

Hamiltonian-based Algorithm for Relaxed Optimal Control[†]

Y. Wardi, M. Egerstedt, and M.U. Qureshi*

Abstract—This paper concerns a first-order algorithmic technique for a class of optimal control problems defined on switched-mode hybrid systems. The salient feature of the algorithm is that it avoids the computation of Fréchet or Gâteaux derivatives of the cost functional, which can be time consuming, but rather moves in a projected-gradient direction that is easily computable (for a class of problems) and does not require any explicit derivatives. The algorithm is applicable to a class of problems where a pointwise minimizer of the Hamiltonian is computable by a simple formula, and this includes many problems that arise in theory and applications. The natural setting for the algorithm is the space of continuous-time relaxed controls, whose special structure renders the analysis simpler than the setting of ordinary controls. While the space of relaxed controls has theoretical advantages, its elements are abstract entities that may not be amenable to computation. Therefore, a key feature of the algorithm is that it computes adequate approximations to relaxed controls without losing its theoretical convergence properties. Simulation results, including cpu times, support the theoretical developments.

I. INTRODUCTION

Consider the following optimal control problem where the state equation is

$$\dot{x}(t) = f(x(t), u(t)), \quad (1)$$

$x(t) \in \mathbb{R}^n$ is the state variable, $u(t) \in \mathbb{R}^k$ is the input, or control variable, time t is confined to a given interval $[0, t_f]$, $f: \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^n$ is the dynamic-response function, the initial state $x(0) := x_0 \in \mathbb{R}^n$ is given, and the cost functional is

$$J = \int_0^{t_f} L(x(t), u(t)) dt + \phi(x(t_f)) \quad (2)$$

for cost functions $L: \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}$ and $\phi: \mathbb{R}^n \rightarrow \mathbb{R}$. Let $U \subset \mathbb{R}^k$ be a compact set, and consider the constraint that $u(t) \in U$ for every $t \in [0, t_f]$. To ensure that Eq. (1) has a unique, continuous and piecewise-differentiable solution, the integral in Eq. (2) is well defined, and other conditions mentioned in the sequel are satisfied, we make the following assumption.

Assumption 1: (1). The function $f(x, u)$ is twice-continuously differentiable in $x \in \mathbb{R}^n$ for every $u \in U$; the functions $f(x, u)$, $\frac{\partial f}{\partial x}(x, u)$, and $\frac{\partial^2 f}{\partial x^2}(x, u)$ are locally-Lipschitz continuous in $(x, u) \in \mathbb{R}^n \times U$; and there exists $K > 0$ such that, for every $x \in \mathbb{R}^n$ and for every $u \in U$, $\|f(x, u)\| \leq K(\|x\| + 1)$. (2). The function $L(x, u)$ is continuously differentiable in $x \in \mathbb{R}^n$ for every $u \in U$; and the functions $L(x, u)$ and $\frac{\partial L}{\partial x}(x, u)$ are locally-Lipschitz continuous in $(x, u) \in \mathbb{R}^n \times U$.

[†]Research supported in part by the NSF under Grant Number CNS-1239225.

*School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332. Email: ywardi@ece.gatech.edu, mag-nus@ece.gatech.edu, umer.qureshi@gatech.edu.

Define an admissible control to be a function $u: [0, t_f] \rightarrow U$ which is piecewise continuously differentiable and has a finite number of points of non-continuity. We denote an admissible control by the bar notation $\bar{u} := \{u(t)\}_{t \in [0, t_f]}$ to distinguish it from the value $u(t) \in U$ for a given $t \in [0, t_f]$. Denote by \mathcal{U} the space of admissible controls.

Observe that J as defined by Eqs. (1)–(2) can be viewed as a function of $\bar{u} \in \mathcal{U}$, hence denoted by $J(\bar{u})$ and called a cost functional. The optimal control problem is to minimize $J(\bar{u})$ over $\bar{u} \in \mathcal{U}$. Note that while the constraint set U is assumed to be compact it need not be convex, may have an empty interior and even be a finite set.

We set the optimal control problem in the framework of relaxed controls [1]–[3], described in detail in the next section. A relaxed control is a mapping μ from the interval $[0, t_f]$ into the set of Borel probability measures on the set U . It is an extension of the notion of the ordinary control, defined as a Lebesgue-measurable function $u: [0, t_f] \rightarrow U$. An ordinary control can be viewed as a relaxed control by associating with each $t \in [0, t_f]$ the Dirac measure at $u(t)$.

The space of relaxed controls is compact (in a suitable sense, discussed below) as well as convex, whereas the space of admissible controls typically is not compact and may not be convex. Therefore the setting of relaxed controls provides certain theoretical advantages over the setting of admissible controls, such as the existence of solutions to the optimal control problems [1]–[3] and the simplicity of analysis of conceptual (abstract) algorithms [4]. However, implementation may be more difficult due to the fact that relaxed controls are more abstract objects than admissible controls. Much of the analysis in the sequel addresses this point by identifying a class of hybrid systems where implementable algorithms are possible with little complexity. This class of systems is broad enough to include various problems of practical and theoretical interest.

Given a relaxed control, let $\{x(t)\}_{t \in [0, t_f]}$ and $\{p(t)\}_{t \in [0, t_f]}$ be the associated state trajectory and costate trajectory defined by Eqs. (8) and (10), below. The algorithm defines a descent direction by computing a pointwise minimizer of the Hamiltonian function $H(x(t), u, p(t))$ over $u \in U$, for a finite set of points $t \in [0, t_f]$. It then takes a suitable step in that direction to compute the next relaxed control.

Ref. [4] defined the algorithm and proved its convergence in the abstract setting of Eqs. (1) and (2). However, this setting does not include a general class of switched-mode hybrid systems of interest. The purpose of this paper is to close this gap, and extend the algorithm to systems and

problems defined as follows. The state equation is

$$\dot{x}(t) \in \{f_i(x(t), u_i(t)) : i = 1, \dots, M\}, \quad (3)$$

where the functions $f_i : \mathbb{R}^n \times \mathbb{R}^{k_i} \rightarrow \mathbb{R}^n$, $i = 1, \dots, M$ represent different modes of the system, $u_i \in U_i \subset \mathbb{R}^{k_i}$, and the mode-dependent set U_i is compact. At each time $t \in [0, t_f]$ the control variable, denoted by $v(t)$, consists of the mode-index i and the continuous-valued control, namely, $v(t) = (i(t), u_{i(t)}(t))$, where $i(t) \in \{1, \dots, M\}$ and $u_{i(t)}(t) \in U_{i(t)}$. Denote by V the set of pairs $v = (i, u_i)$ such that $i \in \{1, \dots, M\}$ and $u_i \in U_i$, and let \bar{v} denote an admissible control, namely the function $\{v(t)\}_{t \in [0, t_f]}$ which is piecewise continuously differentiable and has finite numbers of discontinuities. Denote by \mathcal{V} the space of admissible controls. The cost functional for the optimal control problem is

$$J = J(\bar{v}) = \int_0^{t_f} L_{i(t)}(x(t), u_{i(t)}(t)) dt + \phi(x(t_f)), \quad (4)$$

where $L_i : \mathbb{R}^n \times \mathbb{R}^{k_i} \rightarrow \mathbb{R}$, $i = 1, \dots, M$, are mode-dependent running cost functions, and $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ is a final-state cost function. The optimal control problem is to minimize $J = J(\bar{v})$ over $\bar{v} \in \mathcal{V}$.

The rest of the paper is organized as follows. Section II recounts relevant existing results, Section III extends the algorithm so as to be applicable for a class of switched-mode problems, and Section IV provides simulation results.

II. SURVEY OF ESTABLISHED RESULTS

This section surveys existing results which are relevant to the developments made in the sequel. In particular we discuss E. Polak's framework of infinite-dimensional optimization, the foundations of relaxed controls, and the preliminary algorithm presented in [4].

A. Optimality functions and sufficient descent

Let \mathcal{M} be a Hausdorff topological space with a Borel measure \mathcal{F} , and let $J : \mathcal{M} \rightarrow \mathbb{R}$ be a measurable function. Consider the abstract problem of minimizing $J(\bar{\mu})$ over $\bar{\mu} \in \mathcal{M}$. For a given necessary optimality condition, let $\Delta \subset \mathcal{M}$ be the set of points $\bar{\mu} \in \mathcal{M}$ where it is satisfied, and suppose that Δ is measurable. Let $\theta : \mathcal{M} \rightarrow \mathbb{R}^-$ be a measurable function. Polak defines θ to be an *optimality function* if (i) $\theta(\bar{\mu}) = 0$ iff $\bar{\mu} \in \Delta$, and (ii) $|\theta(\bar{\mu})|$ provides a measure of the extent to which $\bar{\mu}$ fails to satisfy the optimality condition.

Consider an iterative algorithm for minimizing J over \mathcal{M} , and let $\bar{\mu}_j \in \mathcal{M}$, $j = 1, 2, \dots$, be a sequence of points it computes from a given initial point $\bar{\mu}_0 \in \mathcal{M}$. Suppose that we can represent the computation of $\bar{\mu}_{j+1}$ from $\bar{\mu}_j$ via the notation $\bar{\mu}_{j+1} = T(\bar{\mu}_j)$, for a measurable mapping $T : \mathcal{M} \rightarrow \mathcal{M}$.

Definition 1: The algorithm is a *sufficient-descent* method with respect to θ if (i) for every $\bar{\mu} \in \mathcal{M}$, $J(T(\bar{\mu})) \leq J(\bar{\mu})$; and (ii) for every $\eta > 0$ there exists $\delta > 0$ such that, for every $\bar{\mu} \in \mathcal{M}$ such that $\theta(\bar{\mu}) < -\eta$,

$$J(T(\bar{\mu})) - J(\bar{\mu}) < -\delta. \quad (5)$$

E. Polak has offered the following characterization of an algorithm's convergence,

$$\lim_{j \rightarrow \infty} \theta(\bar{\mu}_j) = 0. \quad (6)$$

He applied it to the development of a general framework for analysis and design of algorithms, which is especially useful in infinite-dimensional optimization [5]. Such convergence can be ascertained by the following result whose proof can be found in [5].

Proposition 1: Suppose that $J(\bar{\mu})$ is bounded from below over $\bar{\mu} \in \mathcal{M}$. If the algorithm is of sufficient descent, then every sequence $\{\bar{\mu}_j\}_{j=1}^\infty$ of iteration points computed by the algorithm, satisfies Eq. (6).

B. Relaxed Controls

The theory of relaxed controls was developed in the late nineteen-sixties [1]–[3], and more recent surveys can be found in [6], [7]. This subsection summarizes its main points which are relevant to the present paper.

Consider the optimal control problem defined in Section I. Let M denote the space of Borel probability measures on the set U , and denote by μ a particular measure in M . A relaxed control associated with the system (1) is a mapping $\mu : [0, t_f] \rightarrow M$ which is measurable in the following sense: For every continuous function $\zeta : U \rightarrow \mathbb{R}$, the function $\int_U \zeta(u) d\mu(t)$ is Lebesgue measurable in t . We denote the space of relaxed controls by \mathcal{M} , and an element in this space is denoted by $\bar{\mu} := \{\mu(t)\}_{t \in [0, t_f]}$.

The space of relaxed controls is endowed with the weak star topology whereby $\lim_{k \rightarrow \infty} \bar{\mu}_k = \bar{\mu}$ if for every function $\psi : [0, t_f] \times U \rightarrow \mathbb{R}$ which is measurable and absolutely integrable in t on $[0, t_f]$ for every $u \in U$, and continuous on U for every $t \in [0, t_f]$,

$$\lim_{k \rightarrow \infty} \int_0^{t_f} \int_U \psi(t, u) d\mu_k(t) dt = \int_0^{t_f} \int_U \psi(t, u) d\mu(t) dt. \quad (7)$$

The space \mathcal{M} is compact in the weak star topology. Every ordinary control \bar{u} is associated with the relaxed control $\bar{\mu}$ by defining $\mu(t)$ as the Dirac measure on $u(t)$. Moreover, the space of ordinary controls is dense in \mathcal{M} in the weak-star topology, and hence the space of admissible controls, \mathcal{U} , is dense in \mathcal{M} as well.

An extension of the optimal control problem described by Eqs. (1) and (2) to the setting of relaxed controls is defined by the following equations. The relaxed state equation is

$$\dot{x}(t) = \int_U f(x(t), u) d\mu(t) \quad (8)$$

with the same boundary condition $x_0 = x(0)$ as for (1), and the relaxed cost functional is

$$J(\bar{\mu}) = \int_0^{t_f} \int_U L(x(t), u) d\mu(t) dt + \phi(x(t_f)). \quad (9)$$

The relaxed optimal control problem is to minimize $J(\bar{\mu})$ over $\bar{\mu} \in \mathcal{M}$, and a minimum exists due to the compactness of \mathcal{M} . The relaxed costate (adjoint) equation is

$$\dot{p}(t) = - \int_U \left(\frac{\partial f}{\partial x}(x(t), u)^\top p(t) + \frac{\partial L}{\partial x}(x(t), u)^\top \right) d\mu(t) \quad (10)$$

with the boundary condition $p(t_f) = \nabla\phi(x(t_f))$, and the relaxed Hamiltonian is defined as

$$H(x(t), \mu(t), p(t)) = \int_U \left(p(t)^\top f(x(t), u) + L(x(t), u) \right) d\mu(t). \quad (11)$$

The relaxed maximum principle states that if $\bar{\mu} \in \mathcal{M}$ is a solution for the relaxed optimal control problem then $\mu(t)$ minimizes the Hamiltonian at almost every time-point $t \in [0, t_f]$; see [7].

C. Preliminary version of the algorithm

This subsection describes the algorithm presented in [4], and recounts some theoretical results whose proofs can be found in the latter reference.

Consider a relaxed control $\bar{\mu} \in \mathcal{M}$, and let $x(t)$ and $p(t)$ denote the state variable and costate variable ($t \in [0, t_f]$) defined by Eqs. (8) and (10), respectively. The following optimality function is associated with the relaxed maximum principle:

$$\theta(\bar{\mu}) := \min_{\bar{v} \in \mathcal{M}} \left\{ \int_0^{t_f} \left(H(x(t), v(t), p(t)) - H(x(t), \mu(t), p(t)) \right) dt \right\}. \quad (12)$$

The minimum in (12) exists due to the compactness of the space \mathcal{M} in the weak-star topology. Fix $\eta \in (0, 1)$. Due to the density of the space of admissible controls in \mathcal{M} , there exists an admissible control \bar{u}_η^* such that

$$\int_0^{t_f} \left(H(x(t), \bar{u}_\eta^*(t), p(t)) - H(x(t), \mu(t), p(t)) \right) dt < (1 - \eta)\theta(\bar{\mu}). \quad (13)$$

Furthermore, \bar{u}_η^* can have the following form: There exists a finite grid $\mathcal{G} \subset [0, t_f]$ such that for all $t \in \mathcal{G}$, $\bar{u}_\eta^*(t) \in \operatorname{argmin}\{H(x(t), u, p(t)) : u \in U\}$, and for every $t \in [0, t_f]$, $\bar{u}_\eta^*(t)$ is a zero-order hold (interpolation) of its values at $t \in \mathcal{G}$ (see [4]).

Similarly to the presentation in Section II.A, we describe the algorithm by specifying its main loop, represented by a mapping $T : \mathcal{M} \rightarrow \mathcal{M}$.

Given constants $\eta \in (0, 1)$, $\alpha \in (0, 1)$, and $\beta \in (0, 1)$.

Given $\bar{\mu} \in \mathcal{M}$, compute $T(\bar{\mu}) \in \mathcal{M}$ as follows.

Algorithm 1: Step 1: Compute $\{x(t)\}$ and $\{p(t)\}$, $t \in [0, t_f]$, by numerical means using Eqs. (8) and (10).

Step 2: Compute an admissible control $\bar{u}_\eta^* \in \mathcal{U}$ satisfying Eq. (13).

Step 3: Compute the largest λ from the set $\{1, \beta, \beta^2, \dots\}$ such that,

$$J((1 - \lambda)\bar{\mu} + \lambda\bar{u}_\eta^*) - J(\bar{\mu}) < \alpha\lambda\theta(\bar{\mu}). \quad (14)$$

Denote the resulting λ by $\lambda_{\bar{\mu}}$.

Step 4: Set $T(\bar{\mu}) = (1 - \lambda_{\bar{\mu}})\bar{\mu} + \lambda_{\bar{\mu}}\bar{u}_\eta^*$. \square

The step size defined in Step 3 of the algorithm is due to Armijo [5], and we remark that such a step size has been used extensively in gradient-descent optimization including optimal control problems [5], [8]–[11].

The following convergence result has been proved in [4].

Theorem 1: For every $\eta \in (0, 1)$ there exists $\bar{\alpha} \in (0, 1)$ such that, for every choices of $\alpha \in (0, \bar{\alpha})$ and $\beta \in (0, 1)$, Algorithm 1 has the property of sufficient descent with respect to the relaxed maximum principle. \square

III. SWITCHED-MODE HYBRID SYSTEMS

In recent years there has been a growing interest in the hybrid optimal control problem defined by Eqs. (3) and (4). A number of algorithmic approaches emerged, including first- and second-order gradient-descent techniques [8], [12]–[15], zoning algorithms based on the geometric properties of the underlying systems [16], [17], projection-based algorithms [8], [9], [11], [18], methods based on dynamic programming and convex optimization [19], and needle-variations techniques [20], [21]. Relaxed-control algorithms were proposed in Refs. [10], [22]. An embedded control approach was analyzed in [23] and tested in conjunction with MATLAB's `fmincon` nonlinear-programming solver [23], [24]. The algorithm in [10] operates in the setting of embedded controls as well. A comprehensive survey of algorithmic techniques for the hybrid optimal control problem can be found in [25].

The space of embedded controls provides a natural setting for the hybrid optimal control problem. Therefore, following [10], [23], [24] we also define the algorithm presented below in this setting. Embedded controls are defined as follows (see [7], [23]): Let W denote the set of M -tuples of pairs, $((\alpha_1, u_1), (\alpha_2, u_2), \dots, (\alpha_M, u_M))$, where $\alpha_i \geq 0 \forall i = 1, \dots, M$, $\sum_{i=1}^M \alpha_i = 1$, and $u_i \in U_i$, $i = 1, \dots, M$. An embedded control is a Lebesgue measurable function $w : [0, t_f] \rightarrow W$, and we denote the space of embedded controls by \mathcal{W} . Furthermore, we denote an embedded control $\{w(t)\}_{t \in [0, t_f]}$ by $\bar{w} \in \mathcal{W}$. It can be seen that \mathcal{W} lies between the space of ordinary controls and the space of relaxed controls. For a given $\bar{w} \in \mathcal{W}$, the embedded state equation is defined by

$$\dot{x}(t) = \sum_{i=1}^M \alpha_i(t) f_i(x(t), u_i(t)) \quad (15)$$

with the given boundary condition $x(0) = x_0 \in R^n$, the embedded cost functional has the form

$$J = \sum_{i=1}^M \int_0^{t_f} \alpha_i(t) L_i(x(t), u_i(t)) dt + \phi(x(t_f)), \quad (16)$$

and the embedded costate equation is

$$\dot{p}(t) = - \sum_{i=1}^M \alpha_i(t) \left[\left(\frac{\partial f_i}{\partial x}(x(t), u_i(t)) \right)^\top p(t) + \left(\frac{\partial L_i}{\partial x} L_i(x(t), u_i(t)) \right)^\top \right] \quad (17)$$

with the boundary condition $p(t_f) = \nabla\phi(x(t_f))$. For a detailed expositions of embedded controls, see [7], [23].

The class of problems to which the algorithm defined below is applicable satisfies the following assumptions.

Assumption 2: For every $x \in R^n$, and for every $i = 1, \dots, M$, (i) $f_i(x, u_i)$ is affine in $u_i \in U_i$, and (ii) $L_i(x, u_i)$ is convex in $u_i \in U_i$. \square

Part (i) of the assumption means that for every $i = 1, \dots, M$ there exist functions $\Phi_i: \mathbb{R}^n \rightarrow \mathbb{R}^{n \times k_i}$ and $\Psi_i: \mathbb{R}^n \rightarrow \mathbb{R}^n$ such that,

$$f_i(x, u_i) = \Phi_i(x)u_i + \Psi_i(x). \quad (18)$$

Assumption 3: For every $i = 1, \dots, M$, (i) the functions $\Phi_i(x)$ and $\Psi_i(x)$ are twice-continuously differentiable, and (ii) the function $L_i(x, u)$ satisfies Assumption 1.

Given two embedded controls, $\bar{w}_1 \in \mathcal{W}$ and $\bar{w}_2 \in \mathcal{W}$, and given $\lambda \in [0, 1]$, we use the notation $(1 - \lambda)\bar{w}_1 \oplus \lambda\bar{w}_2$ to designate the convex combination $(1 - \lambda)\bar{w}_1 + \lambda\bar{w}_2$ in the sense of measures. Thus, if $w_1(t) = ((\alpha_{1,1}(t), u_{1,1}(t)), \dots, (\alpha_{1,M}(t), u_{1,M}(t)))$ and $w_2(t) = ((\alpha_{2,1}(t), u_{2,1}(t)), \dots, (\alpha_{2,M}(t), u_{2,M}(t)))$, then the state equation of $(1 - \lambda)\bar{w}_1 \oplus \lambda\bar{w}_2$ is

$$\begin{aligned} \dot{x}(t) = & (1 - \lambda) \sum_{i=1}^M \alpha_{1,i}(t) f_i(x(t), u_{1,i}(t)) \\ & + \lambda \sum_{i=1}^M \alpha_{2,i}(t) f_i(x(t), u_{2,i}(t)), \end{aligned} \quad (19)$$

and the cost functional is

$$\begin{aligned} & J((1 - \lambda)\bar{w}_1 \oplus \lambda\bar{w}_2) \\ = & (1 - \lambda) \sum_{i=1}^M \int_0^{t_f} \alpha_{1,i}(t) L_i(x(t), u_{1,i}(t)) dt \\ & + \lambda \sum_{i=1}^M \int_0^{t_f} \alpha_{2,i}(t) L_i(x(t), u_{2,i}(t)) dt + \phi(x(t_f)). \end{aligned} \quad (20)$$

The following algorithm is described by specifying its main loop, as for Algorithm 1. The input to the main loop is $\bar{w} \in \mathcal{W}$, and the corresponding output is $\bar{y} \in \mathcal{W}$. The first three steps of the algorithm are identical to those of Algorithm 1, and the difference is in Step 4.

Given constants $\eta \in (0, 1)$, $\alpha \in (0, 1)$, and $\beta \in (0, 1)$.

Algorithm 2: Given

$\bar{w} = ((\alpha_1(t), u_1(t)), \dots, (\alpha_M(t), u_M(t))), t \in [0, t_f]$.

Step 1: Compute $\{x(t)\}$ and $\{p(t)\}$, $t \in [0, t_f]$, by numerical means, using Eqs. (15) and (17).

Step 2: Compute an admissible control $\bar{u}_\eta^* \in \mathcal{U}$ satisfying Eq. (13). For every $t \in [0, t_f]$, $u_\eta^*(t) = (j(t), u_{j(t)}^*(t))$ for some $j(t) \in \{1, \dots, M\}$ and $u_{j(t)}^*(t) \in U_j(t)$, and we can view it as an embedded control of the form $u_\eta^*(t) = ((\alpha_1^*(t), u_1^*(t)), \dots, (\alpha_M^*(t), u_M^*(t)))$, where, $\alpha_{j(t)}^*(t) = 1$, and for all $i \neq j(t)$, $\alpha_i^*(t) = 0$.

Step 3: Compute the largest λ from the set $\{1, \beta, \beta^2, \dots\}$ such that,

$$J((1 - \lambda)\bar{w} \oplus \lambda\bar{u}_\eta^*) - J(\bar{w}) < \alpha\lambda\theta(\bar{w}). \quad (21)$$

Denote the resulting λ by $\lambda_{\bar{w}}$.

Step 4: For every $t \in [0, t_f]$, define $\gamma_i(t) = (1 - \lambda_{\bar{w}})\alpha_i(t) + \lambda_{\bar{w}}\alpha_i^*(t)$, and define $\varepsilon_i(t) = \lambda_{\bar{w}}\alpha_i^*(t)/\gamma_i(t)$. For every $i = 1, \dots, M$ define $\tilde{u}_i(t) = (1 - \varepsilon_i(t))u_i(t) + \varepsilon_i(t)u_i^*(t)$, and set

$$\bar{y} = ((\gamma_1(t), \tilde{u}_1(t)), \dots, (\gamma_M(t), \tilde{u}_M(t))), t \in [0, t_f]. \quad (22)$$

□

The following convergence result is proved in Ref. [26].

Theorem 2: Suppose that Assumption 2 and Assumption 3 are in force. For every $\eta \in (0, 1)$ there exists $\bar{\alpha} \in (0, 1)$ such that, for every choices of $\alpha \in (0, \bar{\alpha})$ and $\beta \in (0, 1)$, Algorithm 2 has the property of sufficient descent with respect to the relaxed maximum principle. □

IV. EXAMPLES

This section presents three examples: an autonomous switched-mode system, an unstable hybrid system, and a spring-mass damper system. The algorithm was coded by a MATLAB script, and run on a system based on an Intel Core i5 processor with 2.8 GHz clock. All of the numerical integrations were performed by the forward Euler method or the trapezoidal method.

A. Curve tracking in a double-tank system

Consider two cylindrical fluid tanks situated one on top of the other, each having a hole at the bottom. Fluid flows into each tank from the top and out through the hole. The input flow to the upper tank comes from a valve-controlled hose, and the input flow to the lower tank consists of the output flow from the upper tank. Let $v(t)$ denote the input flow rate to the upper tank, and let $x_1(t)$ and $x_2(t)$ denote the amount of fluid in the upper tank and lower tank, respectively. $v(t)$ is the control input to the system, and $x(t) := (x_1(t), x_2(t))^T$ is its state variable. By Toricelli's law the state equation is

$$\dot{x}(t) = \begin{pmatrix} v(t) - \sqrt{x_1(t)} \\ \sqrt{x_1(t)} - \sqrt{x_2(t)} \end{pmatrix}, \quad (23)$$

and we assume that the initial state is $x(0) = (2.0, 2.0)^T$. The control input $v(t)$ is assumed to be constrained to the two-point set $V := \{1.0, 2.0\}$, and hence we can view the system as having two modes, mode 1 when $v(t) = 1$, and mode 2 when $v(t) = 2$. We consider the problem of having the fluid level in the lower tank track a reference curve $\{r(t)\}_{t \in [0, t_f]}$ for a given $t_f > 0$, and correspondingly we minimize the cost functional

$$J := 2 \int_0^{t_f} (x_2(t) - r(t))^2 dt. \quad (24)$$

This is an autonomous switched-mode system without a continuous-valued control u . Therefore the Hamiltonian function is $H(x, v, p) = p^T f(x, v) + L(x, t)$, with $L(x, t) = 2(x - r(t))^2$, and for given $x \in \mathbb{R}^2$ and $p \in \mathbb{R}^2$, its pointwise minimizer is $v^*(t) \in \{1, 2\}$. A measure $\mu \in M$ can be represented by a point $p \in [0, 1]$, where $\mu(\{1\}) = p$ and $\mu(\{2\}) = 1 - p$, and hence a relaxed control is a function $\mu: [0, t_f] \rightarrow [1, 2]$. Such systems are simpler than the controlled-systems discussed in Section III, the Hamiltonian is easily minimized (pointwise), and Algorithm 2 is reduced to Algorithm 1. We provide this example nonetheless in order to highlight some features of the algorithm.

This problem was addressed in [4], [10], [24] with a constant target $r(t) = 3.0$, while here we track the time-varying target curve $r(t) = 0.5 \sin(0.1\pi t) + 2.5$ over $t \in [0, 30]$. The algorithm's parameters are $\alpha = 0.5$ and $\beta = 0.5$. All of the numerical integrations are performed by the forward Euler

method with the time step $\Delta t = 0.01$. The initial control is $v(t) = 2 \forall t \in [0, t_f]$, and its cost is $J(\bar{v}_1) = 84.185$.

The algorithm was run for 100 iterations, and its execution took 17.207 seconds of cpu time. Figure 1 depicts the graph of $J(\bar{v}_k)$ vs. the iteration count k , and it exhibits sharp decrease before flattening after about 10 iterations. The final cost is $J(\bar{v}_{100}) = 2.627$, and the graphs of the corresponding $x_2(t)$ (solid curve) and its target $r(t)$ (dashed curve) are shown in Figure 2 for the sake of comparison. After the run we projected the final relaxed control \bar{v}_{100} onto the space of ordinary controls by using pulse-width modulation as in [4], and the resulting control, denoted by \bar{v}_{fin} , has a cost of $J(\bar{v}_{fin}) = 2.7051$.

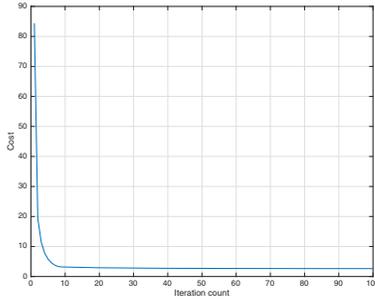


Fig. 1. Double-tank system: Cost function vs. k

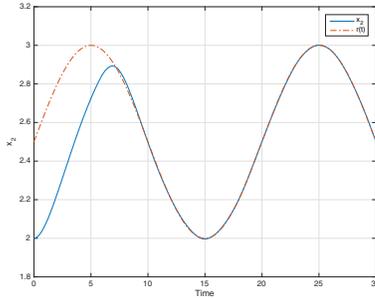


Fig. 2. Double-tank system: $x_2(t)$ and $r(t)$

Finally, in order to explore ways to reduce the run times of the algorithm we increased the integration step size by a factor of 10 to $\Delta t = 0.1$. The resulting final cost was $J(\bar{v}_{100}) = 2.662$, and the cpu time was 1.433 seconds.

B. Control of an unstable hybrid system

The following LQR system was considered in [14], [17]. The system has two modes, indexed by $i = 1, 2$. The dynamic response functions are $f_i(x, u) = A_i x + b_i u$, where $x \in \mathbb{R}^2$, $u \in \mathbb{R}$, $A_i \in \mathbb{R}^{2 \times 2}$, and $b_i \in \mathbb{R}^{2 \times 1}$. The matrices A_i and b_i are

$$A_1 = \begin{pmatrix} 0.6 & 1.2 \\ -0.8 & 3.4 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 4.0 & 3.0 \\ -1.0 & 0 \end{pmatrix},$$

$b_1 = (1, 1)^\top$, and $b_2 = (2, -1)^\top$. The initial condition is $x_0 = (0, 2)^\top$, and the final time is $t_f = 2.0$. The cost functional is $J = \int_0^2 \frac{1}{2} (x_2(t) - 2)^2 + u(t)^2 dt + \frac{1}{2} (x_1(2) - 4)^2 +$

$\frac{1}{2} (x_2(2) - 2)^2$. According to the problem formulation in Refs. [14], [17] the sequence of modes is fixed, and the control variable v consists of the switching times between them and the continuous-valued input $u(t)$, $t \in [0, 2]$. In this paper the mode-sequence is not fixed, and the control variable consists of the mode-schedule as well as the continuous-valued control $u(t)$, $t \in [0, 2]$. We use the trapezoidal method for integrating the differential equations. The initial guess for the algorithm consists of mode 1 and $u(t) = 0$ for all $t \in [0, 2]$, and we ran the algorithm for 400 iterations.

The dominant eigenvalue of both matrices A_1 and A_2 is 3.0, hence the system is highly unstable. Therefore the algorithm did not work well with single-shooting integrations of the state equation, and yielded a final cost of about 14.2, which is higher than that obtained in [14], [17] (9.766) with a far-more-restricted admissible control space. To get around this difficulty we used multi-shooting integrations in the following way. With N denoting the number of shootings, we divided the time-interval $[0, 2]$ into N equal-lengths subintervals with end-points $0 < \tau_1 < \dots < \tau_{N-1} < 2$, introduced the additional variables z_j , $j = 1, \dots, N-1$ as the initial condition for the state equation during the subinterval beginning at τ_j , and added to the cost the penalty term $K \sum_{j=1}^{N-1} \|x(\tau_j^-) - z_j\|^2$. The penalty constant K was determined by the formula $K = 2.5(N-1)$, since we felt that a higher penalty constant was needed for larger numbers of shooting intervals. The integration step size was set to $\Delta t = \frac{0.1}{N-1}$.

After some experimentation we chose $N = 10$, hence $K = 22.5$ and $\Delta t = 0.011$. A 400-iteration run of the algorithm took 14.4 seconds of cpu time, and yielded the final cost of $J(\bar{v}_{400}) = 7.0913$. Additional runs supported this result and indicated that the obtained cost is practically close to the minimum. Moreover, the graph of $J(\bar{v}_k)$ vs. $k = 1, 2, \dots$, not presented here for reasons of space limitations, displays a similar L-shaped curve as in Figure 2. The final state trajectories $x_1(1)$ and $x_2(t)$ are shown in Figure 3.

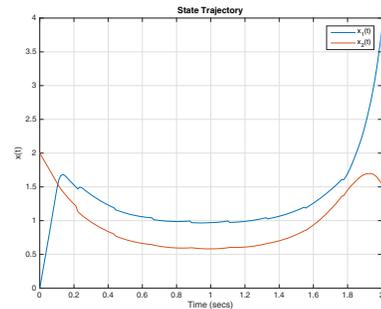


Fig. 3. Unstable hybrid system: final $x_1(t)$ and $x_2(t)$

C. Controlling a mass-spring damper system

The problem described in this subsection has been considered in [27] which applied to it model-predictive control, and a similar problem was solved in [23] by a numerical algorithm.

Consider a mass connected to ground by a spring in series with a damper that represents viscous friction. Let $u(t)$ be an applied external force, and let $x_1(t)$ and $x_2(t)$ be the mass' position and velocity. The system has two modes, indexed by $i \in \{1, 2\}$, representing two levels of viscosity. The state equation is

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ M\dot{x}_2(t) &= -kx_1(t) - b_i x_2(t) + u(t), \end{aligned} \quad (25)$$

where M the mass; the spring coefficient $k(x_1)$ is $k(x_1) = x_1 + 1$ if $x_1 \leq 1$, and $k(x_1) = 3x_1 + 7.5$ if $x_1 > 1$; and the viscous friction coefficients are $b_1 = 1$ and $b_2 = 50$. The initial condition is $x_0 = (3, 4)^\top$. We take the mass to be $M = 1$. The cost functional is $J = \int_0^{t_f} (||x(t)||^2 + L_i(u(t)))dt + ||x(t_f)||^2$, where the mode-dependent cost function is $L_1(u) = 0.2u^2$, and $L_2(u) = 0.2u^2 + 1$. We impose the constraints that, for all $t \in [0, t_f]$, $|x_j(t)| \leq 5$, $j = 1, 2$, and $|u(t)| \leq 10$; and the final-state constraint $|x_j(t_f)| \leq 0.01$, $j = 1, 2$. We chose the final time to be $t_f = 12.0$.

In order to satisfy the final-state constraints we appended the cost functional by the penalty term $5||x_1(t_f)||^2 + 30||x_2(t_f)||^2$. We applied Algorithm 2 with the parameters, $\alpha = 0.01$ and $\beta = 0.5$, and used the integration step size $\Delta t = 0.01$. The initial guess was $\alpha_1(t) = 1$ (i.e., mode 1), and $u_1(t) = u_2(t) = 0$ for all $t \in [0, t_f]$. 50 iterations took 11.6950 seconds of cpu time, and reduced the cost from its initial value of 94.0906 to its final value of 14.5166. The state trajectory is shown in Figure 4, and the final state is $x(t_f) = (0.001, -0.0076)^\top$. A PWM-based projection of the final embedded control onto the space of ordinary controls incurs the cost J , excluding the penalty term, of 15.1954.

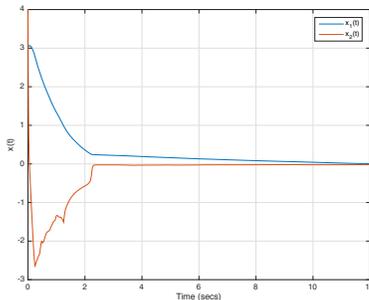


Fig. 4. Mass-spring-damper system: $x_1(t)$ and $x_2(t)$

REFERENCES

- [1] E.J. McShane. Ralexed Controls and Variational Problems. *SIAM Journal on Control*, Vol. 5, pp. 438–485, 1967.
- [2] L.C. Young. *Lectures on the calculus of variations and optimal control theory*. Foreword by Wendell H. Fleming. W.B. Saunders Co., Philadelphia, 1969.
- [3] J. Warga. *Optimal Control of Differential and Functional Equations*, Academic Press, 1972.
- [4] M.T. Hale, Y. Wardi, H. Jaleel, and M. Egerstedt. Hamiltonian-Based Algorithm for Optimal Control. *Arxiv*, <http://arxiv.org/abs/1603.02747>, March 2016. Also submitted to *Nonlinear Analysis: Hybrid Systems*.
- [5] E. Polak. *Optimization Algorithms and Consistent Approximations*. Springer-Verlag, New York, New York, 1997.

- [6] R. Vinter. *Optimal Control*, Birkhauser, Boston, Massachusetts, 2000.
- [7] L.D. Berkovitz and N.G. Medhin. *Nonlinear Optimal Control Theory*, Chapman & Hall, CRC Press, Boca Raton, Florida, 2013.
- [8] T. Caldwell and T.D. Murphey. Switching mode generation and optimal estimation with application to skid-steering. *Automatica*, vol. 47, no. 1, pp. 50–64, 2011.
- [9] T. Caldwell and T.D. Murphey. Single integration optimization of linear time-varying switched systems. *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1592–1597, 2012.
- [10] R. Vasudevan, H. Gonzalez, R. Bajcsy, and S.S. Sastry. Consistent Approximations for the Optimal Control of Constrained Switched Systems - Part 1: A Conceptual Algorithm, and Part 2: An Implementable Algorithm. *SIAM Journal on Control and Optimization*, Vol. 51, pp. 4663–4483 (Part 1) and pp. 4484–4503 (Part 2), 2013.
- [11] T. Caldwell and T. Murphey. Projection-Based Iterative Mode Scheduling for Switched Systems. *Nonlinear Analysis: Hybrid Systems*, to appear, 2016.
- [12] X. Xu and P.J. Antsaklis. Optimal Control of Switched Systems via Nonlinear Optimization Based on Direct Differentiations of Value Functions. *International Journal of Control*, Vol. 75, pp. 1406–1426, 2002.
- [13] M.S. Shaikh and P. Caines. On Trajectory Optimization for Hybrid Systems: Theory and Algorithms for Fixed Schedules. *IEEE Conference on Decision and Control*, Las Vegas, NV, Dec. 2002.
- [14] X. Xu and P. Antsaklis. Optimal control of switched-systems based on parameterization of the switching instants. *IEEE Trans. Automatic Control*, vol. 49, pp. 2–16, 2004.
- [15] M. Egerstedt, Y. Wardi, and H. Axelsson. Transition-Time Optimization for Switched Systems. *IEEE Transactions on Automatic Control*, Vol. AC-51, No. 1, pp. 110–115, 2006.
- [16] M.S. Shaikh and P.E. Caines. On the Optimal Control of Hybrid Systems: Optimization of Trajectories, Switching Times and Location Schedules. In *Proceedings of the 6th International Workshop on Hybrid Systems: Computation and Control*, pp. 466–481, Prague, The Czech Republic, 2003.
- [17] M.S. Shaikh and P.E. Caines. On the Hybrid Optimal Control Problem: Theory and Algorithms. *IEEE Trans. Automatic Control*, Vol. 52, pp. 1587–1603, 2007.
- [18] L. Miller and T.D. Murphey. Simultaneous optimal estimation of mode transition times and parameters applied to simple traction models. *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1496–1503, 2013.
- [19] S. Hedlund and A. Rantzer. Optimal control for hybrid systems. *Proc. 38th CDC*, Phoenix, Arizona, December 7–10, 1999.
- [20] H. Gonzalez, R. Vasudevan, M. Kamgarpour, S.S. Sastry, R. Bajcsy, and C. Tomlin. A Numerical Method for the Optimal Control of Switched Systems. *Proc. 49th CDC*, Atlanta, Georgia, pp. 7519–7526, December 15–17, 2010.
- [21] F. Taringoo and P.E. Caines. On the Optimal Control of Impulsive Hybrid Systems on Riemannian Manifolds. *SIAM Journal on Control and Optimization*, Vol. 51, Issue 4, pp. 3127–3153, 2013.
- [22] X. Ge, W. Kohn, A. Nerode, and J.B. Remmel. Hybrid systems: Chattering approximation to relaxed controls. *Hybrid Systems III: Lecture Notes in Computer Science*, R. Alur, T. Henzinger, E. Sontag, eds., Springer Verlag, Vol. 1066, pp. 76–100, 1996.
- [23] S.C. Bengea and R. A. DeCarlo. Optimal control of switching systems. *Automatica*, Vol. 41, pp. 11–27, 2005.
- [24] R.T. Meyer, M. Zefran, and R.A. DeCarlo. Comparison of the Embedding Method to Multi-Parametric Programming, Mixed-Integer Programming, Gradient Descent, and Hybrid Minimum Principle Based Methods. *IEEE Transactions on Control Systems Technology*, Vol. 22, No. 5, pp. 1784–1800, 2014.
- [25] H. Lin and P. J. Antsaklis. Hybrid Dynamical Systems: An Introduction to Control and Verification. *Foundations and Trends in Systems and Control*, Vol. 1, No. 1, pp. 1–172, March 2014.
- [26] Y. Wardi, M. Egerstedt, and M.U. Qureshi. Hamiltonian-Based Algorithm for Relaxed Optimal Control. *Arxiv*, <http://arxiv.org/abs/1609.03183>, September 2016.
- [27] F. Borrelli, A. Bemporad, and M. Morari. *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, Cambridge, 2013.