Automatica 62 (2015) 184–192

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Local optimization of dynamic programs with guaranteed satisfaction of path constraints[☆]

Jun Fu[a], Johannes M.M. Faust[c], Benoît Chachuat[b], Alexander Mitsos[c,1]

[a] Department of Mechanical Engineering, Massachusetts Institute of Technology (MIT), Cambridge, MA, 02139, USA
[b] Centre for Process Systems Engineering, Department of Chemical Engineering, Imperial College, London, SW7 2AZ, UK
[c] AVT Process Systems Engineering (SVT), RWTH Aachen University, Turmstrasse 46, 52064 Aachen, Germany

ARTICLE INFO

Article history:
Received 15 August 2014
Received in revised form
13 June 2015
Accepted 30 August 2015

Keywords:
Dynamic optimization
Path constraints
Semi-infinite programs
Optimization with dynamics embedded
Optimal control

ABSTRACT

An algorithm is proposed for locating a feasible point satisfying the KKT conditions to a specified tolerance of feasible inequality-path-constrained dynamic programs (PCDP) within a finite number of iterations. The algorithm is based on iteratively approximating the PCDP by restricting the right-hand side of the path constraints and enforcing the path constraints at finitely many time points. The main contribution of this article is an adaptation of the semi-infinite program (SIP) algorithm proposed in Mitsos (2011) to PCDP. It is proved that the algorithm terminates finitely with a guaranteed feasible point which satisfies the first-order KKT conditions of the PCDP to a specified tolerance. The main assumptions are: (i) availability of a nonlinear program (NLP) local solver that generates a KKT point of the constructed approximation to PCDP at each iteration if this problem is indeed feasible; (ii) existence of a Slater point of the PCDP that also satisfies the first-order KKT conditions of the PCDP to a specified tolerance; (iii) all KKT multipliers are nonnegative and uniformly bounded with respect to all iterations. The performance of the algorithm is analyzed through two numerical case studies.

1. Introduction

Dynamic optimization refers to mathematical programs whereby the objective and constraint functions depend on the solution of differential or difference equations. Dynamic optimization has been widely applied in chemical engineering (Biegler, 2010; Srinivasan, Palanki, & Bonvin, 2003), mechanical engineering (Hussein & Bloch, 2008; Shin & McKay, 1986), aerospace engineering (Bainum & Kumar, 1980) and other disciplines (Floudas et al., 1999). Constrained dynamic optimization problems are practically important, e.g., to enforce product quality or to guarantee safety (Feehery & Barton, 1998; Srinivasan et al., 2003). Constraints fall in either one of two categories, namely point constraints and path constraints. The former are usually expressed as functions of the states at the end of time horizon, whereas the latter are functions of the states and/or controls over the entire time horizon. The focus of this article is on dynamic optimization with path constraints. Point constraints, which do not pose any further complication for the approach herein, are omitted for simplicity. Throughout the article, it is assumed that a control vector parameterization has been performed, i.e., a finite number of decision variables is assumed.

Numerical solution methods for such dynamic optimization problems rely on nonlinear programming (NLP) techniques, either with or without parameterization of the state trajectories. In the simultaneous method, also known as orthogonal collocation approach (Betts & Huffman, 1992; Biegler, 2007; Tsang, Himmelblau, & Edgar, 1975), the state trajectories are parameterized and the residuals of the differential equations are enforced as constraints at specified collocation times. In the sequential method (Biegler, 2010; Goh & Teo, 1988), the state trajectories are regarded as functions of the control decision variables. In the direct multiple shooting method (Bock & Plitt, 1984), the state trajectories are formed by piecing together those of finite single shooting problems on the corresponding subintervals over which the parameterized control is applied (see p. 243 in Bock & Plitt, 1984).

☆ Dr. Jun Fu's work was partially supported by Natural Sciences and Engineering Research Council of Canada (NSERC) grant 387509. Prof. Mitsos would like to acknowledge funding from the DFG-Cluster grant DFG-PAK 635/1 Project MA 1188/34-1 "Strategie zur robusten dynamischen Echtzeitoptimierung" and discussions within that cluster. The material in this article was not presented at any conference. This paper was recommended for publication in revised form by Editor Berç Rüstem.

E-mail addresses: junfu@mit.edu, fujuncontrol@gmail.com (J. Fu), johannes.faust@avt.rwth-aachen.de (J.M.M. Faust), b.chachuat@imperial.ac.uk (B. Chachuat), amitsos@alum.mit.edu (A. Mitsos).

[1] Tel.: +49 2418094704; fax: +49 2418092326.

http://dx.doi.org/10.1016/j.automatica.2015.09.013
0005-1098/

Techniques for dealing with inequality path constraints have been developed for the three methods, e.g., Biegler (2010), Bock and Plitt (1984), Dai and Cochran (2009), Feehery and Barton (1998), Fikar (2001), Goh and Teo (1988), Jacobson and Lele (1969), Li, Yu, Teo, and Duan (2011), Loxton, Teo, Rehbock, and Yiu (2009), Parida and Raha (2009), Schlegel, Stockmann, Binder, and Marquardt (2005), Teo, Rehbock, and Jennings (1993), Vassiliadis, Sargent, and Pantelides (1994) and White, Perkins, and Espie (1996). The common feature of these techniques is that the path constraints are (explicitly or implicitly) enforced at finitely many points only. Particularly popular are discretization of the path constraints as interior-point constraints and transcription as integral constraint, possibly used in combination (Vassiliadis et al., 1994). The former method enforces the path constraint at a finite number of time points, so constraint violation can occur at any point other than those where the constraints are enforced. The latter one enforces a time-integral of the constraint violation as a constraint, which is allowed to be less than or equal to a small positive constant for regularity reasons, therefore allowing for small violations along the time horizon, too. Particularly relevant to this article are the works by Chen and Vassiliadis (2005) and by Potschka, Bock, and Schlöder (2009). Chen and Vassiliadis (2005) presents an algorithm solving path-constrained optimal control problems, yet violation of the path constraints by a small amount cannot be prevented for a finite number of iterations. Potschka et al. (2009) develops an algorithm solving path-constrained optimal control problems (without proof of convergence), but to the authors' best knowledge does not achieve both guaranteed rigorous satisfaction of path constraints and finite convergence. More recently, Zhao and Stadtherr (2011) have described an algorithm capable of locating an ϵ-estimated global optimum of path-constrained dynamic systems with guaranteed satisfaction of the path constraints, but this rigorous algorithm uses a deterministic global optimization approach, and as such it is currently applicable to problems with a small number of degrees of freedom only. Note that indirect methods can be used for the continuous optimal control problem under the assumption that the switching structure of the path constraint is known (Hannemann-Tamás & Marquardt, 2012). Note also that the α method in Peter, Parida, and Raha (2010) can be used for infinite dimensional problems subject to the regularization assumptions. With the exception of Zhao and Stadtherr (2011), to our best knowledge, *none of the existing methods can guarantee rigorous satisfaction of path constraints over the entire time horizon within a finite number of iterations*. It is the focus of this article to develop an algorithm for path-constrained dynamic optimization problems that relies on local optimization techniques, while coming with a certification of feasibility for the path constraints.

An important class of optimization problems are semi-infinite programs (SIP), namely optimization problems with a finite number of decision variables but an infinite number of constraints. For theoretical developments and applications of SIP, we refer the reader to reviews (Hettich & Kortanek, 1993; Polak, 1987) and latest results (Mitsos, 2011; Mitsos & Tsoukalas, 2015; Stein & Steuermann, 2012). In the context of path-constrained dynamic optimization, SIP formulations arise naturally if time is viewed as the (single) parameter of SIP (Loxton et al., 2009; Sachs, 1998). Through this connection, the work by Chen and Vassiliadis (2005) can be seen as an adaptation of the SIP algorithm of Blankenship and Falk (1976) to path-constrained dynamic optimization. The work by Potschka et al. (2009) is essentially a first combination of local reduction method of SIP (Hettich & Kortanek, 1993) with the idea of Blankenship and Falk (1976) in the framework of the direct multiple shooting method.

This article develops an algorithm for locating a feasible point satisfying the KKT conditions to a specified tolerance of semi-infinite-dimensional, inequality-path-constrained dynamic programs (PCDP). Based on the right-hand restriction method proposed in Mitsos (2011) for standard SIP, the algorithm proceeds by iteratively approximating the PCDP by restricting the right-hand side of the path constraint and enforcing it at a finite number of time points. A dynamic optimization problem with *finitely* many constraints is solved to local optimality at each iteration, thereby making it possible to combine it with state-of-the-art local dynamic optimization codes. It will be established that the algorithm terminates finitely with a guaranteed feasible point and a certificate of satisfaction of the first-order KKT conditions of the PCDP to a specified tolerance under the following main assumptions: (i) availability of a nonlinear program (NLP) local solver that generates a KKT point of the constructed approximate PCDP at each iteration if this problem is indeed feasible; (ii) existence of a Slater point of the PCDP that also satisfies the first-order KKT conditions of the PCDP to a specified tolerance; and (iii) KKT multipliers are nonnegative and uniformly bounded with respect to all iterations.

The remaining part of the article is organized as follows. Section 2 states the path-constrained dynamic optimization problems of interest, where for simplicity a single constraint is considered. Section 3 describes the algorithm to locate a feasible approximate KKT point of the path-constrained dynamic optimization problem with guaranteed satisfaction of path constraints, and it also presents a proof of finite convergence of the algorithm. Section 4 illustrates the property of guaranteed satisfaction of path constraints and analyzes the effect of tuning parameters in the algorithm using two numerical case studies. Section 5 presents conclusions and an outlook on future work.

2. Problem statement

We consider semi-infinite-dimensional, inequality-path-constrained dynamic optimization problems of the form:

$$\begin{aligned} \min_{u \in U} \quad & S(x(t_f, u)) \\ \text{s.t.} \quad & g(x(t, u), u) \le 0, \quad \forall t \in T, \\ & \dot{x}(t, u) = f(x(t, u), u), \quad \forall t \in T, \\ & x(t_0, u) = x_0(u), \end{aligned} \tag{PCDP}$$

where $t \in T := [t_0, t_f]$ represents the independent variable, e.g., time; $u \in U$ denote the time-invariant control/decision variables, with $U \subset \mathbb{R}^n$ nonempty and compact; and $x(\cdot, u)$ is the state response to a given control u, with $x(t, u) \in X, \forall (t, u) \in T \times U$ and $X \subset \mathbb{R}^{n_x}$ nonempty and compact. The objective function $S : X \to \mathbb{R}$, path-constraint function $g : X \times U \to \mathbb{R}$, right-hand-side function $f : X \times U \to \mathbb{R}^{n_x}$, and initial-value function $x_0 : U \to \mathbb{R}^{n_x}$ are all assumed to be continuously differentiable in their respective arguments. No convexity assumptions are made, but local solutions are considered.

Remark 1. Optimal control problems with control trajectories as their decision variables can be approximated (restricted) into (PCDP) via the control vector parameterization technique (Biegler, 2010; Lin, Loxton, & Teo, 2014; Loxton, Lin, Rehbock, & Teo, 2012; Teo, Goh, & Wong, 1991). Moreover, problems with an integral term as part of their objective function or with explicit time dependence can be transformed into (PCDP) via the introduction of extra variables and equations in the dynamic system (Chachuat, 2006–2007; Teo et al., 1991).

The main objective of this article is to develop an algorithm to obtain a feasible point satisfying the KKT conditions of (PCDP) to a specified tolerance.

3. Algorithm development and analysis

This section starts by formulating the lower-level problem of (PCDP) and solving it to global optimality to determine the largest violation of a path constraint along the time horizon for a given control, and discusses possible solution strategies. Then, the algorithm to locate a feasible point satisfying the KKT conditions of (PCDP) to a specified tolerance is described and its finite convergence is established.

3.1. Feasibility subproblem

Motivated by checking feasibility of a given point for standard nonconvex SIPs (Mitsos, Lemonidis, Lee, & Barton, 2008), the feasibility of a given control $\bar{u} \in U$ for the dynamic program (PCDP) can be established via globally solving (LLP) below and determining the largest violation of the path constraint over T as:

$$g^{\max}(\bar{u}) := \max_{t\in T} \; g(x(t,\bar{u}),\bar{u}) \quad \text{s.t.} \quad \dot{x}(t,\bar{u}) = f(x(t,\bar{u}),\bar{u}), \quad \forall t \in T, \quad x(t_0,\bar{u}) = x_0(\bar{u}). \tag{LLP}$$

Problem (LLP) can be addressed by integrating the dynamic system with available ODE solvers and then determining the maximal value of $g(x(t,\bar{u}),\bar{u})$ based on the intermediate state values $x(t,\bar{u})$. However, this simple approach requires sufficiently many points in approximating the path constraint and provides no direct error control. A more efficient way of controlling the error level involves formulating and solving a hybrid discrete–continuous dynamic system (Barton & Lee, 2002; Branicky & Mattsson, 1997). Specifically, an extra variable $\gamma(t,\bar{u})$ representing the maximal constraint violation up to t is appended to the dynamic system and state events are defined ensuring that the activations/deactivations of the path constraint function are precisely located:

$$\dot{\gamma}(t,\bar{u}) = \begin{cases} 0, & \text{if } \gamma(t,\bar{u}) \geq g(x(t,\bar{u}),\bar{u}) \\ & \text{or } \dot{g}(x(t,\bar{u}),\bar{u}) < 0, \\ \dot{g}(x(t,\bar{u}),\bar{u}), & \text{otherwise}, \end{cases} \tag{HDS}$$

with $\gamma(t_0,\bar{u}) = g(x_0(\bar{u}),\bar{u})$.

This hybrid system can be solved accurately using adaptive step-size solvers with rigorous state-event location (Park & Barton, 1996), thereby providing the maximum constraint violation as $g^{\max}(\bar{u}) = \gamma(t_f,\bar{u})$. Likewise, the global maximizer $t^{\max}(\bar{u})$ is obtained at the time point at which the last event is triggered during the integration.

3.2. Local optimization algorithm of path-constrained dynamic programs with guaranteed feasibility

The proposed Algorithm 1 (graphically illustrated in Fig. 1) locates a feasible point satisfying the KKT conditions of (PCDP) to a specified tolerance, by solving an approximation of (PCDP) as:

$$\min_{u\in U} \; S(x(t_f,u)) \quad \text{s.t.} \quad g(x(t,u),u) \leq -\epsilon_g^k, \quad \forall t \in T^k, \quad \dot{x}(t,u) = f(x(t,u),u), \quad x(t_0,u) = x_0(u), \tag{APCDP^k}$$

where $\epsilon_g^k > 0$ and $T^k \subset T$ denote the restriction parameter of the path constraint and the (finite) discretized constraint points, respectively, at iteration k. Discretizing the path constraint on a subset of T results in a relaxation of (PCDP), and consequently, (APCDP^k) is neither a restriction nor a relaxation of (PCDP) in general.

Algorithm 1 Local optimization of *feasible* path-constrained dynamic program (PCDP) with guaranteed feasibility.

Input: finite or empty set $T^0 \subset T$; restriction parameter $\epsilon_g^0 > 0$; reduction parameter $r > 1$; tolerances $\epsilon_{\text{act}}, \epsilon_{\text{stat}} > 0$; iteration counter $k = 0$

Repeat:

- Solve (APCDP^k) to local optimality

  If feasible

  - Set u^k equal to locally optimal solution point, and obtain n gradients $\nabla_u g(x(t_i,u^k),u^k)$, $i = 1,2,\ldots,n$, and n multipliers, which consists of *all* (at most n) linearly independent gradients of active constraints at u^k with their respective multipliers, and enough inactive constraint gradients (if needed) with zero multipliers
  - Solve (LLP) to global optimality, and obtain $g^{\max}(u^k)$ and $t^{\max}(u^k)$

    If $g^{\max}(u^k) \leq 0$ (Case II)

    ∗ **If** $\|\nabla_u S(x(t_f,u^k)) + \sum_{i=1}^{n} \lambda_i^k \nabla_u g(x(t_i^k,u^k),u^k)\| \leq \epsilon_{\text{stat}}$ and $\lambda_i^k g(x(t_i^k,u^k),u^k) \in [-\lambda_i^k \epsilon_{\text{act}}, 0]$ **Terminate**

    ∗ **Else** Set $\epsilon_g^{k+1} \leftarrow \epsilon_g^k / r$ and $T^{k+1} \leftarrow T^k$

    Else Set $T^{k+1} \leftarrow T^k \cup t^{\max}(u^k)$ (Case III)

  Else Set $\epsilon_g^{k+1} \leftarrow \epsilon_g^k / r$ and $T^{k+1} \leftarrow T^k$ (Case I)

  Set $k \leftarrow k+1$

Remark 2. Since the state trajectories $x(\cdot,u)$ are assumed to be in the compact set X on T, and since $f : X \times U \to \mathbb{R}^{n_x}$ is assumed to be continuously differentiable, solutions of $x(\cdot,u)$ of the differential equations in (APCDP^k) are guaranteed to exist and be unique on T for each $u \in U$. Moreover, the first-order sensitivity trajectories $\frac{\partial x}{\partial u}$ exist, and are continuous and well-defined on $T \times U$ (Coddington & Levinson, 1955).

We make the following assumption regarding the existence of a Slater point satisfying the first-order KKT conditions of problem (PCDP) to a specified tolerance. Such a relaxation of the exact KKT conditions leads to only approximately optimal solution points, but is necessary for the finite termination of Algorithm 1 as established later on in Theorem 1. The suboptimality can be controlled via adjusting the tolerances.

Assumption 1. For given tolerances $\epsilon_{\text{stat}}, \epsilon_{\text{act}} > 0$, there exist $u^s \in U \subset \mathbb{R}^n$, a positive constant $\epsilon^s \leq \epsilon_{\text{act}}$, a finite set $\{t_1^s,\ldots,t_n^s\} \subset T$, and nonnegative bounded multipliers λ_i^s, $1 \leq i \leq n$, such that

$$g(x(t,u^s),u^s) \leq -\epsilon^s, \quad \forall t \in T, \tag{5}$$

$$\left\| \nabla_u S(x(t_f,u^s)) + \sum_{i=1}^{n} \lambda_i^s \nabla_u g(x(t_i^s,u^s),u^s) \right\| \leq \epsilon_{\text{stat}},$$

$$\lambda_i^s g(x(t_i^s,u^s),u^s) \in [-\lambda_i^s \epsilon_{\text{act}}, 0], \quad i = 1,2,\ldots,n.$$

Remark 3. There are two types of KKT conditions for SIPs, namely infinite-dimensional representation (Cánovas, López, Mordukhovich, & Parra, 2010; Jahn, 2007) and finite-dimensional (recall that n is the number of optimization variables u) representation (Bertsekas, 1999; John, 2014). Under extended Mangasarian–Fromovitz constraint qualification (EMFCQ), John's theorem in John (2014) coincides with the finite-dimensional representation of KKT conditions (Floudas & Stein, 2007; Stein & Steuermann, 2012), which states that for a local minimizer of SIP there exists *at most* n active indices of the minimizer to characterize its KKT (first-order) necessary optimality conditions of SIP. In Assumption 1, we assume by analogy that there exists *at most* n active indices of (PCDP) for u^s, or in other words, that there exists an approximately

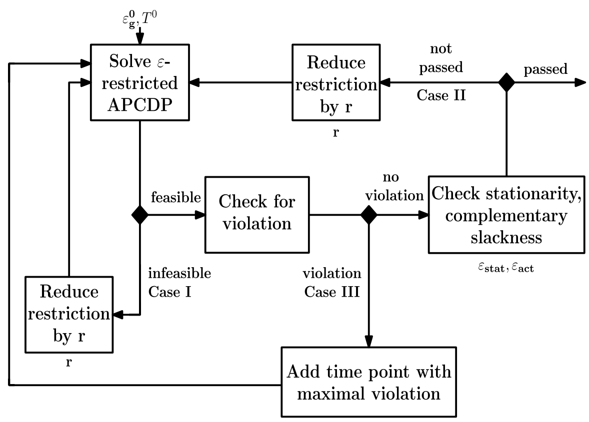


Fig. 1. Graphic illustration of Algorithm 1.

optimal point for which the path constraint is not active over an interval. Moreover, as indicated in Floudas and Stein (2007), in the case that less than n active indices are sufficient for the KKT conditions, one can satisfy the second and third conditions of (5) by artificially listing enough inactive points with zero multipliers.

Remark 4. Note that the EMFCQ assumption in the SIP literature (Bertsekas, 1999; Floudas & Stein, 2007) guarantees that for a point on the boundary of the feasible set of an SIP there exists a feasible direction directly leading to the interior of the feasible set. In Assumption 1 we therefore (implicitly) suppose that (PCDP) is feasible.

A local NLP solver may fail to locate a feasible point, even if feasible points do indeed exist. When a local solver is applied to a feasible NLP with a nonconvex feasible set for instance, it can return an infeasible point. We make the following assumption to rule out this possibility.

Assumption 2. A local NLP solver is available that generates a KKT point of (APCDP^k) at each iteration, whenever (APCDP^k) is feasible.

Assumption 3 below is used to guarantee that all KKT multipliers are uniformly bounded in the convergence proof. Note that in Floudas and Stein (2007) it was assumed that all multipliers belong to a standard simplex.

Assumption 3. All KKT multipliers are nonnegative and uniformly bounded with respect to all iterations.

Now the convergence result can be given as follows.

Theorem 1. *Under Assumptions 1–3, Algorithm 1 terminates finitely and generates a feasible ϵ_{stat}-approximate KKT point of (PCDP) with ϵ_{act}-active indices. This holds for any reduction parameter $r > 1$, any initial restriction parameter $\epsilon_g^0 > 0$, and any finite set $T^0 \subset T$.*

Proof. At each iteration of Algorithm 1, we only need to consider the three possible outcomes Cases I–III (labeled in Algorithm 1 and shown in Fig. 1), as Assumption 2 excludes the possibility that a feasible (APCDP^k) is reported infeasible when it is feasible due to failure of the local solver. Finite convergence is established next by excluding infinite occurrences of these cases. In order to exclude infinite occurrences of Cases I and III we use the proof idea in Mitsos (2011), whereas excluding infinite occurrences of Case II bears similarities with the proof idea in Floudas and Stein (2007).

Exclusion of infinite Case I: At all iterations we have $T^k \subset T$. By Assumption 1 there exists at least one Slater point u^s of (PCDP). This point is also feasible in (APCDP^k) for $\epsilon_g^k < \epsilon^s$ regardless of what T^k is. Therefore, after at most $\lceil \log_r \epsilon_g^0/\epsilon^s \rceil$ updates of the restriction parameter, (APCDP^k) becomes feasible irrespective of T^k and thus we exclude infinite occurrences of Case I.

Exclusion of infinite Case II: Now we show that an infinite occurrence of Case II is impossible. Assume that the algorithm does not terminate finitely. Then, there exists a subsequence $\{u^{k_j}\}$ of the returned solution sequence of (APCDP^{k_j}) such that each u^{k_j} is a KKT point for (APCDP^{k_j}) but the termination criterion is not satisfied for any $k_j \geq 0$. Since u^{k_j} is a KKT point for (APCDP^{k_j}), for each k_j we have

$$g(x(t, u^{k_j}), u^{k_j}) \leq -\epsilon_g^{k_j}, \quad \forall t \in T^{k_j}, \tag{6}$$
$$\left\| \nabla_u S(x(t_f, u^{k_j})) + \sum_{i=1}^{n} \lambda_i^{k_j} \nabla_u g(x(t_i^{k_j}, u^{k_j}), u^{k_j}) \right\| = 0,$$
$$\lambda_i^{k_j}[g(x(t_i^{k_j}, u^{k_j}), u^{k_j}) + \epsilon_g^{k_j}] = 0, \quad i = 1, 2, \ldots, n.$$

According to Assumption 3, $(\lambda_1^{k_j}, \lambda_2^{k_j}, \ldots, \lambda_n^{k_j})$ belong to a compact set, say Λ, and the sequence $(u^{k_j}, \lambda_1^{k_j}, \lambda_2^{k_j}, \ldots, \lambda_n^{k_j}, t_1^{k_j}, t_2^{k_j}, \ldots, t_n^{k_j})$ is contained in the compact set $U \times \Lambda \times (T)^n$, thus it possesses an accumulation point $(u^*, \lambda_1^*, \lambda_2^*, \ldots, \lambda_n^*, t_1^*, t_2^*, \ldots, t_n^*)$ in the same compact set. Thus, by continuity the second equation of (6) gives

$$\left\| \nabla_u S(x(t_f, u^*)) + \sum_{i=1}^{n} \lambda_i^* \nabla_u g(x(t_i^*, u^*), u^*) \right\| = 0. \tag{7}$$

From the assumption that there are at most n active indices (see Remark 3), we have

$$\left\| \nabla_u S(x(t_f, u^{\bar{k}_j})) + \sum_{i=1}^{n} \lambda_i^{\bar{k}_j} \nabla_u g(x(t_i^{\bar{k}_j}, u^{\bar{k}_j}), u^{\bar{k}_j}) \right\| \leq \epsilon_{stat} \tag{8}$$

for some $\bar{k}_j \in \mathbb{N}$.

After at most $N_r = \max\{\bar{k}_j, \lceil \log_r(\epsilon_g^0/\epsilon^s) \rceil\}$ updates of ϵ_g, the current reduction parameter satisfies $\epsilon_g^{N_r} \leq \epsilon^s$. Thus the third equation of (6) gives

$$\lambda_i^{N_r}[g(x(t_i^{N_r}, u^{N_r}), u^{N_r}) + \epsilon_g^{N_r}] = 0, \quad i = 1, 2, \ldots, n. \tag{9}$$

From $\epsilon_g^{N_r} \leq \epsilon^s$, $\epsilon^s \leq \epsilon_{act}$ in Assumption 1 and (9),
$\lambda_i^{N_r} g(x(t_i^{N_r}, u^{N_r}), u^{N_r}) \geq -\lambda_i^{N_r}\epsilon^s$, $i = 1, 2, \ldots, n$, and therefore

$$-\lambda_i^{N_r} g(x(t_i^{N_r}, u^{N_r}), u^{N_r}) \in [-\lambda_i^{N_r}\epsilon_{act}, 0], \quad i = 1, 2, \ldots, n. \tag{10}$$

Therefore, from (8) and (10), we showed that after at most N_r iterations the candidate point u^{N_r} is located such that the termination criteria (8) and (10) are satisfied, which contradicts the assumption that the termination criterion is not satisfied for any $k_j \in \mathbb{N}$. If u^{N_r} is (PCDP)-feasible, the desired result holds. Otherwise the restriction parameter is no longer updated. Therefore, $\epsilon_g^k \geq \epsilon_g^{\min} := \epsilon_g^0/r^{N_r}$ for all iterations.

Exclusion of infinite Case III: Finally, we show that an infinite sequence of (PCDP)-infeasible points generated by (APCDP^k) is impossible. Note that $\epsilon_g^k \geq \epsilon_g^{\min} > 0$ holds for all iterations. We first consider a sequence of solutions to (APCDP^k). Since U is compact, we can select a converging subsequence $\{u^m\}$ with the limit point $\hat{u}$. Consider the corresponding solutions $t^m := t^{\max}(u^m)$ of (LLP). By construction of (APCDP^l), we have $g(x(t^m, u^l), u^l) \leq -\epsilon_g^{\min} < 0$, $\forall l, m \in \mathbb{N}$ with $l > m$. By continuity of g and compactness of $X \times U$, we know that g is uniformly continuous on $X \times U$. Because $x(t, u) \in X, \forall (t, u) \in T \times U$, g is essentially uniformly continuous on $T \times U$. For convenience, define $\tilde{g}(t, u) := g(x(t, u), u)$. Thus, for all $\bar{\epsilon} > 0$ there exists a $\bar{\delta} > 0$ (independent of any $t \in T$ and any $u \in U$ due to the uniform continuity of

$\tilde{g}(t, u)$ *jointly* on $T \times U$) such that for all u and the u^l that satisfies $|u - u^l| < \bar{\delta}$, we have $|\tilde{g}(t^m, u) - \tilde{g}(t^m, u^l)| < \bar{\epsilon}$, for all $l, m \in \mathbb{N}$ with $l > m$. It follows that $\tilde{g}(t^m, u) < \bar{\epsilon} + \tilde{g}(t^m, u^l)$. Taking $\bar{\epsilon} = \frac{\epsilon_g^{\min}}{2} > 0$ and noting that $\tilde{g}(t^m, u^l) \leq -\epsilon_g^{\min} < 0$, we have $\tilde{g}(t^m, u) < -\frac{\epsilon_g^{\min}}{2} < 0$. Since $u^m \to \hat{u}$, for any $\bar{\delta}$ there exists K such that $|u^m - u^l| < \bar{\delta}$, for all $l, m \in \mathbb{N}$ with $l > m > K$. Then $g(x(t^m, u^m), u^m) < -\frac{\epsilon_g^{\min}}{2} < 0$. Therefore, after a finite K, the points given by (APCDP^l) in this case are (PCDP)-feasible. □

Remark 5. If (PCDP) does not have a Slater point, the approximate problems (APCDP^k) cannot generate feasible points of (PCDP), and thus Algorithm 1 loops infinitely. If (PCDP) has Slater points which are not locally approximate KKT points, the approximate problems (APCDP^k) can generate feasible points of (PCDP), but the algorithm may not terminate finitely either, since Algorithm 1 is designed not to terminate just for feasible points. Obviously, the infinite iterations can be avoided by introducing a maximal number of iterations, albeit with the consequence that with that change feasibility and optimality cannot be guaranteed.

Remark 6. In this article, for simplicity it is assumed that local solvers return exact KKT points. In practice the solvers return approximate KKT points, i.e., points satisfying the KKT conditions within some tolerances. It is relatively easy to adjust the proposed algorithm to take into account these approximate points. Roughly speaking the tolerances used for the local solvers should be substantially tighter than the tolerances used in the algorithm.

4. Numerical case studies

This section illustrates and verifies the distinguishing property of guaranteed satisfaction of path constraints, and analyzes the effects of the tuning parameters in Algorithm 1 using two small numerical case studies. A thorough numerical experiment for many complicated and large-scale systems and revealing the practical features of our algorithm computationally is beyond the scope of this article. The implementation is carried out in MATLAB Version 7.13.0.564 (R2011b, win32), and runs on a Intel Xeon E5-2630 v2 @ 2.60 GHz, 128 GB terminal server operating Windows 2008 R2 Datacenter. The dynamic optimization is carried out using Dyos HoneyBee 1.9 with single shooting and SNOPT as optimizer (Schlegel, 2005). The sensitivity information is obtained from SLIMEX via the integration of the sensitivity equations (Cao, Li, Petzold, & Serban, 2003; Chachuat, 2006–2007; Schlegel, Marquardt, Ehrig, & Nowak, 2004). The lower-level program is solved by integration over a fine grid. The problems selected are the fed-batch penicillin fermentation process in Visser, Srinivasan, Palanki, and Bonvin (2000) and the Van der Pol (VDP) oscillator in Chen and Vassiliadis (2005) and Gritsis (1990). The formulations for both problems are reported in the Appendix.

We start by verifying the property of guaranteed satisfaction of path constraints of Algorithm 1 for both problems. Then, based on the implementation results of the two examples, we analyze the effect of the restriction parameter ϵ_g^0, the control parameter r, and the initial set T^0 in terms of iterations.

4.1. Verification of guaranteed feasibility

For both examples, we use the control vector parameterization technique with piecewise-constant basis functions. For the fed-batch penicillin fermentation process, we consider 40 equidistant intervals resulting in 40 decision variables and $T^0 = \{0, 2, 4, 6, \ldots, 40\}$. For the Van der Pol oscillator, we consider 100 equidistant intervals resulting in 100 decision variables and $T^0 = \{5\}$. The results obtained by running the case study problems with the parameters $\epsilon_{\text{stat}} = 0.001$, $\epsilon_{\text{act}} = 0.001$, $\epsilon_g^0 = 0.05$ and $r = 4$ are reported in Table 1. These results confirm that the path constraint is rigorously satisfied upon termination of Algorithm 1. For the fed-batch penicillin fermentation process, the algorithm terminates after 48 iterations with the parameter $\epsilon_g^{48} = 7.81 \times 10^{-4}$. Fig. 2 shows the corresponding path constraint profile upon termination under the optimized control, where the red star indicates that guaranteed feasibility is achieved over the entire time horizon [0, 40], and that this point is the point at which the maximal value of the path constraint function occurs over the entire time horizon at last iteration. Fig. 3 gives a zoomed-in Fig. 2 around this point. Fig. 4 shows the optimal control profile of the fed-batch penicillin fermentation process. For the VDP oscillator, similar results are shown in Figs. 5–7.

Table 1
Summary table for both numerical case studies, where "iter" stands for iterations.

Problem	Opt-cost	Iter	$\|T^{final}\|$	ϵ_g^{final}	$g^{\max}$
Penicillin	−0.79	48	65	7.81×10^{-4}	-4.12×10^{-4}
VDP	2.96	18	15	7.81×10^{-4}	-3.53×10^{-4}

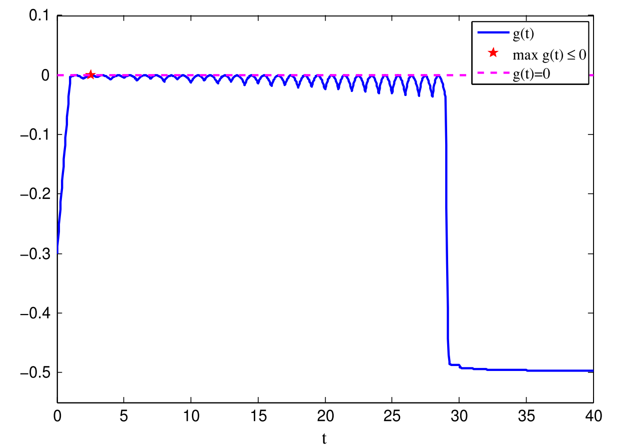


Fig. 2. Path constraint profile upon termination of Algorithm 1 for the fed-batch penicillin fermentation process. The red star indicates rigorous satisfaction of the path constraint upon termination over the entire time horizon [0, 40], and the point at which the maximal value of the path constraint function occurs over the entire time horizon. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

From Fig. 2 and its zoomed-in Fig. 3 around the maximum value of the path constraint, and Fig. 5 and its zoomed-in Fig. 6 likewise, for both cases we can see that the time point corresponding to $g^{\max}$ (i.e., $t = 2.5626$ in Fig. 3, and $t = 0.85$ in Fig. 6) is not one of the points the constraint was enforced on (i.e., $t = 2, 2.7227$ in Fig. 3, and $t = 0.8158, 0.9209$ in Fig. 6), and that the values of the path constraint at two consecutive enforced points are less than those values of the path constraint in-between. Moreover, it is observed that the constraint is not binding, suggesting that the restriction parameter is too large for the discretization set. However, the inactive point satisfies the stationarity within the prespecified tolerance. The above observations agree to what we expect from the discretization technique of using finite constraints to approximate infinite ones.

4.2. Effect of tuning parameters ϵ_g^0, r and T^0

Similar to Mitsos (2011), the restriction parameter ϵ_g^0 can have a large influence on the computational performance of Algorithm 1. Too large a value for ϵ_g^0 can make (APCDP^k) become infeasible or generate significantly suboptimal points. With too small a value for

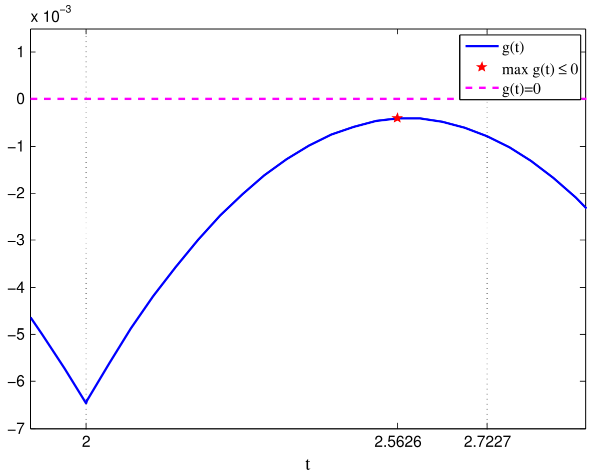


Fig. 3. Zoomed-in Fig. 2 around the maximum value of the path constraint upon termination for the fed-batch penicillin fermentation process. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

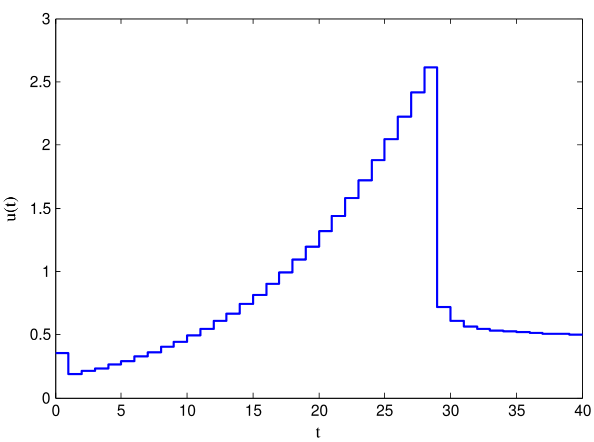


Fig. 4. Control input profile upon termination of Algorithm 1 for the fed-batch penicillin fermentation process.

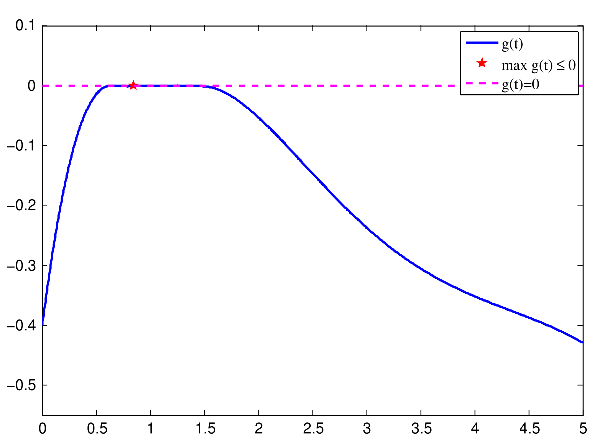


Fig. 5. Path constraint profile upon termination of Algorithm 1 for the Van der Pol oscillator. The red star indicates rigorous satisfaction of the path constraint upon termination over the entire time horizon [0, 5], and the point at which the maximal value of the path constraint function occurs over the entire time horizon. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

ϵ_g^0 on the other hand, (APCDP^k) yields an outer-approximation of (PCDP) until T^k closely approximates T, thereby resulting in many

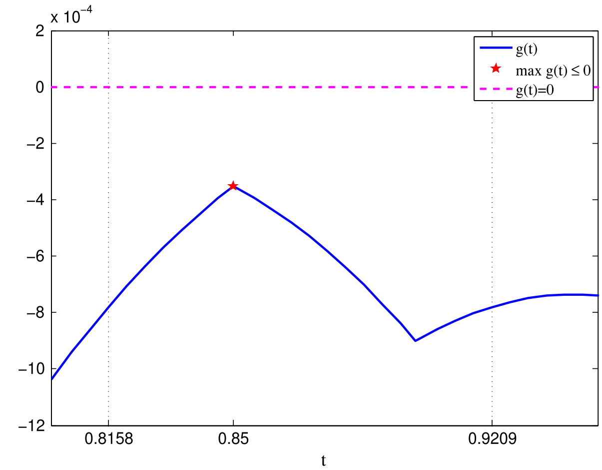


Fig. 6. Zoomed-in Fig. 5 around the maximum value of the path constraint upon termination for the Van der Pol oscillator. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

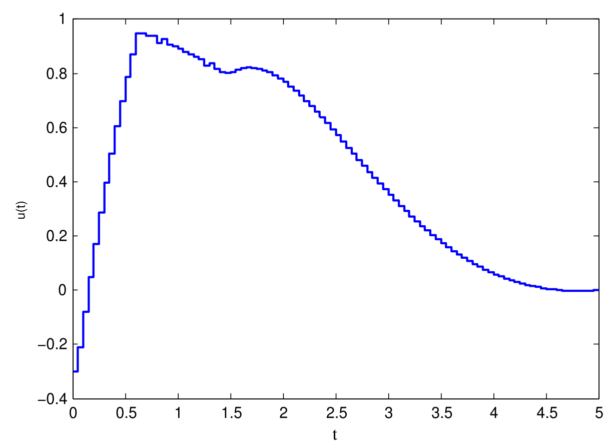


Fig. 7. Control input profile upon termination of Algorithm 1 for the Van der Pol oscillator [0, 5].

computationally demanding iterations due to a large number of interior-point constraints. The effect of the reduction parameter r on the computational performance is shown in Fig. 8 for both case studies. Too slow a decrease of ϵ_g (small value of r) results in a large number of iterations. The number of iterations are found to be at a minimum for r in the range between 3 and 10 here, and the performance appears to be rather insensitive in this range. Note that these observed trends are similar to those reported in Mitsos (2011).

For a fixed restriction parameter ϵ_g^0 and a fixed r, the number of iterations varies with the cardinality of T^0, an effect not studied in Mitsos (2011). A too small cardinality usually results in a large number of iterations due to the need for populating enough cuts before feasible points can be generated. On the other hand, the cost per iteration is initially very large even for the first iterations when the initial cardinality of T^0 is itself large due to the presence of many interior point constraints in the (APCDP^k). In Fig. 9 for both case studies, the effect of initial cardinality to the number of iterations is shown.

5. Conclusions and future work

Based on the global SIP algorithm proposed in Mitsos (2011), an algorithm is proposed for local optimization of path-constrained dynamic optimization problem with guaranteed constraint satisfaction, a distinguishing feature compared to existing algorithms in

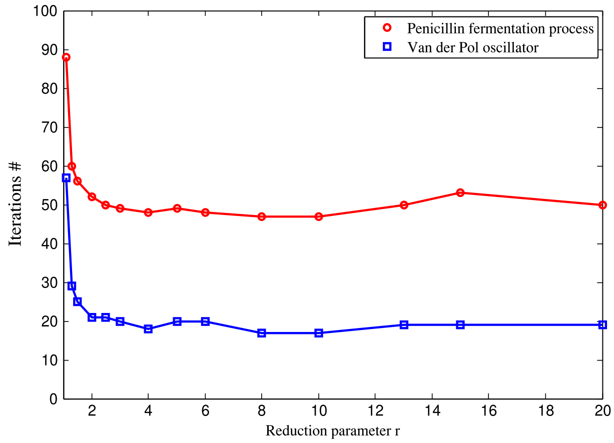


Fig. 8. Iterations vs. reduction parameter r for the fed-batch penicillin fermentation process in Visser et al. (2000) and the Van der Pol oscillator in Chen and Vassiliadis (2005) and Gritsis (1990), and for the fed-batch penicillin fermentation process $T^0 = \{0, 2, 4, 6, \ldots, 40\}$, and for the Van der Pol oscillator $T^0 = \{5\}$.

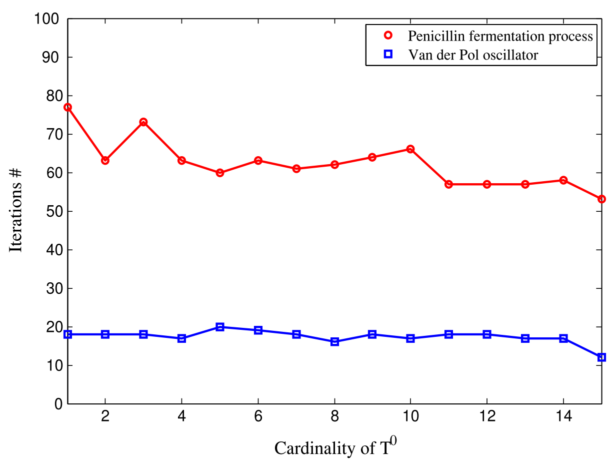


Fig. 9. Iterations vs. cardinality of T^0 for the fed-batch penicillin fermentation process in Visser et al. (2000) and the Van der Pol oscillator in Chen and Vassiliadis (2005) and Gritsis (1990). Values of $r = 4$ and $\epsilon_g^0 = 0.05$ are used in both cases.

the literature. It is shown that this algorithm converges to a feasible point satisfying the KKT conditions of the problem to a specified tolerance subject to mild assumptions. Another interesting feature of this algorithm lies in its simplicity as the dynamic optimization subproblems can be solved to local optimality with state-of-the-art solvers and the constraint violation subproblems can be addressed efficiently using state-of-the-art hybrid discrete–continuous numerical integrators that feature rigorous event detection.

Direct extension may lie in the following aspects. In the current algorithm the time point at which the largest violation of the path constraint occurs is added to the existing T^k. We could of course add more or even all local maxima of (LLP) to T^k at each iteration. The compromise here is between increasing the number of iterations and the computational burden per iteration. It is also easy to consider multiple path constraints, which is typical for applications. There is also no problem with handling dynamic optimization problems with additional point constraints and/or integral objective. Other global optimization algorithms for SIP in the literature can be integrated with our algorithm, for example, using the methods in Tsoukalas and Rustem (2011). In our algorithm we solve the subproblems (APCDP^k) locally despite their possible nonconvex feasible sets and/or objective functions. To prove convergence, we assume that the local solver used can nevertheless return a KKT point. However, in the case of nonconvex feasible sets, local solvers may fail to return feasible points. Thus, it would be desirable to weaken the assumption.

The local solution method proposed herein could also be adapted back to local solution of standard SIPs, where local optimization of discretized NLP could be solved by using available local NLP solvers (e.g., SNOPT Gill, Murray, & Saunders, 2002), and the global optimization of (LLP) could be solved by existing global NLP solvers such as BARON. The local solution method also could be extended to deal with path-constrained dynamic optimization embedded with differential algebraic equations (Peter et al., 2010). Finally, it would be interesting to also consider the adaptation of Mitsos (2011) to global solution of path-constrained dynamic optimization. This would require use of global dynamic optimization techniques (Chachuat, Singer, & Barton, 2006; Singer & Barton, 2006).

Throughout the article a finite number of decision variables has been assumed. It definitely deserves further studies on extending the proposed algorithm to the infinite dimensional path-constrained dynamic programs (i.e., without applying the control vector parameterization).

Acknowledgments

We are thankful to Dr. Hannemann-Tamás at RWTH Aachen University, Prof. Xiang Li at Queen's University and Dr. Yang Chen at the National Energy Technology Laboratory and Carnegie Mellon University for helpful discussions. Last but not the least, we are grateful to the anonymous reviewers whose comments lead to substantial improvement of this article, especially Reviewer 5.

Appendix. Case study problems

The problem formulations are given below, prior to applying the control vector parameterization.

A.1. Penicillin fed-batch fermentation

We consider a fixed final time formulation of the Penicillin example in Visser et al. (2000):

$$\begin{aligned}
&\min_{u(t)} -x_3(t_f)\\
&\text{s.t. } \dot{x}_1(t) = \frac{\mu x_1(t)x_2(t)}{K_l x_1(t) + x_2(t)} - \frac{u(t)x_1(t)}{x_4(t)},\\
&\quad \dot{x}_2(t) = \frac{-\mu x_1(t)x_2(t)}{(K_l x_1(t) + x_2(t))Y_{xs}} - M_x x_1(t),\\
&\quad - \frac{\theta_m x_1(t)x_2(t)}{Y_p(x_2(t) + K_p + x_2(t)^2/K_i)} + u(t)\frac{S_0 - x_2(t)}{x_4(t)},\\
&\quad \dot{x}_3(t) = \frac{\theta_m x_1(t)x_2(t)}{x_2(t) + K_p + x_2(t)^2/K_i} - K_{xp}x_3(t) - u(t)\frac{x_3(t)}{x_4(t)},\\
&\quad \dot{x}_4(t) = u(t),\\
&\quad x(0) = [1, 0.2, 0.001, 250],\\
&\quad x_2(t) \le -0.5, \quad \forall t,\\
&\quad 0 \le u(t) \le 10, \quad \forall t,
\end{aligned}$$

with $t_f = 40$, $S_0 = 400$ and the other parameter values are listed in Table A.1. The path constraint is $g(t) = x_2(t) - 0.5 \le 0$.

A.2. Van der Pol oscillator

We consider a fixed final time formulation of the Van der Pol oscillator as given in Chen and Vassiliadis (2005) and

Table A.1
Parameter values.

K_l	μ	Y_{xs}	θ_m	Y_p	K_i	M_x	K_{xp}	K_p
6e−3	0.11	0.47	4e−3	1.2	0.1	0.029	0.01	1e−4

Gritsis (1990):

$$\min_{u(t)} x_3(t_f)$$
$$\text{s.t.}\ \dot{x}_1(t) = (1 - x_2(t)^2)x_1(t) - x_2(t) + u(t),$$
$$\dot{x}_2(t) = x_1(t),$$
$$\dot{x}_3(t) = x_1(t)^2 + x_2(t)^2 + u(t)^2,$$
$$x(0) = [0, 1, 0],$$
$$x_1(t) \le -0.4, \quad \forall t,$$
$$-0.3 \le u(t) \le 1, \quad \forall t,$$

with $t_f = 5$. The path constraint is $g(t) = x_1(t) + 0.4 \le 0$.

References

Bainum, P. M., & Kumar, V. K. (1980). Optimal control of the shuttle-tethered-subsatellite system. *Acta Astronautica*, *7*(12), 1333–1348.

Barton, P. I., & Lee, C. K. (2002). Modeling, simulation, sensitivity analysis, and optimization of hybrid systems. *ACM Transactions on Modeling and Computer Simulation*, *12*(4), 256–289.

Bertsekas, D. P. (1999). *Nonlinear programming* (2nd ed.). Athena Scientific.

Betts, J. T., & Huffman, W. P. (1992). Application of sparse nonlinear programming to trajectory optimization. *Journal of Guidance, Control & Dynamics*, *15*(1), 198–206.

Biegler, L. T. (2007). An overview of simultaneous strategies for dynamic optimization. *Chemical Engineering & Processing: Process Intensification*, *46*(11), 1043–1053.

Biegler, L. T. (2010). *MOS-SIAM series on optimization*, *Nonlinear programming: concepts, algorithms, and applications to chemical processes*. SIAM-Society for Industrial and Applied Mathematics.

Blankenship, J. W., & Falk, J. E. (1976). Infinitely constrained optimization problems. *Journal of Optimization Theory & Applications*, *19*, 261–281.

Bock, H. G., & Plitt, K. J. (1984). A multiple shooting algorithm for direct solution of optimal control problems. In *Proceedings of IFAC 9th world congress, Budapest, Hungary* (pp. 242–247).

Branicky, M. S., & Mattsson, S. E. (1997). Simulation of hybrid systems. In P. Antsaklis, et al. (Eds.), *Lecture notes in computer science*: *Vol. 1273*. *Hybrid systems IV* (pp. 31–45). Berlin: Springer-Verlag.

Cánovas, M. J., López, M. A., Mordukhovich, B. S., & Parra, J. (2010). Variational analysis in semi-infinite and infinite programming, II: Necessary optimality conditions. *SIAM Journal on Optimization*, *20*(6), 2788–2806.

Cao, Y., Li, S., Petzold, L., & Serban, R. (2003). Adjoint sensitivity analysis for differential–algebraic equations: The adjoint DAE system and its numerical solution. *SIAM Journal on Scientific Computing*, *24*(3), 1076–1089.

Chachuat, B. (2006–2007). *Nonlinear and dynamic optimization: from theory to practice. Lecture notes-IC-32: Winter semester*. EPFL.

Chachuat, B., Singer, A. B., & Barton, P. I. (2006). Global methods for dynamic optimization and mixed-integer dynamic optimization. *Industrial and Engineering Chemistry Research*, *45*(25), 8373–8392.

Chen, T. W. C., & Vassiliadis, V. S. (2005). Inequality path constraints in optimal control: A finite iteration ϵ-convergent scheme based on pointwise discretization. *Journal of Process Control*, *15*(3), 353–362.

Coddington, A., & Levinson, N. (1955). *Theory of ordinary differential equations*. Tata McGraw-Hill Education.

Dai, R., & Cochran, J. E., Jr. (2009). Wavelet collocation method for optimal control problems. *Journal of Optimization Theory & Applications*, *143*, 265–278.

Feehery, W. F., & Barton, P. I. (1998). Dynamic optimization with state variable path constraints. *Computers & Chemical Engineering*, *22*(9), 1241–1256.

Fikar, M. (2001). *On inequality path constraints in dynamic optimisation. Technical report, Technical report mf0102*. Nancy, France: Laboratoire des Sciences du Génie Chimique, CNRS.

Floudas, C. A., Pardalos, P. M., Adjiman, C., Esposito, W. R., Gümüs, Z. H., Harding, S. T., et al. (1999). *Handbook of test problems in local and global optimization*. Kluwer Academic Publishers.

Floudas, C., & Stein, O. (2007). The adaptive convexification algorithm: A feasible point method for semi-infinite programming. *SIAM Journal on Optimization*, *18*(4), 1187–1208.

Gill, P. E., Murray, W., & Saunders, M. A. (2002). SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, *12*(4), 979–1006.

Goh, C. J., & Teo, K. L. (1988). Control parametrization: A unified approach to optimal control problems with general constraints. *Automatica*, *24*(1), 3–18.

Gritsis, D. (1990). *The dynamic simulation and optimal control of systems described by index two differential–algebraic equations*. (PhD thesis), Imperial College London (University of London).

Hannemann-Tamás, R., & Marquardt, W. (2012). How to verify optimal controls computed by direct shooting methods?–A tutorial. *Journal of Process Control*, *22*(2), 494–507.

Hettich, R., & Kortanek, K. O. (1993). Semi-infinite programming: Theory, methods, and applications. *SIAM Review*, *35*(3), 380–429.

Hussein, I. I., & Bloch, A. M. (2008). Optimal control of underactuated nonholonomic mechanical systems. *IEEE Transactions on Automatic Control*, *53*(3), 668–682.

Jacobson, D. H., & Lele, M. M. (1969). A transformation technique for optimal control problems with a state variable inequality constraint. *IEEE Transactions on Automatic Control*, *14*, 457–464.

Jahn, J. (2007). *Introduction to the theory of nonlinear optimization*. Springer.

John, F. (2014). Extremum problems with inequalities as subsidiary conditions. In *Traces and emergence of nonlinear programming* (pp. 197–215). Springer.

Li, B., Yu, C., Teo, K. L., & Duan, G. (2011). An exact penalty function method for continuous inequality constrained optimal control problem. *Journal of Optimization Theory & Applications*, *151*, 260–291.

Lin, Q., Loxton, R., & Teo, K. L. (2014). The control parameterization method for nonlinear optimal control: A survey. *Journal of Industrial and Management Optimization*, *10*(1), 275–309.

Loxton, R., Lin, Q., Rehbock, V., & Teo, K. L. (2012). Control parameterization for optimal control problems with continuous inequality constraints: New convergence results. *Numerical Algebra, Control and Optimization*, *2*(3), 571–599.

Loxton, R. C., Teo, K. L., Rehbock, V., & Yiu, K. F. C. (2009). Optimal control problems with a continuous inequality constraint on the state and the control. *Automatica*, *45*(10), 2250–2257.

Mitsos, A. (2011). Global optimization of semi-infinite programs via restriction of the right-hand side. *Optimization*, *60*(10–11), 1291–1308.

Mitsos, A., Lemonidis, P., Lee, C. K., & Barton, P. I. (2008). Relaxation-based bounds for semi-infinite programs. *SIAM Journal on Optimization*, *19*(1), 77–113.

Mitsos, A., & Tsoukalas, A. (2015). Global optimization of generalized semi-infinite programs via restriction of the right hand side. *Journal of Global Optimization*, *61*(1), 1–17.

Parida, N. C., & Raha, S. (2009). The α-method direct transcription in path constrained dynamic optimization. *SIAM Journal on Scientific Computing*, *31*, 2386–2417.

Park, T. S., & Barton, P. I. (1996). State event location in differential–algebraic models. *ACM Transactions on Modeling and Computer Simulation*, *6*(2), 137–165.

Peter, J., Parida, N. C., & Raha, S. (2010). The α method for solving differential algebraic inequality (DAI) systems. *International Journal of Numerical Analysis and Modeling*, *7*(2), 240–260.

Polak, E. (1987). On the mathematical foundations of nondifferentiable optimization in engineering design. *SIAM Review*, *29*(1), 21–89.

Potschka, A., Bock, H. G., & Schlöder, J. P. (2009). A minima tracking variant of semi-infinite programming for the treatment of path constraints within direct solution of optimal control problems. *Optimization Methods & Software*, *24*(2), 237–252.

Sachs, E. W. (1998). Semi-infinite programming in control. In *Semi-infinite Programming* (pp. 389–411). Springer.

Schlegel, M. (2005). *Adaptive discretization methods for the efficient solution of dynamic optimization problems*. VDI-Verlag.

Schlegel, M., Marquardt, W., Ehrig, R., & Nowak, U. (2004). Sensitivity analysis of linearly-implicit differential–algebraic systems by one-step extrapolation. *Applied Numerical Mathematics*, *48*(1), 83–102.

Schlegel, M., Stockmann, K., Binder, T., & Marquardt, W. (2005). Dynamic optimization using adaptive control vector parameterization. *Computers & Chemical Engineering*, *29*(8), 1731–1751.

Shin, K. G., & McKay, N. D. (1986). A dynamic programming approach to trajectory planning of robotic manipulators. *IEEE Transactions on Automatic Control*, *31*(6), 491–500.

Singer, A. B., & Barton, P. I. (2006). Global optimization with nonlinear ordinary differential equations. *Journal of Global Optimization*, *34*(2), 159–190.

Srinivasan, B., Palanki, S., & Bonvin, D. (2003). Dynamic optimization of batch processes. I. Characterization of the nominal solution. *Computers & Chemical Engineering*, *27*(1), 1–26.

Stein, O., & Steuermann, P. (2012). The adaptive convexification algorithm for semi-infinite programming with arbitrary index sets. *Mathematical Programming*, *136*(1), 183–207.

Teo, K. L., Goh, C. J., & Wong, K. H. (1991). *A unified computational approach to optimal control problems*. John Wiley & Sons.

Teo, K. L., Rehbock, V., & Jennings, L. S. (1993). A new computational algorithm for functional inequality constrained optimization problems. *Automatica*, *29*(3), 789–792.

Tsang, T. H., Himmelblau, D. M., & Edgar, T. F. (1975). Optimal control via collocation and nonlinear programming. *International Journal of Control*, *21*, 763–768.

Tsoukalas, A., & Rustem, B. (2011). A feasible point adaptation of the Blankenship and Falk algorithm for semi-infinite programming. *Optimization Letters*, *5*(4), 705–716.

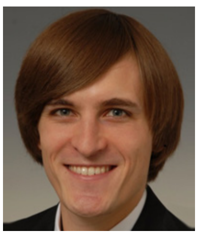
Vassiliadis, V. S., Sargent, R. W. H., & Pantelides, C. C. (1994). Solution of a class of multistage dynamic optimization problems. 2. Problems with path constraints. *Industrial and Engineering Chemistry Research*, *33*(9), 2123–2133.

Visser, E., Srinivasan, B., Palanki, S., & Bonvin, D. (2000). A feedback-based implementation scheme for batch process optimization. *Journal of Process Control*, *10*(5), 399–410.

White, V., Perkins, J. D., & Espie, D. M. (1996). Switchability analysis. *Computers & Chemical Engineering*, *20*(4), 469–474.

Zhao, Y., & Stadtherr, M. A. (2011). Rigorous global optimization for dynamic systems subject to inequality path constraints. *Industrial and Engineering Chemistry Research*, *50*(22), 12678–12693.

Jun Fu received the Ph.D. degree in mechanical engineering from Concordia University, Montreal, Quebec, Canada, in 2009. He started his postdoctoral research in the Department of Mechanical Engineering, Massachusetts Institute of Technology (MIT), USA in 2010. His current research is on optimization algorithms, dynamic optimization, switched systems, control of hysteretic systems, and robust control of nonlinear systems. He received the Chinese Government Award for Outstanding PhD Student Abroad. He has actively participated in a considerable number of conference organizations as session chair including the 2011 American Control Conference. He was a founding member of MIT Postdoctoral Association (PDA), and was Chair of Information Flow & IT Committee of MIT PDA. He is a senior member of IEEE.



Johannes M.M. Faust was born in Füssen, Germany, in 1989. He studied mechanical engineering at RWTH Aachen, Germany and spent the academic year 2011/2012 abroad at Carnegie Mellon University, Pittsburgh, USA majoring in chemical engineering. In 2013 he obtained his B.Sc. and in 2015 he received his M.Sc. in automation sciences, both from RWTH Aachen. Shortly after, he joined the Process Systems Engineering group at RWTH Aachen University, Germany, as a doctoral researcher. His current research interests include optimal control, and its applications to emulsion polymerization.

Benoît Chachuat is a Reader in the Department of Chemical Engineering at the Imperial College London and a member of the Centre of Process Systems Engineering (CPSE). He received his Ph.D. in Chemical Engineering in 2001 from INPL, France, and he held post-doctoral positions at INRIA Sophia-Antipolis (2002–2003), MIT (2003–2005), and EPFL (2005–2008). Prior to joining the Imperial College in 2010, he was an Assistant Professor at McMaster University. His primary research focus is the development of new methods and tools for optimization and control of complex process systems, with emphasis on set-theoretic approach, complete search optimization and optimization-based process control. Current application areas in his group are on microalgae culture systems, resource recovery from wastewater, and membrane cascades.

Alexander Mitsos is a Full Professor (W3) in RWTH Aachen University, and the Director of the Laboratory for Process Systems Engineering (AVT.SVT), comprising 40 research and administrative staff. He received his Dipl.-Ing. from University of Karlsruhe in 1999 and his Ph.D. from MIT in 2006, both in Chemical Engineering. Prior appointments include military service, freelance engineering, involvement in a start-up company, a junior research group leader position at the Aachen Institute of Computational Engineering Science and the Rockwell International Assistant Professorship at MIT. He has numerous publications in peer-reviewed journals and has received several awards. His research focuses on optimization of energy and chemical systems and development of enabling numerical algorithms.