Automatica 78 (2017) 72–78

Contents lists available at ScienceDirect

Automatica

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

Brief paper

Method of evolving junctions: A new approach to optimal control with constraints☆

Wuchen Li[a], Jun Lu[b], Haomin Zhou[a], Shui-Nee Chow[a]

[a] School of Mathematics, Georgia Institute of Technology, United States

[b] Wells Fargo Securities, United States

ARTICLE INFO

Article history:
Received 22 November 2015
Received in revised form
8 November 2016
Accepted 23 November 2016

Keywords:
Optimal control
Constraints
Stochastic differential equations
Junctions

ABSTRACT

We propose a new strategy, called method of evolving junctions (MEJ), to compute the solutions for a class of optimal control problems with constraints on both state and control variables. Our main idea is that by leveraging the geometric structures of the optimal solutions, we recast the infinite dimensional optimal control problem into an optimization problem depending on a finite number of points, called junctions. Then, using a modified gradient flow method, whose dimension can change dynamically, we find local solutions for the optimal control problem. We also employ intermittent diffusion, a global optimization method based on stochastic differential equations, to obtain the global optimal solution. We demonstrate, via a numerical example, that MEJ can effectively solve path planning problems in dynamical environments.

1. Introduction

An optimal control problem with constraints seeks to determine the input (control) $u(t) \in \mathbb{R}^r$ to a dynamical system that optimizes a given performance functional (maximize profit, minimize cost, etc.), while satisfying various state and/or control constraints,

$$\min_{x,u} \int_{t_0}^{t_f} L(x(t), u(t), t)dt + \psi(x(t_f), t_f), \quad (1)$$

subject to

$$\dot{x} = f(x(t), u(t), t), \quad t \in [t_0, t_f],$$
$$x(t_0) = x_0, \quad M(t_f, x(t_f)) = 0,$$
$$\phi(x(t), t) \geq 0, \qquad \varphi(u(t), t) \geq 0, \quad t \in [t_0, t_f],$$

where the state variable $x(t) \in \mathbb{R}^n$ is often called the trajectory or path in the phase space. $u(t) \in \mathbb{R}^r$ is the control variable; $L: \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{R}^+ \to \mathbb{R}$ is the running cost; $\psi : \mathbb{R}^n \times \mathbb{R}^+ \to \mathbb{R}$ is the terminal cost, and t_f is the terminal time, which may be undetermined in some problems. $\phi : \mathbb{R}^n \times \mathbb{R}^+ \to \mathbb{R}^p$ is the state constraint and $\varphi : \mathbb{R}^r \times \mathbb{R}^+ \to \mathbb{R}^q$ is the control constraint. For technical simplicity, we assume that L, ϕ, φ, M are continuously differentiable with respect to x and t in this paper.

Because many engineering problems can be formulated into the framework of (1) (Loffe, Tikhomirov, & Makowski, 2009; Malanowski, 1995), optimal control theory has vast applications (Ancona & Bressan, 1999, 2007; Hong, 2007). However, due to the complexity of those applications, few of them can be solved analytically. Thus numerical methods are often employed instead. Traditionally, the methods are divided into three categories,
(1) state-space, (2) indirect, and (3) direct methods. *State-space* approaches apply the principle of dynamic programming, which states that each sub-arc of the optimal trajectory must be optimal. It leads to the well-known Hamilton–Jacobi–Bellman (HJB) equations, which are non-linear partial differential equations (PDEs) (Bellman, 1956; Navasca & Krener, 2007). *Indirect methods* employ the necessary condition of optimality known as Pontryagin's maximum Principle (Pontrigin, 1962). This leads to a boundary value problem, which is then solved by numerical methods. Thus this approach is also referred to as "first optimize, then discretize". Typical examples are neighboring extremal algorithm, gradient algorithm, and quasi-linearization algorithm (Balakrishnan & Neustadt, 1964; Bryson & Ho, 1975; Livne, Mineau, Bryson, & Denham, 1964; Mc Reynolds & Bryson, 1965), just to name a few. *Direct methods* take

☆ This work is partially supported by NSF Awards DMS–1419027, DMS–1620345, and ONR AwardN000141310408. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Kok Lay Teo under the direction of Editor Ian R. Petersen.

E-mail addresses: wli83@gatech.edu (W. Li), junlu0@icloud.com (J. Lu), hmzhou@math.gatech.edu (H. Zhou), chow@math.gatech.edu (S.-N. Chow).

http://dx.doi.org/10.1016/j.automatica.2016.12.023
0005-1098/

the idea of "first discretize, then optimize". They convert the original continuous infinite dimensional control problem into a finite dimensional optimization problem. The resulting discrete problem becomes a large scale standard nonlinear programming problem (NLP) which can be solved by many well established algorithms such as Newton's method, Quasi-Newton methods (Acikmese & Blackmore, 2011; Dennis & Schnabel, 1987; Fletcher, 2013; Gill, Murray, & Wright, 1981; Harris & Acikmese, 2014; Lin, Loxton, & Teo, 2014; Loxton, Lin, Rehbock, & Teo, 2012; Loxton, Teo, & Rehbock, 2008; Loxton, Teo, Rehbock, & Yiu, 2009; Nocedal & Wright, 2006). Direct methods are nowadays the most widespread and successfully used techniques.

Different from the existing methods, in this paper, we design a new fast numerical method focusing on a class of optimal control problems called separable problems (Chow, Lu, & Zhou, 2013, 2012a,b; Lu, Diaz-Mercado, Egerstedt, Zhou, & Chow, 2014). Simply put, a path is said to be separable, if there exists finite number of points, called junctions, that divide the path into segments, such that a constraint can only switch from inactive to active (or vice versa) at junctions.

The significance of being separable is that the determination of the entire path boils down to the determination of only a finite number of junctions and the determination of a finite number of optimal trajectories of smaller sizes, for which the constraints are either inactive or active on the entire segment. On the other hand, in many applications, the optimal solution on each segment can be computed either analytically or numerically by efficient algorithms. In this way, the original infinite dimensional problem of finding the whole path is converted into a finite dimensional problem, i.e. determining a finite number of junctions. Thus one may[1] gain a tremendous dimensional reduction.

The resulting finite dimensional problem can be handled by many established algorithms, for example, the gradient descent method. In this case, each steady state of the gradient descent flow can be a local minimizer. It is evident from many applications that the total number of minimizers can often be very large. Therefore, it is highly desirable to design methods that are capable of obtaining the global optimal trajectory. In this paper, we adopt a recently developed global optimization strategy, called intermittent diffusion (ID) (Chow, Yang, & Zhou, 2013). The idea is to add noises (diffusions) to the gradient flow intermittently. When the noise is turned off, one gets a pure gradient flow and it quickly converges to a local minimizer. When the noise is turned on, the perturbed flow becomes stochastic differential equations (SDEs), which has a positive probability to jump out of local traps and converges to other minimizers, including the global one. It can be shown that the local minimizers obtained will include the global one with probability arbitrarily close to 1 if appropriate perturbations are added. We call the method outlined above *Method of Evolving Junctions* (MEJ).

In the literature, the concept of a junction has been introduced in the past (Hartl, Sethi, & Vickson, 1995; Khmelnitsky, 2002), and used in indirect methods (Bonnans & Hermant, 2008; Bonnard, Faubourg, Launay, & Trélat, 2003). Most of them use junctions as shooting parameters to solve the Hamiltonian systems. For example, the one proposed in Bonnans and Hermant (2008) uses a continuation method, also called homotopy method, together with the shooting method for the boundary value ODEs derived from the maximum principle. This is different from how we use junctions. Namely, we directly derive equations that govern the evolution of junctions to achieve the optimal control requirements.

[1] We assume that constraints can be handled efficiently through functions of junctions.

Because MEJ is designed for separable problems, especially for those whose optimal solutions and constraints can be dealt efficiently through junctions, it has the ability to overcome some well-known limitations of the aforementioned three methods. Namely, the HJB approach, which gives the global solution, can be computationally expensive and suffers from the notorious problem known as "curse of dimensionality". Indirect methods guarantee to find local optimal solutions, while finding the global optimal solutions, if possible, requires carefully designed initializations. Direct methods often require finer discretization (smaller time steps) if better accuracy is desired, which increases computational demands. In recent years, various efforts have been devoted to reduce the computational cost (Loxton et al., 2012, 2008, 2009).

We arrange this paper as follows. In Section 2, we explain the idea of separability and give the algorithm for MEJ. In Section 3, we use the new method to solve a robotic path-planning problem, through which we demonstrate the advantages of MEJ.

2. Method of evolving junctions

To simplify our discussion, we introduce MEJ to (1) with only state constraints. Its extension to control constraints is straightforward and hence omitted here.

2.1. The separable structure

Definition 1. A path $x(t)$ is said to be separable if there exists a finite partition: $t_0 < t_1 < t_2 < \cdots < t_N < t_{N+1} = t_f$ such that $x(t)|_{[t_i,t_{i+1}]}$ alternates between segments where constraints are either active or inactive. An optimal control problem (1) is called separable if its optimal path is separable.

The notion of separability has been previously described in Speyer, Mehra, and Bryson (1969) where only trajectories consisting of three parts are considered ($N = 2$). Here we define $\tilde{x}_i := (t_i, x(t_i))$ and call them *junctions*. A junction pair $\tilde{x}_i, \tilde{x}_{i+1}$ determines a trajectory connecting them, which solves either the optimal control in the free space or with active constraints, denoted by $\gamma_0(\tilde{x}_i, \tilde{x}_{i+1})$ or $\gamma_c(\tilde{x}_i, \tilde{x}_{i+1})$ respectively.

The separability allows us to restrict the search of optimal trajectories in a subset H defined by

$$H := \{\gamma : \gamma \text{ is determined by finite junctions}\}.$$

More precisely, if $\gamma \in H$, there exists a sequence of junctions on the boundary of the constraints, $(\tilde{x}_0, \ldots, \tilde{x}_N, \tilde{x}_{N+1})$ such that γ can be represented as

$$\gamma_1(\tilde{x}_0, \tilde{x}_1) \cdot \gamma_2(\tilde{x}_1, \tilde{x}_2) \ldots \gamma_N(\tilde{x}_N, \tilde{x}_{N+1}), \tag{2}$$

where γ_i is either γ_0 or γ_c, and $\gamma_i \cdot \gamma_{i+1}$ denotes the concatenation of two trajectories.

As a result, the cost functional of (1) can be expressed as a function of junctions:

$$J(\tilde{x}_1, \ldots, \tilde{x}_{N+1}) := \sum_{i=1}^{N} J_i(\tilde{x}_i, \tilde{x}_{i+1}),$$

where J_i is the cost on γ_i, i.e.

$$J_i(\tilde{x}_i, \tilde{x}_{i+1}) := \min_{x,u} \int_{t_i}^{t_{i+1}} L(x(t), u(t), t)dt.$$

Moreover, the optimal trajectory connecting $\tilde{x}_i$ and $\tilde{x}_{i+1}$ must not violate the constraints,

$$V(\tilde{x}_i, \tilde{x}_{i+1}) := \min_{t_i \le t \le t_{i+1}, 1 \le k \le p} \phi_k(\gamma_i(t), t) = 0. \tag{3}$$

For convenience, we call $V(\tilde{x}_i, \tilde{x}_{i+1}) = 0$ the visibility constraints.