Control Engineering Practice 18 (2010) 190–197

Contents lists available at ScienceDirect

Control Engineering Practice

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

Min–max control using parametric approximate dynamic programming

Hussam Nosair, Yu Yang, Jong Min Lee *

Chemical and Materials Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 2G6

ARTICLE INFO

Article history:
Received 11 January 2009
Accepted 1 September 2009
Available online 23 September 2009

Keywords:
Worst-case formulation
Uncertain linear systems
Robust control
Optimal control

ABSTRACT

This study presents a computationally efficient approximate dynamic programming approach to control uncertain linear systems based on a min–max control formulation. The optimal cost-to-go function, which prescribes an optimal control policy, is estimated using piecewise parametric quadratic approximation. The approach requires simulation or operational data only at the bounds of additive disturbances or polyhedral uncertain parameters. This strategy significantly reduces the computational burden associated with dynamic programming and is not limited to a particular form of performance criterion as in previous approaches.

1. Introduction

Classical model predictive control (MPC) is susceptible to three main difficulties: obtaining an accurate model, ensuring robustness/stability with respect to uncertainties, and solving a complex online optimization problem such as solving a nonlinear program (Morari & Lee, 1999). In particular, robust stability is a major concern in industrial MPC applications and is mostly addressed through the use of extensive closed-loop simulation prior to implementation (Qin & Badgwell, 2003). This method is expensive and time-consuming because it requires simulation test for all possible combination of important dynamics based on the control engineer's knowledge on the process (Badgwell, 1997).

Robust MPC, on the other hand, is an emerging alternative technique which does not require an accurate deterministic model. The underlying concept is to construct a linear model with uncertain parameters or additive stochastic disturbances for description of all possible processes and to utilize the uncertainty information within the receding horizon optimization framework. Robust MPC minimizes the worst-case performance (i.e., the maximum cost-to-go) based on possible parameter values or stochastic disturbances within their deterministic bounds while respecting constraints for all possible scenarios (Campo & Morari, 1987; Witsenhausen, 1968). The key advantage of robust MPC over classical MPC is thus its indifference to the accuracy of the model in presence of uncertainties.

Lee and Yu (1997) summarizes well two general min–max formulations for solving the robust control problem. One is open-loop formulation where the uncertainty and feedback in the future time steps are ignored. The second is min–max formulation from the viewpoint of closed-loop control where a dynamic program is solved. The open-loop formulation is the essence of most min–max MPC techniques, but it may lead to infeasibility, conservative closed-loop performance, and instability. On the other hand, the closed-loop formulation provides less conservative solution and robust stability under infinite prediction horizon setup (Bemporad, Borrelli, & Morari, 2003; Lee & Yu, 1997; Scokaert & Mayne, 1998). Nevertheless, the excessive computational burden associated with solving DP limits its applications to small-size systems (Bertsekas, 2005).

This class of multi-stage min–max optimization problems have been addressed by different approaches. A scenario tree formulation for linear systems with additive disturbances is suggested to treat a single optimization problem for one initial state only (Scokaert & Mayne, 1998). Linear matrix inequality (LMI) techniques are employed to efficiently compute the worst-case performance (Kothare, Balakrishnan, & Morari, 1996; Wan & Kothare, 2003). These formulations compute the control laws based on the upper bound of a Lyapunov function in the worst case for the stability of feedback control policy. Hence, the resulting control law can be very conservative. An approximate solution to min–max MPC is also suggested to solve the open-loop min–max formulation efficiently using quadratic programming derived from the upper bound of the worst case cost. This approach is validated through its application to an open-loop stable, pilot-scale exothermic reactor (Gruber, Ramirez, Alamo, Bordons, & Camacho, 2009).

Most of the computational burden in solving DP lies in the off-line procedure where the optimal cost-to-go function is

* Corresponding author. Tel.: +1 780 492 8092; fax: +1 780 492 2881.
E-mail addresses: nosair@ualberta.ca (H. Nosair), yyang6@ualberta.ca (Y. Yang), jongmin.lee@ualberta.ca (J.M. Lee).

0967-0661/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.conengprac.2009.09.001

calculated. The resulting cost-to-go function can be used to find an optimal policy mapping a state to a control action, which provides much simpler online implementation than a family of receding horizon techniques. Motivated by this, multi-parametric programming (Borrelli, Bemporad, & Morari, 2003; Tødel, Johansen, & Bemporad, 2003) and DP-based approaches (Börnberg & Diehl, 2006; Diehl & Börnberg, 2004) have been proposed. These app-
roaches, however, are limited to a particular form of performance criterion, e.g., linear cost term for using linear programming.

Recent advancements in the field of approximate dynamic programming (ADP) showed potential success of solving closed-loop formulation via ADP (Lee & Lee, 2006). ADP attempts to circumvent the off-line computational burden, referred to as the curse of dimensionality, by approximately computing the optimal cost-to-go values for potentially important states only. Among the works on ADP is the use of instance-based approximation (Lee, Kaisare, & Lee, 2006), linear programming (de Farias & van Roy, 2003), and polyhedral approximation (Börnberg & Diehl, 2006). Two important preconditions apply to the development of an effective approximation: the choice of approximation that closely approximates the desired cost-to-go function and the efficiency of the update algorithm (de Farias & van Roy, 2003). This paper presents a tailored ADP approach for solving min–max control of linear systems with bounded parameters or additive disturbances. The approach is based on simulation at the bounds of uncertain parameters and piecewise quadratic parametric approximation with online gradient-descent update. The key advantages of this method are its accommodation for a general class of high dimensional linear systems, as well as the need to only consider a simplified description of uncertainties. Three numerical examples including a high-purity distillation column control are provided to illustrate the efficacy of the proposed method.

2. Closed-loop min–max formulation

Consider the following discrete-time uncertain linear state-space system:

$$x_{k+1} = f_k(x_k, u_k, d_k) = A(\theta_k)x_k + B(\theta_k)u_k + Ed_k \tag{1}$$

where $x \in \mathcal{X}$, $u \in \mathcal{U}$, $d \in \mathcal{D}$ are state, input, and exogenous disturbance vectors, respectively. $\theta \in \Theta$ is an uncertain vector that parameterizes the system matrices and can be constant or time-varying. The sets $\mathcal{X}, \mathcal{U}$, and $\mathcal{D}$ are subject to constraints in the form of lower and upper bounds. For simplicity, the full state feedback is assumed to be available at each time.

The following discounted infinite horizon worst-case cost is minimized:

$$\max_{\substack{d_0,\ldots,d_\infty \\ \theta_0,\ldots,\theta_\infty}} \sum_{k=0}^{\infty} \alpha^k \phi(x_k, u_k) \tag{2}$$

where ϕ is the single-stage cost incurred at time k and $\alpha \in (0, 1)$ is a discount factor. It should be noted that ϕ is not limited to a certain type of norm in this formulation and α is used to prevent the total worst-case cost from diverging to infinity.

The closed-loop formulation, where the state is measured at each sample time over the prediction horizon, leads to the following dynamic program:

$$J(x_k, u_k) = \max_{d_k \in \mathcal{D}, \theta_k \in \Theta} [\phi(x_k, u_k) + \alpha J^*(A(\theta_k)x_k + B(\theta_k)u_k + Ed_k)] \tag{3}$$

$$J^*(x_k) = \min_{u_k} J(x_k, u_k) \tag{4}$$

$J^*(x)$ is the optimal cost-to-go value for state x in the infinite horizon formulation. The optimal feedback control policy μ^* maps a state to its optimal control action as follows:

$$u_k = \mu^*(x_k) = \operatorname{argmin}_{u_k} J(x_k, u_k) \tag{5}$$

One can also use a finite-horizon formulation. The only difference from the infinite horizon case is that it necessitates evaluating cost-to-go value at each time step, and thus the resulting optimal control policy depends on time.

The optimal cost-to-go as a function of the state can be computed off-line by applying Bellman's principle of optimality and using DP (Bertsekas, 2005). For the infinite-horizon problem, the cost-to-go function for given state x is obtained by solving this following Bellman equation iteratively:

$$J^*(x_k) = \min_{u_k \in \mathcal{U}_{d_k}} \max_{\in \mathcal{D}, \theta_k \in \Theta} [\phi(x_k, u_k) + \alpha J^*(A(\theta_k)x_k + B(\theta_k)u_k + Ed_k)] \tag{6}$$

The above formulation is equivalent to that found in literature on closed-loop formulation of the min–max problem. For the general case, the various conventional approaches including value iteration and policy iteration to solving this equation suffer from computational burden as exhaustive sampling and discretization are required (Bertsekas, 2005). The following section presents a novel way of representing this cost-to-go function using piecewise quadratic parametric approximation with a sample-based scheme for updating the cost-to-go function, rendering the proposed approach suitable for systems with high dimensionality and/or continuous state and action spaces.

3. Suggested approach

As mentioned in de Farias and van Roy (2003), two preconditions are important in constructing an effective approximation of cost-to-go function. First, a proper approximation structure for the optimal min–max cost function is chosen. The proposed approach, for simplicity and without sacrificing accuracy, uses piecewise parametric quadratic approximation. The advantage of such representation is stability in control and convergence in learning as opposed to using one global nonlinear representation (Nosair & Lee, 2008). Second, it is computationally manageable and requires very little storage of data, as opposed to instance-based learning (Lee et al., 2006) where an enormous amounts of cost-to-go values must be stored and searched through.

However, to choose the appropriate number and locations of the piecewise quadratic approximators, prior knowledge on the shape and structure of the min–max cost-to-go function is necessary. This can be done by simulating the system under possible worst-case scenarios, from which initial cost-to-go values can be computed using the cost function defined previously. More importantly, the proposed scheme takes advantage of the fact that the worst-case always happens at the bounds or vertices of stochastic variables or uncertain parameters in linear systems affected by additive norm-bounded exogenous disturbances and/or polyhedral parametric uncertainty (Bemporad et al., 2003). Hence, simulation scenarios need only to alternate between upper and lower bounds of the uncertainty vector. The proposed approach in a nutshell is therefore:

1. Data acquisition: Acquire initial min–max data to understand the structure of the optimal min–max cost function.
2. Value function approximation off-line: Construct a piecewise parametric quadratic cost function from acquired data.
3. Online value function update: Update the approximate function using online data until it converges.