$V_k^1(z) \geq J_k^1(z,\mu_z) - \epsilon.$$

Define the control

$$\overline{\mu}(s) = \begin{cases} \sigma & s \leq \tau(\sigma,x) \\ \mu_z(s-t) & s > \tau(\sigma,x). \end{cases}$$

Then

$$\begin{aligned} V_k^1(x) \leq J_k^1(x,\overline{\mu}) &= \int_0^{\tau(\sigma,x)} L(\phi_s(x,\sigma),\sigma)ds + J_k^1(z,\mu_z) \\ &\leq \int_0^{\tau(\sigma,x)} L(\phi_s(x,\sigma),\sigma)ds + V_k^1(z) + \epsilon. \end{aligned}$$

Since $\sigma \in \Sigma_k$ and $\epsilon > 0$ are arbitrary

$$V_k^1(x) \leq \min_{\sigma\in\Sigma_k}\left\{\int_0^{\tau(\sigma,x)} L(\phi_s(x,\sigma),\sigma)ds + V_k^1(\phi_{\tau(\sigma,x)}(x,\sigma))\right\}.$$

□

We would like to show that V_k^1 is uniformly bounded and locally uniformly continuous. Considering uniform continuity of V_k^1, let C_k be as in (9) and γ_n^σ the transversal foliation of $\dot{x} = f(x,\sigma)$. For each $\sigma \in \Sigma_k$ we define the regions in $\mathbb{R}^n$

$$\begin{aligned} M_c^\sigma &:= \{\ x \in (\gamma_n^\sigma)^{-1}(c) \mid c \in C_k\ \} \\ M_{c-}^\sigma &:= \{\ x \in (\gamma_n^\sigma)^{-1}((-1,c)) \mid c \in C_k\ \}. \end{aligned}$$

Remark 6.1.

(a) Let $x \in \mathcal{R}_k^1$ and $\mu = \sigma_1\sigma_2\ldots \in \mathcal{U}_k^1$. Suppose that $x_{j-1} \in M_c^{\sigma_j}$ for some $c \in C_k$ so that $\tau_j = 0$ and $\sigma_{j+1} \neq \sigma_j$. Let $\tilde{\mu} = \ldots\sigma_{j-1}\sigma_{j+1}\ldots$. Then $J(x,\mu) = J(x,\tilde{\mu})$. Therefore, whenever we construct an ϵ-optimal control for x we may assume that if $\tau_j = 0$ then $\sigma_{j+1} = \sigma_j$.

(b) If $x, y \in M^{\sigma}_{c-}$ for some $c \in C_k$ and $\tau(\sigma, x)$ and $\tau(\sigma, y)$ are defined using (tb) then $|\tau(\sigma, x) - \tau(\sigma, y)| \to 0$ and $\|\phi_{\tau(\sigma,x)}(x, \sigma) - \phi_{\tau(\sigma,y)}(y, \sigma)\| \to 0$ as $\|x - y\| \to 0$ in M^{σ}_{c-}, since M^{σ}_{c} is a smooth submanifold. For the details, see Theorem 6.1, p. 91-94, [10]. If instead $\tau(\sigma, x)$ and $\tau(\sigma, y)$ are defined using (ta) and σ is an ϵ-optimal control for x, then by Assumption 2.2 the same results hold.

(c) For each $x \in \cup_k \mathcal{R}^1_k$ and $\epsilon > 0$ there exists $m \in \mathbb{Z}^+$ and $\mu \in \mathcal{U}^1_m$ such that μ is an ϵ-optimal control for x w.r.t. V^1 satisfying Assumptions 2.2. This follows from Assumptions 2.2, $V^1_k(x) \geq V(x)$, and the fact that we can well-approximate an ϵ-optimal control for V by a control in $\mathcal{U}^1_m$, for large enough m.

Lemma 2. *For each $y \in \cup_k \mathcal{R}^1_k$ and $\epsilon > 0$, there exists $m_\epsilon \in \mathbb{Z}^+$ and $\eta_\epsilon > 0$ such that*

$$|V^1_k(x) - V^1_k(y)| < 2\epsilon$$

for all $|x - y| < \eta_\epsilon$ and $k > m_\epsilon$.

Proof. Fix $y \in \cup_k \mathcal{R}^1_k$. By Remark 6.1(c) there exists $m_1 > 0$ and $\mu \in \mathcal{U}^1_{m_1}$ such that μ is an ϵ-optimal control for y satisfying Assumptions 2.2. Let $x \in \mathcal{R}^1_{m_1}$. Then $V^1_k(x) - V^1_k(y) \leq J^1_k(x, \mu_x) - J^1_k(y, \mu) + \epsilon$ for any $\mu_x \in \mathcal{U}^1_{m_1}$ and $k > m_1$. If we can show that for fixed y and μ there exists $\mu_x \in \mathcal{U}^1_{m_1}$ such that

$$J^1_k(x, \mu_x) - J^1_k(y, \mu) < \epsilon \tag{21}$$

for all $x \in \mathcal{R}^1_{m_1}$ sufficiently close to y, then $V^1_k(x) - V^1_k(y) \leq 2\epsilon$ for all $k \geq m_1$.

Conversely, by Remark 6.1(c) there exists $m_2 > 0$ and $\mu_x \in \mathcal{U}^1_{m_2}$ such that μ_x is an ϵ-optimal control for x satisfying Assumptions 2.2. Then $V^1_k(y) - V^1_k(x) \leq J^1_k(y, \mu) - J^1_k(x, \mu_x) + \epsilon$ for any $\mu \in \mathcal{U}^1_{m_2}$ and $k > m_2$. If we can show that for fixed y there exists $\mu \in \mathcal{U}^1_{m_2}$ such that

$$J^1_k(y, \mu) - J^1_k(x, \mu_x) < \epsilon \tag{22}$$

for all $x \in \mathcal{R}^1_{m_2}$ sufficiently close to y, then $V^1_k(x) - V^1_k(y) \geq -2\epsilon$ for all $k \geq m_2$. The result follows by letting $m_\epsilon = \min\{m_1, m_2\}$. Thus, we must show (21) and (22).

Consider first (21). Let $\overline{\mu} = \overline{\sigma}_1 \overline{\sigma}_2 \ldots \in \mathcal{U}^1_k$ be an ϵ-optimal control for y such that $y_N \in \partial \Omega_f$ and Remark 6.1(a) holds. By redefining indices, we can associate with $\overline{\mu}$ the open-loop control $\tilde{\mu} = (\sigma_1, \tilde{\tau}_1)(\sigma_2, \tilde{\tau}_2) \ldots$, where τ_i is the time σ_i is applied. We claim there exists $\tilde{\mu}^x = (\sigma_1, \tilde{\tau}^x_1)(\sigma_2, \tilde{\tau}^x_2) \ldots$ such that as $x \to y$, (a)

$x_j \to y_j$, (b) $\tilde{\tau}_j^x \to \tilde{\tau}_j$, and (c) $x_N \in \partial\Omega_f$. Then we have

$$J_k^1(x,\tilde{\mu}^x) - J_k^1(y,\tilde{\mu}) \le \sum_{j=1}^{N} \int_0^{\tilde{\tau}_j} |L(\phi_s(x_{j-1},\sigma_j),\sigma_j) - L(\phi_s(y_{j-1},\sigma_j),\sigma_j)|ds$$
$$+\sum_{j=1}^{N} \Big| \int_{\tilde{\tau}_j}^{\tilde{\tau}_j^x} L(\phi_s(x_{j-1},\sigma_j),\sigma_j)ds \Big| + |h(y_N) - h(x_N)|$$
$$\le L_L T_K \exp(L_f T_K) \sum_{j=1}^{N} \|x_{j-1} - y_{j-1}\|$$
$$+M_L \sum_{j=1}^{N} |\tilde{\tau}_j^x - \tilde{\tau}_j| + L_h|x_N - y_N|.$$

By the claim the r.h.s. can be made less than ϵ. Thus, we need only show there exists $\tilde{\mu}^x = (\sigma_1,\tilde{\tau}_1^x)(\sigma_2,\tilde{\tau}_2^x)\ldots$ which satisfies the claim and $\mu^x \in \mathcal{U}_k^1$ can be reconstructed from it, based on the discrete states in Q visited by $\phi_t(x,\tilde{\mu}^x)$. We argue by induction. Suppose (a)-(c) hold at $j-1$. We show they hold at j. By Remark 6.1(a) we need only consider the case when $y_{j-1} \in M_{c-}^{\sigma_j}$ and $y_j \in M_c^{\sigma_j}$ for some $c \in C_k$. For x_{j-1} sufficiently close to y_{j-1} $x_{j-1} \in M_{c-}^{\sigma_j}$. By Remark 6.1(b) there exists $\tilde{\tau}_j^x$ such that $x_j = \phi_{\tilde{\tau}_j^x}(x_{j-1},\sigma_j) \in M_c^{\sigma}$ and $\tilde{\tau}_j^x \to \tilde{\tau}_j$ and $x_j \to y_j$ as $x_{j-1} \to y_{j-1}$. The case $y_{j-1} \in M_{c-}^{\sigma_j}$ and $y_j \in \partial\Omega_f$ follows in the same way from Remark 6.1(a) and Assumption 2.2. Proving (22) follows along the same lines as the proof for (21). □

To show boundedness of V_k^1, let

$$T(x) := \inf_{\mu\in\mathcal{U}_k^1} T(x,\mu).$$

In light of Assumption 2.1(2), we have that for all $x \in \mathbb{R}^n$, $|V_k^1(x)| \le T(x) \cdot M_L + M_h$. Consider the set

$$K_a := \{x \in \mathcal{R}_k^1 \mid T(x) < a\}.$$

Then $|V_k^1(x)| \le a \cdot M_L + M_h$, $\forall x \in K_a$.

We have shown that on each $K_a \subseteq \mathbb{R}^n$, $\{V_k^1\}$ forms a family of equibounded, locally equicontinuous functions. It follows by Arzela-Ascoli Theorem that along some subsequence k_n $V_{k_n}^1$ converges to a continuous function V_*.

Proposition 3. *V_* is the unique viscosity solution of HJB.*

Proof. We show that V_* solves HJB in the viscosity sense. Let $\psi \in C^1(\mathbb{R}^n)$ and suppose $x_0 \in \Omega$ is a strict local maximum for $V_* - \psi$. There exists a closed ball B centered at x_0 such that $(V_* - \psi)(x_0) > (V_* - \psi)(x)$, for all $x \in B$. Let $x_{0\delta_k}$ be a maximum point for $V_k^1 - \psi$ over B. Since $V_k^1 \to V_*$ locally uniformly it follows that $x_{0\delta_k} \to x_0$ as $\delta_k \to 0$. Then, for any $\sigma \in \Sigma_k$, the point $\phi_\tau(x_{0\delta_k},\sigma)$ is in B

(using boundedness of f), for sufficiently small δ_k and $0 \le \tau \le \tau(x_{0\delta_k}, \sigma)$, since $\tau(x_{0\delta_k}, \sigma) \to 0$ as $\delta_k \to 0$. Therefore,

$$V_k^1(x_{0\delta_k}) - \psi(x_{0\delta_k}) \ge V_k^1(\phi_\tau(x_{0\delta_k}, \sigma)) - \psi(\phi_\tau(x_{0\delta_k}, \sigma)).$$

Considering Equation 19, we have

$$\begin{aligned} 0 &= -\min_{\sigma \in \Sigma_k} \left\{ V_k^1(\phi_\tau(x_{0\delta_k}, \sigma)) - V_k^1(x_{0\delta_k}) + \int_0^\tau L(\phi_s(x_{0\delta_k}, \sigma), \sigma) ds \right\} \\ &\ge -\min_{\sigma \in \Sigma_k} \left\{ \psi(\phi_\tau(x_{0\delta_k}, \sigma)) - \psi(x_{0\delta_k}) + \int_0^\tau L(\phi_s(x_{0\delta_k}, \sigma), \sigma) ds \right\}. \end{aligned}$$

Since $\psi \in C^1(\mathbb{R}^n)$, we have by the Mean Value Theorem,

$$0 \ge -\min_{\sigma \in \Sigma_k} \left\{ \frac{\partial \psi}{\partial x}(y) \cdot \int_0^\tau f(\phi_s(x_{0\delta_k}, \sigma), \sigma) ds + \int_0^\tau L(\phi_s(x_{0\delta_k}, \sigma), \sigma) ds \right\}$$

where $y = \alpha x_{0\delta_k} + (1-\alpha)\phi_\tau(x_{0\delta_k}, \sigma)$ for some $\alpha \in [0,1]$. Dividing by $\tau > 0$ on each side and taking the limit as $\delta_k \to 0$, we have $V_k^1 \to V_*$, $x_{0\delta_k} \to x_0$, $\tau \to 0$, and $y \to x_{0\delta_k}$. By the Fundamental Theorem of Calculus, the continuity of f and L, and the uniform continuity in u of the expression in brackets, we obtain

$$0 \ge -\inf_{u \in U} \left\{ \frac{\partial \psi}{\partial x}(x_0) \cdot f(x_0, u) + L(x_0, u) \right\}.$$

This confirms part (i) of the viscosity solution definition. Part (ii) is proved in an analogous manner. □

Step 2: approximate cost functions.

In this step we keep the semantics on piecewise constant controls of Step 1 but replace cost functions L and h by approximations L^2 and $\hat{h}$. We define an approximate instantaneous cost $L^2 : \Omega \times \Sigma_k \to \mathbb{R}$ given by

$$L^2(x, \sigma) := \hat{L}(q) \tag{23}$$

where $(\sigma, x) \in q$. For $x \in \Omega$ and $\mu = \sigma_1\sigma_2 \ldots \in \mathcal{U}_k^1$, if $T(x, \mu) < \infty$, the cost-to-go function $J_k^2 : \Omega \times \mathcal{U}_k^1 \to \mathbb{R}$ is

$$J_k^2(x, \mu) = \sum_{j=1}^{N} L^2(x_{j-1}, \sigma_j) + \hat{h}(x_N)$$

where $N = \min\{j \ge 0 \mid x_j \in \partial\Omega_f\}$.

We define a value function $V_k^2 : \mathbb{R}^n \to \mathbb{R}$ as follows. For $x \in \Omega \setminus \Omega_f$,

$$V_k^2(x) = \inf_{\mu \in \mathcal{U}_k^1} J_k^2(x, \mu) \tag{24}$$

and for $x \in \Omega_f$, $V_k^2(x) = \hat{h}(x)$. For $x \in \Omega$ such that $V_k^2(x) < \infty$, V_k^2 satisfies the DPP

$$V_k^2(x) = \min_{\sigma \in \Sigma_k} \{ L^2(x, \sigma) + V_k^2(\phi_{\tau(\sigma,x)}(x, \sigma)) \}.$$

The proof is along the same lines as that of Proposition 1.

The following facts are useful for the subsequent result.

Fact 1. *If $\delta_k < \frac{m_f}{L_f}$, then for all $q \in Q$,*

$$\tau_q \leq \frac{\delta_k}{m_f - L_f \delta_k}. \tag{25}$$

Proof. Let $q \in Q$. Fix $x \in \Omega$ and $\sigma \in \Sigma_k$ such that $(\sigma, x) \in q$ and $\|\phi_{\tau_q}(x,\sigma) - x\| \leq \delta_k$. We have

$$\begin{aligned}
\delta_k \geq \|\phi_{\tau_q} - x\| &= \left\| \int_0^{\tau_q} f(\phi_s(x,\sigma),\sigma) ds \right\| \\
&\geq \left\| \int_0^{\tau_q} f(x,\sigma) ds \right\| - \left\| \int_0^{\tau_q} [f(\phi_s(x,\sigma),\sigma) - f(x,\sigma)] ds \right\| \\
&\geq \tau_q \|f(x,\sigma)\| - \tau_q L_f \delta_k.
\end{aligned}$$

Therefore,

$$\tau_q \leq \frac{\delta_k}{\|f(x,\sigma)\| - L_f \delta_k}.$$

Using Assumption 4.1(2) the result follows. □

Fact 2. *Let $x, x' \in M_c^\sigma$ for some $c \in C_k$ and $\sigma \in \Sigma_k$ such that $\|x - x'\| \leq \delta_k$. Let τ, τ' be times such that $\phi_\tau(x,\sigma), \phi_{\tau'}(x',\sigma) \in M_{c+\Delta}^\sigma$. Then $|\tau - \tau'| \leq c_\gamma \tau \delta_k$ for some $c_\gamma > 0$.*

Proof. We have

$$\int_0^\tau \frac{d}{ds}(\gamma_n^\sigma(\phi_s(x,\sigma)))ds = \int_0^{\tau'} \frac{d}{ds}(\gamma_n^\sigma(\phi_s(x',\sigma)))ds.$$

Let $f = f(\phi_s(x,\sigma),\sigma)$, $f' = f(\phi_s(x',\sigma),\sigma)$, $d\gamma = d\gamma_n^\sigma(\phi_s(x,\sigma))$ and $d\gamma' = d\gamma_n^\sigma(\phi_s(x',\sigma))$. Then rearranging terms

$$\int_0^\tau (f' \cdot d\gamma')ds - \int_0^\tau (f \cdot d\gamma)ds = \int_{\tau'}^\tau (f' \cdot d\gamma')ds.$$

Let L_1 be the Lipschitz constant of $f \cdot d\gamma$ (using the fact that γ_n^σ is smooth). Then

$$\int_{\tau'}^\tau f' \cdot d\gamma' \leq L_1 \tau \|x - x'\| \leq L_1 \tau \delta_k.$$

Since γ_n^σ defines a transversal foliation to vector field $f(\cdot,\sigma)$, $f \cdot d\gamma > 0$. Let $c = \min_{s \in [\tau,\tau']}\{f' \cdot d\gamma'\} > 0$. Letting $c_\gamma = \frac{L_1}{c}$ we obtain the result. □

Proposition 4. *Let $k_0 \in \mathbb{Z}^+$ be arbitrary, $x \in \mathcal{R}_{k_0}^1$, and $\mu \in \mathcal{U}_{k_0}^1$ be an ϵ-optimal control for x. Then $|J_k^1(x,\mu) - J_k^2(x,\mu)| \to 0$ as $k \to \infty$.*

Proof. We have

$$\left|J_k^1(x,\mu) - J_k^2(x,\mu)\right| \le \left|\sum_{j=1}^{N}\Big[\int_0^{\tau(\sigma_j,x_{j-1})} L(\phi_s(x_{j-1},\sigma_j),\sigma_j)ds\Big] + h(x_N) - \sum_{j=1}^{N}\big[\tau_{q_{j-1}}L(\xi_{j-1},\sigma_j)\big] - \hat{h}(x_N)\right|$$

where $(x_{j-1},\sigma_j) \in q_{j-1}$ and $q_{j-1} = [(\xi_{j-1},\sigma_j)]$. There exists ξ_N such that $\hat{h}(x_N) = h(\xi_N)$ and $\|x_N - \xi_N\| \le \delta_k$. Also, using the Mean Value Theorem, there exists $\tilde{t}$ with $\tilde{x} = \phi_{\tilde{t}}(x_{j-1},\sigma_j)$ and $\|\tilde{x} - \xi_{j-1}\| \le \delta_k$ such that

$$\begin{aligned}\left|J_k^1(x,\mu) - J_k^2(x,\mu)\right| &\le \sum_{j=1}^{N}\left|\tau(\sigma_j,x_{j-1})L(\tilde{x},\sigma_j) - \tau_{q_{j-1}}L(\xi_{j-1},\sigma_j)\right| + \left|h(x_N) - \hat{h}(x_N)\right| \\ &\le \sum_{j=1}^{N}\tau_{q_{j-1}}L_L\delta_k + \sum_{j=1}^{N}[\tau_{q_{j-1}} - \tau(\sigma_j,x_{j-1})]L(\tilde{x},\sigma_j) + L_h\delta_k.\end{aligned}$$

Using Fact 1 the first term on the r.h.s. decreases linearly as δ_k. Call the second term on the r.h.s. "B". Splitting B into sums over control switches and time steps, we have

$$\begin{aligned}B &\le M_L\sum_{j=2}^{N}[\tau_{q_{j-1}} - \tau(\sigma_j,x_{j-1})]\mathbf{1}(\sigma_j = \sigma_{j-1}) + M_L\sum_{j=1}^{N}[\tau_{q_{j-1}} - \tau(\sigma_j,x_{j-1})]\mathbf{1}(\sigma_j \ne \sigma_{j-1}) \\ &\le M_L\sum_{j=2}^{N}c_{j-1}\tau_{q_{j-1}}\delta_k + M_L\sum_{j=1}^{N}\tau_{q_{j-1}}\mathbf{1}(\sigma_j \ne \sigma_{j-1})\end{aligned}$$

for some $c_{j-1} \in \mathbb{R}$. In the second line we used Fact 2 and the fact that $\tau_{q_{j-1}} \ge \tau(\sigma_j,x_{j-1})$. Using Fact 1 the first term on the r.h.s. decreases linearly as δ_k. The second term on the r.h.s. goes to zero since μ has a fixed number of control switches for all $k \ge k_0$. □

Step 3: discrete states and non-determinacy.

We define

$$\hat{V}_k(x) := \min_{\sigma\in\Sigma_k}\{\ \hat{V}_k(q) \mid (\sigma,x) \in q\ \}.$$

Also let $\hat{\mathcal{R}}_k = \{x \in \Omega \mid \hat{V}_k(x) < \infty\}$ and $\hat{\mathcal{R}} = \cup_k\hat{\mathcal{R}}_k$.

Remark 6.2.

(a) By Remark 6.1(c) and $V_k^1(x) \le V_k^2(x)$, for each $x \in \cup_k\mathcal{R}_k^1$ and $\epsilon > 0$ there exists $m_\epsilon \in \mathbb{Z}^+$ and $\mu \in \mathcal{U}_{m_\epsilon}^1$ such that μ is an ϵ-optimal control for x w.r.t. V^2 satisfying Assumptions 2.2.

(b) $\hat{\mathcal{R}} \subset \cup_k\mathcal{R}_k^1$, but the converse is not true, in general.

(c) If μ is an ϵ-optimal control for x w.r.t. V_k^2, then we can assume $\phi_t(x,\mu)$ does not self-intersect, for if it did we can find $\tilde{\mu}$, also ϵ-optimal, which eliminates loops in $\phi_t(x,\mu)$.

(d) $\|x-y\| \to 0$ as $k \to \infty$ for all $y \in r_e(x)$ and all edges e of H_k, the hybrid automaton defined using Σ_k and C_k given in (9).

Proposition 5. *For all* $x \in \hat{\mathcal{R}}$, $|\hat{V}_k(x) - V_k^2(x)| \to 0$ *as* $k \to \infty$.

Proof. Fix $\epsilon > 0$ and $x \in \hat{\mathcal{R}}$. By Remark 6.2(a) there exists $m_\epsilon > 0$ and an ϵ-optimal control $\mu \in \mathcal{U}^1_{m_\epsilon}$ for x w.r.t. $V^2_{m_\epsilon}$. Denote $\mu = ((\sigma_1,\tau_1),\dots,(\sigma_N,\tau_N))$, where τ_i is the time σ_i is applied. If c is a policy derived using δ_k and C_k, for $k \geq m_\epsilon$, then $0 \leq \hat{V}_k(q) - V_k^2(x) \leq \hat{J}_k(q,c) - J_k^2(x,\mu) + \epsilon$, where $q = [(\sigma_1,x)]$. If we can show there exists $\overline{k} \geq m_\epsilon$ such that for $k > \overline{k}$, there exists a policy $\overline{c}$ such that $\hat{J}_k(q,\overline{c}) - J_k^2(x,\mu) < \epsilon$ then the result follows.

We can find $\overline{k} \geq m_\epsilon$ such that, by Remark 6.2(d) and the transversality of $\phi_t(x,\mu)$ with the submanifolds where it switches controls and with Ω_f, there exists $\tilde{\mu} \in \mathcal{U}_k$, $k > \overline{k}$, such that each trajectory $\phi_t(x,\tilde{\mu})$ of H_k switches controls on the same (transversal) submanifolds as $\phi(x,\mu)$ and reaches Ω_f. Let Ψ_k be this set of trajectories of H_k starting at x. Let $W_k(\phi) = \sum_{j=1}^N L^2(x_{j-1},\sigma_j) + \hat{h}(x_N)$ where $\tilde{\mu} = ((\sigma_1,\tilde{\tau}_1,\dots,(\sigma_N,\tilde{\tau}_N))$, $x_j^- = \phi_{\tilde{\tau}_j}(x_{j-1},\sigma_j)$, and $x_j \in r_e(x_j^-)$, where $e = (\sigma_j,\sigma_{j+1})$ is an edge of H_k.

We observe that for $\phi,\phi' \in \Psi_k$, $\tilde{\mu} \in \mathcal{U}_k^1$, $k > \overline{k}$, $|W_k(\phi) - W_k(\phi')| \to 0$ as $k \to \infty$, using Lipschitz continuity of L and h, Remark 6.2(d), and the fact that $\tilde{\mu}$ is fixed for $k > \overline{k}$. Notice that $\phi(x,\mu) \in \Psi_k$, $k > \overline{k}$. We can define the control policy $\overline{c}$ such that automaton A accepts the time abstract trajectory starting at q corresponding to each trajectory of Ψ_k and with all other control assignments of $\overline{c}$ as time steps. $\overline{c}$ is admissible because otherwise some $\phi' \in \Psi_k$ would have a Zeno loop. Since ϕ' approaches $\phi_t(x,\mu)$ as $k \to \infty$, this would imply $\phi_t(x,\mu)$ has a loop, contradicting Remark 6.2(c). Now we observe that $\hat{J}(q,\overline{c}) = \max_{\phi\in\Psi_k} W_k(\phi) := W_k(\overline{\phi})$. Thus, $\hat{J}_k(q,\overline{c}) - J_k^2(x,\mu) \leq |W_k(\overline{\phi}) - W_k(\phi(x,\mu))| \to 0$ as $k \to \infty$. □

Combining Propositions 3, 4, and 5, we have

Theorem 1. *For all* $x \in \hat{\mathcal{R}}$, $\hat{V}_k(x) \to V(x)$ *as* $k \to \infty$.

7 Implementation

So far we have developed a discrete method for solving an optimal control problem based on hybrid systems and bisimulation. We showed that the solution of the discrete problem converges to the solution of the continuous problem as a discretization parameter δ tends to zero. Now we focus on the pragmatic question of how the discretized problem can be efficiently solved.

7.1 Motivation

Following the introduction of the concept of viscosity solution [16, 6], Capuzzo-Dolcetta [5] introduced a method for obtaining approximations of viscosity solutions based on time discretization of the Hamilton-Jacobi-Bellman (HJB) equation. The approximations of the value function correspond to a discrete time optimal control problem, for which an optimal control can be synthesized which is piecewise constant. Finite difference approximations were also introduced in [7] and [24]. In general, the time discretized approximation of the HJB equation is solved by finite element methods. Gonzales and Rofman [12] introduced a discrete approximation by triangulating the domain of the finite horizon problem they considered, while the admissible control set is approximated by a finite set. Gonzales and Rofman's approach is adapted in several papers, including [9].

Our work was inspired by the ideas of [26] which uses the special structure of an optimal control problem to obtain a single-pass algorithm to solve the discrete problem, thus bypassing the expensive iterations of a finite element method. The key property to find a single pass algorithm is to obtain a partition of the domain so that the cost-to-go function from any equivalence class of the partition is determined from knowledge of the cost-to-go function from those equivalence classes with strictly smaller cost-to-go functions. In our approach, we start with a triangulation of the domain provided by a bisimulation partition. *The combination of the structure of the bisimulation partition and the requirement of non-Zeno trajectories enables us reproduce the key property of [26], so that we obtain a Dijkstra-like algorithmic solution.* Our approach has the same complexity as that reported in [26] of $O(N \log N)$ if suitable data structures are used, where N is the number of locations of the finite automaton.

7.2 Non-deterministic Dijkstra algorithm

The dynamic programming solution (14)-(15) can be viewed as a shortest path problem on a non-deterministic graph subject to all optimal paths satisfying a non-Zeno condition. We consider an example to motivate the difference between the deterministic and non-deterministic cases.

Example 5. Consider the automaton of Figure 4. Suppose that $obs(e) = \sigma$ for $e = \{e_1, e_2, e_5, e_7\}$, $obs(e) = \sigma'$ for $e = \{e_3, e_4, e_6, e_8, e_9, e_{10}\}$, and $obs(e_{11}) = \sigma''$. Also, $\hat{L}(e_1) = 1$, $\hat{L}(e_4) = 4$, $\hat{L}(e_5) = 2$, $\hat{L}(e_8) = 1$, and $\hat{L}(e_{11}) = 1$, while $\hat{L}$ is zero for the other edges. If the automaton were interpreted as deterministic, one obtains

$$\begin{aligned}
\hat{V}(q_1) &= \min\{\ \hat{L}(e_1) + \hat{h}(q_f), \hat{V}(q_2), \hat{V}(q_3)\ \} \\
\hat{V}(q_2) &= \min\{\ \hat{L}(e_4) + \hat{h}(q_f), \hat{V}(q_1), \hat{V}(q_4)\ \} \\
\hat{V}(q_5) &= \min\{\ \hat{L}(e_{11}) + \hat{h}(q_f), \hat{V}(q_2), \hat{V}(q_3)\ \}.
\end{aligned}$$

These equations resolve to $\hat{V}(q_1) = \hat{V}(q_2) = \hat{V}(q_5) = 1 + \hat{h}(q_f)$, and the control policy is $c(q_1) = c(q_2) = \sigma$, and $c(q_5) = \sigma'$. If the automaton is non-deterministic,

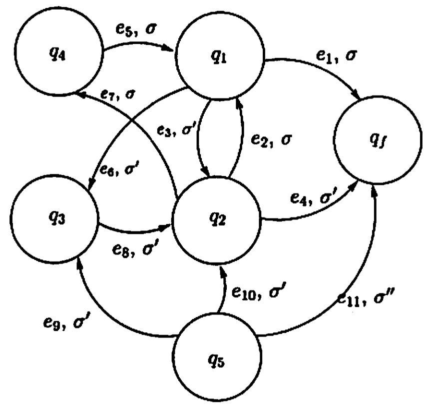

Fig. 4. Nondeterministic automaton

then the control policy is deduced as follows. Using (14) we have

$$\hat{V}(q_1) = \min\{\hat{L}(e_1) + \hat{h}(q_f), \max\{\hat{V}(q_2), \hat{V}(q_3)\}\}$$
$$\hat{V}(q_2) = \min\{\hat{L}(e_4) + \hat{h}(q_f), \max\{\hat{V}(q_4), \hat{V}(q_1)\}\}$$
$$\hat{V}(q_5) = \min\{\hat{L}(e_{11}) + \hat{h}(q_f), \max\{\hat{V}(q_3), \hat{V}(q_2)\}\}.$$

Substituting known quantities we find

$$\hat{V}(q_1) = \min\{1 + \hat{h}(q_f), 1 + \hat{V}(q_2)\}$$
$$\hat{V}(q_2) = \min\{4 + \hat{h}(q_f), 2 + \hat{V}(q_1)\}$$
$$\hat{V}(q_5) = \min\{5 + \hat{h}(q_f), 1 + \hat{V}(q_2)\}.$$

When solved simultaneously, these equations yield $\hat{V}(q_1) = 1 + \hat{h}(q_f)$, $\hat{V}(q_2) = 3 + \hat{h}(q_f)$, $\hat{V}(q_5) = 4 + \hat{h}(q_f)$, and $c(q_1) = c(q_2) = \sigma$, and $c(q_5) = \sigma'$. Notice that all trajectories are non-Zeno inspite of the fact that a trajectory starting from q_5 may take two consecutive control switches.

7.3 Description of NDD

The algorithm is a modification of the Dijkstra algorithm for deterministic graphs [8] and synthesizes an optimal, memoryless, admissible control policy that takes the states of a non-deterministic graph to a target set. As in the deterministic case, the algorithm is greedy: if a step can be taken into a set of states whose controls have already been assigned and have a minimum cost, the step is assigned.

First we define the notation. F_n is the set of states that have been assigned a control and are deemed "finished" at iteration n, while U_n are the unfinished states. At each n, $Q = U_n \cup F_n$. $\Sigma_n(q) \subseteq \Sigma_\delta$ is the set of control events at iteration n that take state q to finished states exclusively. $\tilde{U}_n$ is the set of states for which there exists a control event that can take them to finished states exclusively. $\tilde{V}_n(q)$ is a tentative cost-to-go value at iteration n. B_n is the set of "best" states among $\tilde{U}_n$.

The non-deterministic Dijkstra (NDD) algorithm first determines $\tilde{U}_n$ by checking if any q in U_n can take a step to states belonging exclusively to F_n. For states belonging to $\tilde{U}_n$, an estimate of the value function $\tilde{V}$ following the prescription of (14) is obtained: among the set of control events constituting a step into states in F_n, select the event with the lowest worst-case cost. Next, the algorithm determines B_n, the states with the lowest $\tilde{V}$ among $\tilde{U}_n$, and these are added to F_{n+1}. The iteration counter is incremented until it reaches $N = |Q|$. It is assumed in the following description that initially $\hat{V}(q) = \infty$ and $c(q) = \emptyset$ for all $q \in Q$.

```
Procedure NDD:

F_1 = Q_f; U_1 = Q − Q_f;
for each q ∈ Q_f, V̂(q) = ĥ(q);

for n = 1 to N, do
    for each q ∈ U_n,
            Σ_n(q) = {σ' ∈ Σ_δ | if q –σ'→ q', then q' ∈ F_n};
    Ũ_n = {q ∈ U_n | Σ_n(q) ≠ ∅};
    for each q ∈ Ũ_n,
            Ṽ_n(q) = min_{σ'∈Σ_n(q)} {max_{e=(q,q')∈E_σ'(q)} {L̂(e) + V̂(q')}};
    B_n = argmin_{q∈Ũ_n} {Ṽ_n(q)};
    for each q ∈ B_n,
            V̂(q) = Ṽ_n(q);
            c(q) = argmin_{σ'∈Σ_n(q)} {max_{e=(q,q')∈E_σ'(q)} {L̂(e) + V̂(q')}};
    endfor
    F_{n+1} = F_n ∪ B_n; U_{n+1} = Q − F_{n+1};
endfor
```

The algorithm is opportunistic in assigning control switches. At the first iteration, say n, that a state can take a control switch to finished states, it will be assigned the control switch by the algorithm. This is because control switches have zero instantaneous cost, so the state will have a minimum $\tilde{V}$ and will be included in B_n. In fact, B_n will include either states that can take control switches and zero cost time steps to F_n, or states that can take a non-zero cost time step to F_n. The opportunistic assignment of control switches is intuitively what we expect: waiting for a later iteration to assign them does not make sense because states that finish later have a higher or equal cost-to-go.

7.4 Justification

In this section we show that the control policy synthesized by algorithm NDD allows non-Zeno trajectories only and is optimal in the required worst-case sense.

Lemma 3. *Algorithm NDD synthesizes a control policy with no Zeno loops.*

Proof. We argue by induction. The claim is obviously true for F_1. Suppose that the states of F_n have been assigned controls forming no Zeno loops. Consider F_{n+1}. Each state of B_n takes either a time step or a control switch to F_n so there cannot be a Zeno loop in B_n. The only possibility is for some $q \in B_n$ to close a Zeno loop with states in F_n, as shown in Figure 5. This implies there exists a control assignment that allows an edge from F_n to q to be taken; but this is not allowed by NDD. Thus, F_{n+1} has no Zeno loops.

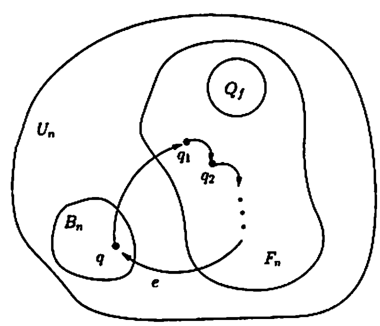

Fig. 5. A loop of control switches

Next we prove that the algorithm is *optimal*; that is, it synthesizes a control policy so that each $q \in Q$ reaches Q_f with the best worst-case cost. We observe a few properties of the algorithm. First, if all states of Q can reach Q_f then $Q - Q_f = \cup_n B_n$. Second, as in the deterministic case, the algorithm computes $\hat{V}$ in order of level sets of $\hat{V}$. In particular, $\hat{V}(B_n) \leq \hat{V}(B_{n+1})$. Finally, we need the following property.

Lemma 4. *For all $q \in Q$ and $\sigma' \in \Sigma_\delta$,*

$$\hat{V}(q) \leq \max_{e=(q,q')\in E_{\sigma'}(q)} \{\hat{L}(e) + \hat{V}(q')\}.$$

Proof. Fix $q \in Q$ and $\sigma' \in \Sigma_\delta$. There are two cases.
Case 1.

$$\hat{V}(q) \leq \max_{e=(q,q')\in E_{\sigma'}(q)} \{\hat{V}(q')\}.$$

In this case the result is obvious.
Case 2.

$$\hat{V}(q) > \max_{e=(q,q')\in E_{\sigma'}(q)} \{\hat{V}(q')\}. \tag{26}$$

We observed above that q belongs to some B_n. Suppose w.l.o.g. that $q \in B_j$. Together with (26) this implies $q' \in F_j$ for all q' such that $q \xrightarrow{\sigma'} q'$. This, in turn, means that $\sigma' \in \Sigma_j(q)$ and according to the algorithm

$$\hat{V}(q) = \tilde{V}_n(q) \leq \max_{e=(q,q')\in E_{\sigma'}(q)} \{\hat{L}(e) + \hat{V}(q')\}$$

which proves the result.

The main result of this section is the following.

Theorem 2. *Algorithm NDD is optimal.*

Proof. Let $V(q)$ be the optimal (best worst-case) cost-to-go for $q \in Q$ and $\overline{Q} = \{q \in Q \mid V(q) < \hat{V}(q)\}$. Let $l(\pi_q)$ be the number of edges taken by the shortest optimal (best worst-case) trajectory π_q from q. Define $\overline{q} = \arg\min_{q\in\overline{Q}}\{l(\pi_q)\}$. Suppose that the best worst-case trajectory starting at $\overline{q}$ is $\pi_{\overline{q}} = \overline{q} \xrightarrow{\sigma'} \overline{\overline{q}} \rightarrow \ldots$. We showed in the previous lemma that

$$\hat{V}(\overline{q}) \leq \max_{e=(\overline{q},q')\in E_{\sigma'}(\overline{q})} \{\hat{L}(e) + \hat{V}(q')\} \leq \hat{L}(e) + \hat{V}(\overline{\overline{q}}).$$

Since $\pi_{\overline{q}}$ is the best worst-case trajectory from $\overline{q}$ and by the optimality of $V(\overline{q})$

$$V(\overline{q}) = \max_{e=(\overline{q},q')\in E_{\sigma'}(\overline{q})} \{\hat{L}(e) + V(q')\} = \hat{L}(e) + \hat{V}(\overline{\overline{q}}).$$

Since $\pi_{\overline{q}}$ is the shortest best worst-case trajectory, we know that $\overline{\overline{q}} \notin \overline{Q}$, so $V(\overline{\overline{q}}) = \hat{V}(\overline{\overline{q}})$. This implies $\hat{V}(\overline{q}) \leq \hat{L}(e) + V(\overline{\overline{q}}) = V(\overline{q})$, a contradiction.

Remarks:

1. It is intuitively reasonable that the algorithm cannot synthesize a controller with Zeno loops. This worst-case behavior would show up in the value function, forcing it to be infinite for states that can reach the loop.
2. When we say that the algorithm is optimal, we mean the algorithm determines the best worst-case cost to take each state to the target set. In fact, (see remark below) the hybrid system or continuous system using the synthesized controller may perform better than worst case.
3. The non-deterministic automaton predicts more trajectories than what either the continuous system or the hybrid system can exhibit. Indeed, the automaton may exhibit a trajectory that reaches the target set using only control switches, and thus accruing zero cost. This is not of concern. Such a trajectory is an artifact of the non-determinacy of the automaton, and is not used in the determination of the value function, which accounts only for worst-case behavior, nor is it exhibited in either the hybrid system or the continuous system when the control policy synthesized by Algorithm NDD is used.

4. Related to the previous remark is that the non-deterministic automaton may also predict worst-case behavior which is not exhibited by the continuous system. It would appear that a discrepancy will develop between the cost-to-go obtained by applying the synthesized controller to the continuous system and the cost-to-go predicted by the nondeterministic automaton. This error is incurred every time a control switch is taken and is effectively an error in predicting the state and has an upper bound of δ at each iteration. This error was accounted for in our proof of convergence of the method, and the convergence result essentially depends on the fact that only a finite number of control switches occur.

Example 6. Consider the example of Figure 6. The states are labeled q_i and the number in the lower, left corner is the instantaneous cost of a time step. States can take a time step to the state immediately to the right, and they can take a control switch to states with a different σ_i value and overlapping vertically. (Edges are not drawn to keep the figure readable.) For example, state q_3 can take a time step to q_2 and a control switch to q_{12} or q_{13} using control σ_2, and to q_{17} or q_{20} using control σ_3. The algorithm generates the following data:

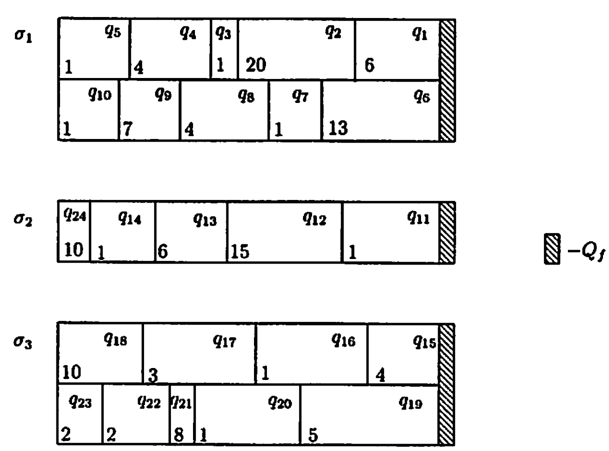

Fig. 6. Example of algorithm NDD

n	B_n	$\hat{V}(B_n)$	control
1	$\{q_{11}\}$	1	σ_2
2	$\{q_{15}, q_1\}$	1	σ_2
3	$\{q_{16}\}$	2	σ_3
4	$\{q_{17}, q_{19}\}$	5	σ_3
5	$\{q_6\}$	5	σ_3
6	$\{q_7, q_{20}\}$	6	$c(q_7) = \sigma_1$, $c(q_{20}) = \sigma_3$
7	$\{q_2, q_3, q_{12}\}$	6	σ_3
8	$\{q_4, q_8\}$	10	σ_1
9	$\{q_5\}$	11	σ_1
10	$\{q_{13}\}$	12	σ_2
11	$\{q_{21}\}$	12	σ_2
12	$\{q_{14}\}$	13	σ_2
13	$\{q_9, q_{22}\}$	13	σ_2
14	$\{q_{10}\}$	14	σ_1
15	$\{q_{24}, q_{18}, q_{23}\}$	14	σ_1

8 Implementation issues

We have implemented algorithm NDD and the transformation from the hybrid automaton to a finite automata using our bisimulation method in a prototype tool. The steps involved are:

1. Generate first integrals and a transversal foliation for each control value. We rely on the Prelle-Singer procedure to automatically generate the first integrals [20]. For all the examples we have worked with, however, we found the first integrals manually. One can also construct the transversal foliation and check the independence of the folations manually or using a symbolic algebra package such as REDUCE.
2. Enumerate all equivalence classes for each partition.
3. Determine edges of the finite automaton. This is a problem of existential quantifier elimination, as we need to determine which are the overlapping equivalence classes between the partitions of two locations.
4. Apply algorithm NDD to the resulting finite automaton.

In the conclusion we make some remarks about future developments for this tool.

9 Examples

9.1 Double integrator system

We apply our method to the time optimal control problem of a double integrator system. Given the equations of motion

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = u$$

and the set of admissible controls $U = \{u \ : \ |u| \leq 1\}$, we select $\Omega = (-1,1) \times (-1,1)$ and $\Omega_f = \overline{B}_\epsilon(0)$, the closed epsilon ball centered at 0. The cost-to-go function is $J(x,\mu) = \int_0^{T(x,\mu)} dt$. The bang-bang solution obtained using Pontryagin's maximum principle is well known to involve a single switching curve shown in Figure 7. In region R_1, the control $u = 1$ is applied. When the switching curve is reached, the control is switched to $u = -1$. In region R_{-1}, the control $u = -1$ is applied, and when the switching curve is reach, the control is switched to $u = 1$. The continuous value function V is shown in Figure 8.

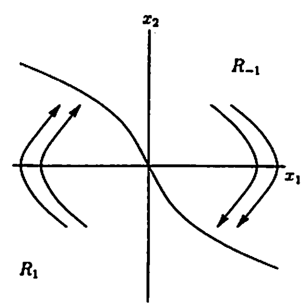

Fig. 7. The switching curve for the double integrator system.

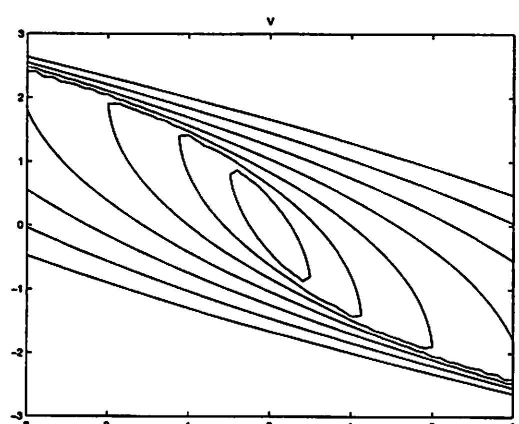

Fig. 8. Value function for the continuous problem.

The results of algorithm NDD are shown in Figures 9, 10, and 11. Figure 9 shows $\hat{V}$ for $\Delta = 0.1$. The enabling conditions $g_{e_{-1}}$ and g_{e_1} are shown in Figures 10 and 11, respectively. The roughness in the boundaries of the enabling conditions is caused both by the discretization of the state space and by the non-determinism of the finite automaton.

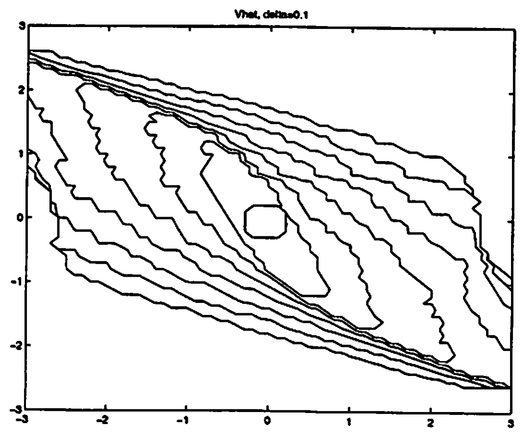

Fig. 9. $\hat{V}$ for $\Delta = 0.1$.

9.2 Fuller's problem

In this example we discuss how our method can be applied in the canonically difficult situation of Fuller's problem. Fuller's problem is of interest because all of its trajectories are Zeno. We propose an ad hoc method to avoid the Zeno behavior.

Consider the optimal control problem (2) with $|u| \leq 1$ and the cost function $J(x, \mu) = \int_0^{T(x,\mu)} x_1^2(s)ds$. Let $\xi = .4446$, the unique positive root of $x^4 + \frac{x^2}{12} - \frac{1}{18} = 0$. It was shown in [11] that the optimal switching curves Γ_- and Γ_+ are given by

$$\begin{aligned} \Gamma_+ : \quad & x_i = -\xi x_2^2 \;\; x_2 > 0 \\ \Gamma_- : \quad & x_i = \xi x_2^2 \;\; x_2 < 0. \end{aligned}$$

The situation is the same as in Figure 7. The upper vector field X_- uses $u = -1$ while the lower vector field X_+ uses $u = 1$. The combined vector field is denoted X.

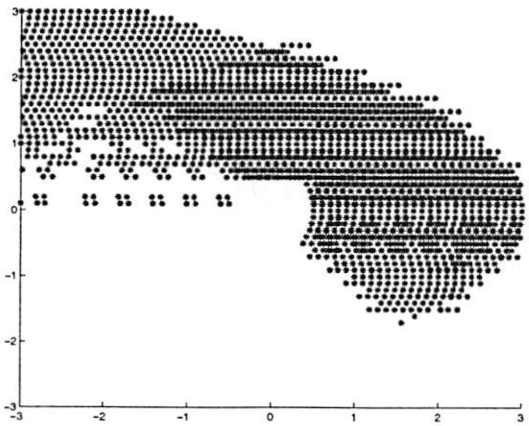

Fig. 10. Enabling condition $g_{e_{-1}}$.

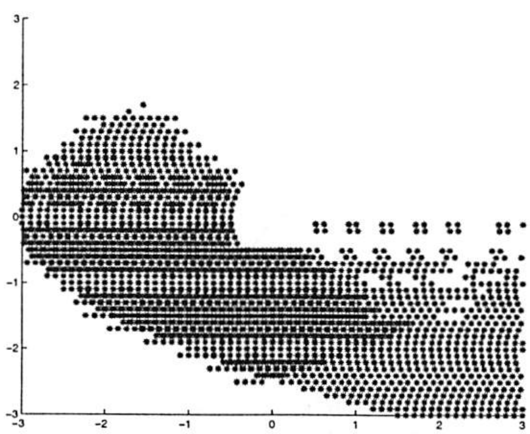

Fig. 11. Enabling condition g_{e_1}.

Solutions of X_- are parabolic curves $x_1 - \frac{1}{2}x_2^2 = c$, $c \in \mathbb{R}$. The parabola meets Γ_+ at $x_1 = -\xi x_2^2 = c + \frac{1}{2}x_2^2$ or

$$\left(\frac{-\xi c}{\frac{1}{2}-\xi}, \sqrt{\frac{c}{\frac{1}{2}-\xi}}\right). \tag{27}$$

The parabola meets Γ_- when $x_1 = \xi x_2^2 = c - \frac{1}{2}x_2^2$ or

$$\left(\frac{\xi c}{\frac{1}{2}+\xi}, -\sqrt{\frac{c}{\frac{1}{2}+\xi}}\right). \tag{28}$$

Let the X trajectory cross Γ_+ at $p^n = (x_1^n, x_2^n)$ and Γ_- at $\overline{p}^n = (\overline{x}_1^n, \overline{x}_2^n)$. (27) and (28) given

$$\overline{x}_2^n = \sqrt{\frac{\frac{1}{2}-\xi}{\frac{1}{2}+\xi}}x_2^n. \tag{29}$$

Since the picture is symmetric with respect to x_1 and $-x_1$ and by (29)

$$x_2^{n+1} = \frac{\frac{1}{2}-\xi}{\frac{1}{2}+\xi}x_2^n.$$

Hence $x_1^n, x_2^n \to 0$ as $n \to \infty$. The time it takes the X trajectory to go from p^n to $\overline{p}^n$ to p^{n+1} is

$$\begin{aligned} t^n &= x_2^n + 2|\overline{x}_2^n| + x_2^{n+1} \\ &= \frac{1 + 2\sqrt{\frac{1}{4}-\xi^2}}{\frac{1}{2}+\xi}x_2^n. \end{aligned}$$

The total time elapsed for an X trajectory to reach the origin is

$$\begin{aligned} T &= \sum_{n=1}^{\infty} t^n \\ &= \frac{1 + 2\sqrt{\frac{1}{4}-\xi^2}}{\frac{1}{2}+\xi} \sum_{n=0}^{\infty} \left(\frac{\frac{1}{2}-\xi}{\frac{1}{2}+\xi}\right)^n x_{20} \\ &= \frac{1 + 2\sqrt{\frac{1}{4}-\xi^2}}{2\xi} x_{20}. \end{aligned}$$

Thus, every trajectory takes an infinite number of switches in finite time. The origin is called a *Zeno point.* It was shown in [23] that Zeno points are stationary points of the hybrid system but not equilibrium points of either vector field X_- or X_+. The continuous value function is shown in Figure 12.

Because our method cannot be proved to converge to the continuous value function in the presence of Zeno behavior, we propose the ad hoc fix of enlarging the target set to be a closed ball around the origin. The results of algorithm NDD are shown in Figure 13 for $\Delta = 0.1$.

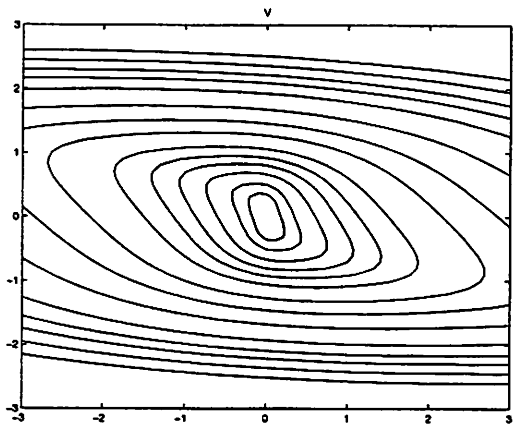

Fig. 12. Value function for the continuous problem.

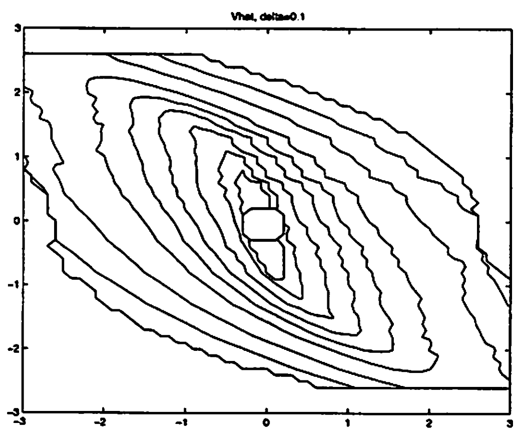

Fig. 13. $\hat{V}$ for $\Delta = 0.1$.

9.3 Nonlinear system

Consider the optimal control problem

$$\begin{aligned} \dot{x}_1 &= u_1 x_2 \\ \dot{x}_2 &= u_2 \end{aligned} \tag{30}$$

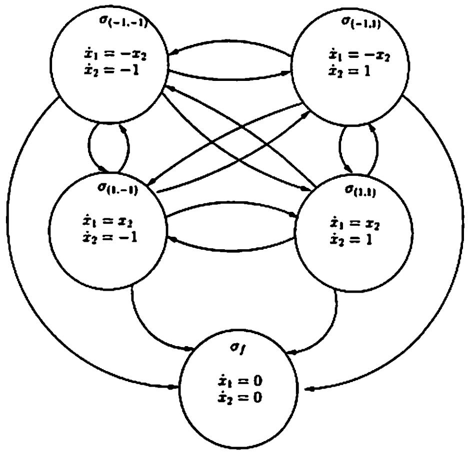

Fig. 14. Hybrid automaton for nonlinear example

The control u takes four values: $(-1,-1)$, $(-1,1)$, $(1,-1)$, $(1,1)$. The cost is $J = \int dt$. Also $u_1 \in [-1,1]$ and $u_2 \in [-1,1]$.

The first integrals for $u = (-1,-1)$ and $u = (1,1)$ are $x_1 - \frac{1}{2}x_2^2 = c_1$ and $x_2 = c_2$, and for $u = (-1,1)$ and $u = (1,-1)$, $x_1 + \frac{1}{2}x_2^2 = c_1$ and $x_2 = c_2$.

10 Conclusion

In this paper we have developed a methodology for the synthesis of optimal controls based on hybrid systems and bisimulations. The idea is to translate an optimal control problem to a switching problem on a hybrid system whose locations describe the dynamics when the control is constant. When the vector fields for each location of the hybrid automaton have local first integrals which can be expressed analytically we are able to define a finite bisimulation using the approach of [4]. From the finite bisimulation we obtain a (time abstract) finite automaton upon which a dynamic programming problem can be formulated that can be solved efficiently.

We developed an efficient single-pass algorithm to solve a dynamic programming problem on a non-deterministic graph which arose in the solution of a continuous optimal control problem using hybrid systems and bisimulation. The efficacy of the method was demonstrated on two examples. In particular, the second example showed that the canonically difficult situation of Fuller's example can be handled by our method.

The paper suggests some areas for future research. Foremost we formulated a hybrid optimal control problem used as a conceptual step in the translation to the dynamic programming problem. This problem is of interest in its own right, and further work must be done to characterize its solution. On the implementation side, we mentioned the need for efficient existential quantifier elimination to go from hybrid automata to finite automata. Also, heuristics can be introduced in our prototype tool to improve its performance. These heuristics can include building the finite automaton on the fly based on a course reachability analysis while algorithm NDD is executing.

References

1. M. Bardi and I. Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations.* Birkhäuser, Boston, 1997.
2. V.G. Boltyanskii. Sufficient conditions for optimality and the justification of the dynamic programming method. *SIAM Journal of Control*, 4, pp. 326-361, 1966.
3. M. Branicky, V. Borkar, S. Mitter. A unified framework for hybrid control: model and optimal control theory. *IEEE Trans. AC*, vol. 43, no. 1, pp. 31-45, January, 1998.
4. M. Broucke. A geometric approach to bisimulation and verification of hybrid systems. In *Hybrid Systems: Computation and Control*, F. Vaandrager and J. van Schuppen, eds., LNCS 1569, p. 61-75, Springer-Verlag, 1999.
5. I. Capuzzo Dolcetta and L.C. Evans. Optimal switching for ordinary differential equations. *SIAM J. Control and Optimization*, vol. 22, no. 1, pp. 143-161, January 1984.
6. M Crandall, P. Lions. Viscosity solutions of Hamilton-Jacobi equations. *Trans. Amer. Math. Soc.*, vol. 277, no. 1, pp. 1-42, 1983.
7. M.G. Crandall, P.L. Lions. Two approximations of solutions of Hamilton-Jacobi equations. *Mathematics of Computation*, vol. 43, no. 176, pp. 1-19, July 1984.
8. E.W. Dijkstra. A note on two problems in connection with graphs. *Numerische Mathematik* 1, p. 269-271, 1959.
9. M. Falcone. A numerical approach to the infinite horizon problem of deterministic control theory. *Applied Mathematics and Optimization*, 15, pp. 1-13, 1987.
10. W.H. Fleming, R.W. Rishel. *Deterministic and stochastic optimal control.* Springer-Verlag, New York, 1975.
11. A. T. Fuller. Study of an optimum non-linear control system. *J. Electronics Control*, 15, pp.63-71, 1963.
12. R. Gonzales and E. Rofman. On deterministic control problems: an approximation procedure for the optimal cost. I: the stationary problem. *SIAM J. Contr. Optim.*, vol. 23, no. 2, pp. 242-266, 1985.
13. T. Henzinger. Hybrid automata with finite bisimulations. In *"Proc. 22nd ICALP: Automata, Languages and Programming*, LNCS 944, pp. 324-335, Springer-Verlag, 1995.

14. I. Kupka. The ubiquity of the Fuller phenomenon. In *Nonlinear controllability and optimal control*, H.J. Sussmann, ed. Dekker, 1990.
15. X. Yang, M. Lemmon, and P. Antsaklis. On the supremal controllable sublanguage in the discete-event model of nondeterministic hybrid control systems. em IEEE Trans. AC, vol. 40, no. 12, pp. 2098-2103, December 1995.
16. P.L. Lions. *Generalized solutions of Hamilton-Jacobi equations.* Pitman, Boston, 1982.
17. O. Maler, A Pnueli, J. Sifakis. On the synthesis of discrete controllers for timed systems. In *Proc. STACS '95*, E.W. Mayr and C. Puech, eds. LNCS 900, Springer-Verlag, p. 229-242, 1995.
18. L. Polymenakos, D. Bertsekas, and J. Tsitsiklis. Implementation of efficient algorithms for globally optimal trajectories. *IEEE Trans. AC*, vol.43, no.2, pp. 278-83, Feb. 1998.
19. L. S. Pontryagin, et. al. *The mathematical theory of optimal processes.* Interscience Publishers, 1962.
20. M.J. Prelle and M.F. Singer. Elementary first integrals of differential equations. *Transactions of the American Mathematical Society*, vol. 279, no. 1, September 1983, pp. 215-229.
21. J. Raisch. Controllability and observability of simple hybrid control systems-FDLTI plants with symbolic measurements and quantized control inputs. *International Conference on Control '94*, IEE, vol. 1, pp. 595-600, 1994.
22. P. Ramadge, W. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, vol.77, no.1, pp. 81-98, Jan. 1989.
23. S. Simic, K. Johansson, S. Sastry, and J. Lygeros. Towards a geometric theory of hybrid systems. In *Hybrid Systems: Computation and Control*, LNCS 1790, pp. 421-436, Springer-Verlag, 2000.
24. P.E. Souganidis. Approximation schemes for viscosity solutions of Hamilton-Jacobi equations. *Journal of Differential Equations*, vol. 59, no. 1, p. 1-43, August 1985.
25. J. Stiver, P. Antsaklis, M. Lemmon. A logical DES approach to the design of hybrid control systems. *Mathematical and computer modelling.* vol. 23, no. 11-1, pp. 55-76, June, 1996.
26. J.N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, vol. 40, no. 9, pp. 1528-1538, September 1995.
27. H.S. Witsenhausen. A class of hybrid-state continuous-time dynamic systems. *IEEE Trans. AC*, vol. 11, no. 2, pp. 161 - 167, April, 1966.
28. H. Wong-Toi. The synthesis of controllers for linear hybrid automata. In *Proc. 36th IEEE Conference on Decision and Control*, pp. 4607-4612, 1997.