Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-14 (2015) 302–307

Reachability Analysis and Control Synthesis for Uncertain Linear Systems in MPT ⋆

Michal Kvasnica [*] Bálint Takács [*] Juraj Holaza [*] Deepak Ingole [*]

[*] *Faculty of Chemical and Food Technology, Slovak University of Technology in Bratislava (e-mail: {michal.kvasnica, balint.takacs, juraj.holaza, deepak.ingole}@stuba.sk).*

Abstract: Software tools play an important dissemination role by bringing cutting-edge theoretical algorithms into the hands of researchers and practitioners. This paper introduces a new robust analysis and control module of the Multi-Parametric toolbox, which is one of the most successful open-source tools in the field. We discuss how robust reachable and invariant sets can be computed using a convenient user interface. Such sets play an important role in many control-oriented tasks, such as in design of recursively feasible optimization-based control laws. Moreover, the new module also allows to synthesize robustly stabilizing linear controller and, more importantly, offers robust Model Predictive Control (MPC) synthesis features. The main aim of the paper is illustrate how complex tasks can be implemented using simple operators such that more sophisticated algorithms could be developed easily by the research community.

Keywords: Reachability analysis, robust control, computer-aided control design

1. INTRODUCTION

Although robust analysis and control synthesis for uncertain systems is a vibrant research topic, few software tools are available to give powerful theoretical concepts into the hands of general public, especially when the analysis/synthesis tasks are to be performed while taking process constraints into account.

The primary objective of this paper is to introduce a new robust analysis and control module of the Multi-Parametric Toolbox (MPT), which was originally introduced in Kvasnica et al. (2003) and rewritten from the ground up in Herceg et al. (2013). MPT is a *freely* available Matlab-based toolbox which allows to analyze dynamical systems, synthesize optimization-based controllers for them and deploy such controllers to hardware platforms in form of executable code. Moreover, the toolbox also provides a rich library for computational geometry and exposes sophisticated algorithms in an easily accessible manner. Popularity of the toolbox among theoreticians and practitioners alike is documented by more than 6000 unique installations in the past 1.5 years. However, the original implementation of MPT did not support uncertain systems. In fact, the number one feedback we have received was the request to include support for robust analysis and control synthesis.

In this paper we fix this omission and discuss how robust analysis and control synthesis can be performed for uncertain linear systems in the discrete time domain via a convenient user interface. For such systems the new [1] robust module of MPT allows to perform following tasks:

- compute forward and backward reachable sets;
- construct maximal positive invariant and control invariant sets;
- design robust linear controllers;
- synthesize robust model predictive controllers (MPC).

The paper is composed of five main parts. First, Section 2 provides a detailed description of modeling principles and shows how the dynamics of uncertain systems can be defined in the MPT framework. Then, Section 3 discusses theoretical concepts of robust one-step reachable sets and illustrates how they can be computed using a convenient user interface. Subsequently, Section 4 illustrates how the concept of one-step reachability can be extended to construct (maximal) positive and control invariant sets. Such sets play an important role e.g. in MPC design when recursive feasibility is to be enforced. The theoretical foundations of the algorithms discussed in Sections 3 and 4 are based on Blanchini (1999), Blanchini and Miani (2007), and (Borrelli et al., 2014, Chapter 11). Design of linear state-feedback controllers for uncertain linear systems is then discussed in Section ??. Finally, Section 5 describes synthesis of robust MPC controllers (Bemporad and Morari, 1999; Camacho and Bordons, 1999) that explicitly take process constraints into account.

⋆ The authors gratefully acknowledge the contribution of the Scientific Grant Agency of the Slovak Republic under the grants 1/0403/15 and 1/0973/12. The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement no 607957 (TEMPO) and the internal grant of the Slovak University of Technology in Bratislava for support of young researchers.

[1] The robust module is available since MPT version 3.1. If the reader is a first-time user, she should follow the installation instructions at `http://control.ee.ethz.ch/~mpt/3/Main/Installation`. If a previous version was already installed, the update can be obtained by running `tbxmanager update` in Matlab.

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.09.474

2. UNCERTAIN LINEAR SYSTEMS

Throughout this paper we consider linear time-invariant (LTI) systems in the discrete time domain whose dynamics is driven by a state-space equation of the form

$$x^+ = A(\lambda)x + B(\lambda)u + Ed, \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ is the state vector at the current time instant, x^+ is the successor state at the next time instant, $u \in \mathbb{R}^{n_u}$ is the vector of control inputs, and $d \in \mathbb{R}^{n_d}$ is the unmeasured additive disturbance. The dynamics is subject to polytopic uncertainty in the A and B matrices:

$$A(\lambda) = \sum_{i=1}^{n_A} \lambda_i A_i,\ 0 \le \lambda_i \le 1,\ \sum_{i=1}^{n_A} \lambda_i = 1, \tag{2a}$$

$$B(\lambda) = \sum_{j=1}^{n_B} \lambda_j B_j,\ 0 \le \lambda_j \le 1,\ \sum_{j=1}^{n_B} \lambda_j = 1, \tag{2b}$$

where A_i, $i = 1, \ldots, n_A$ and B_j, $j = 1, \ldots, n_B$ are the vertices of the polytopic uncertainty.

The system in (1) operates subject to following constraints:

$$x \in \mathcal{X},\ u \in \mathcal{U},\ d \in \mathcal{D} \tag{3}$$

where $\mathcal{X} \subset \mathbb{R}^{n_x}$, $\mathcal{U} \subset \mathbb{R}^{n_u}$, and $\mathcal{D} \subset \mathbb{R}^{n_d}$ are bounded polytopic sets.

To define an uncertain linear system, the user calls the `ULTISystem` constructor (ULTI stands for "uncertain linear time-invariant") as follows:

```
sys = ULTISystem('A', A, 'B', B, 'E', E)
```

Here, the vertices of the polytopic uncertainty can be specified as a cell array, i.e., `A = {A1, A2, ..., An}` and `B = {B1, B2, ..., Bm}`. The `E` matrix can be omitted, in which case $E = I_{n_x}$ is assumed [2]. The `B` parameter can also be left out, in which case the autonomous system

$$x^+ = A(\lambda)x + Ed \tag{4}$$

will be considered along with constraints $x \in \mathcal{X}$ and $d \in \mathcal{D}$.

The constraints in (3) can be specified in two ways. If only simple min/max bounds are desired, i.e., $\mathcal{X} = \{x \mid x_{\min} \le x \le x_{\max}\}$, $\mathcal{U} = \{u \mid u_{\min} \le u \le u_{\max}\}$, $\mathcal{D} = \{d \mid d_{\min} \le d \le d_{\max}\}$, these can be specified by setting

```
sys.x.min = xmin, sys.x.max = xmax
sys.u.min = umin, sys.u.max = umax
sys.d.min = dmin, sys.d.max = dmax
```

Remark 2.1. Note that $d_{\min} = 0$ and $d_{\max} = 0$ are assumed by default, i.e., no additive disturbance acts in (1) unless explicitly set otherwise by the user. On the contrary, $\pm\infty$ min/max bounds on states and inputs are assumed by default. □

Alternatively, it is possible to specify $\mathcal{X}$, $\mathcal{U}$, and/or $\mathcal{D}$ as generic polytopic sets $\mathcal{S} = \{z \mid Hz \le h\}$. This is achieved by creating such a set using the `Polyhedron` command [3]:

```
S = Polyhedron(H, h)
```

Subsequently, the polytope can be used as a state constraint:

```
sys.x.with('setConstraint')
sys.x.setConstraint = S
```

Here, the first command adds the ability to specify polytopic constraints while the second line actually assigns a specific polytopic set. Polytopic input/disturbance constraints can be specified similarly by using `sys.u` and `sys.d`, respectively.

Remark 2.2. Important to notice is that the min/max bounds can be used simultaneously with the polytopic constraints. In such a case the final constraint is the intersection of the two sets. As a consequence of this behavior with conjunction of Remark 2.1, if only polytopic disturbance bounds $\mathcal{D}$ are desired, it is necessary to set `sys.d.min = -Inf` and `sys.d.max = Inf`. □

3. ROBUST REACHABLE SETS

MPT allows to compute forward and backward reachable sets for the dynamics in (1), as well as for the autonomous version in (4) provided the sets to be propagated are polytopic. In this section we first cover the theoretical definitions of such reachable sets before describing the low-level code implementation in Section 3.2. Finally, a convenient single-command user interface is described in Section 3.3.

3.1 Theoretical Background

The backward reachable set of a set $\mathcal{X}$ is the set of states from which the evolution of the uncertain system enters the set $\mathcal{X}$ in one time step despite all possible variations of the polytopic uncertainty in (2) and all possible disturbances $d \in \mathcal{D}$, cf. (Borrelli et al., 2014, Def. 11.14). Such a set is often called the *preimage* of the set $\mathcal{X}$ under the dynamics in (1) or (4). Formally, for the autonomous system (4) the one-step backward reachable set of the set $\mathcal{X}$ is

$$\mathrm{Pre}(\mathcal{X}) \triangleq \{x \mid A(\lambda)x + Ed \in \mathcal{X},\ \forall d \in \mathcal{D},\ \forall \lambda \in \Lambda\}, \tag{5}$$

where $\Lambda = \{\lambda \mid 0 \le \lambda_i \le 1,\ \sum_i \lambda_i = 1\}$ is a unit simplex. If the system has control inputs, the definition is

$$\mathrm{Pre}(\mathcal{X}) \triangleq \{x \mid \exists u \in \mathcal{U} \text{ s.t. } A(\lambda)x + B(\lambda)u + Ed \in \mathcal{X},\ \forall d \in \mathcal{D},\ \forall \lambda \in \Lambda\}. \tag{6}$$

Simply speaking, the *pre* set answers the following question: where can the system start from such that the state update enters the target set $\mathcal{X}$ in one time step despite the worst possible disturbance/uncertainty.

For the autonomous system in (4) with no polytopic uncertainty, i.e., $n_A = 1$ in (2), the pre-set of $\mathcal{X}$ in (5) can be computed by

$$\mathrm{Pre}(\mathcal{X}) = (\mathcal{X} \ominus (E \circ \mathcal{D})) \circ A. \tag{7}$$

Here, $\ominus$ is the Pontryagin difference operator of two sets, i.e., $\mathcal{P} \ominus \mathcal{Q} = \{x \mid p + q \in \mathcal{P},\ \forall q \in \mathcal{Q}\}$. Moreover, $\mathcal{P} \circ A$ with a set $\mathcal{P}$ and a matrix A denotes the inverse mapping of $\mathcal{P}$ under linear map A, i.e., $\mathcal{P} \circ A = \{x \mid Ax \in \mathcal{P}\}$. Finally, $E \circ \mathcal{D}$ denotes the direct mapping of a set $\mathcal{D}$ under linear map E, i.e., $E \circ \mathcal{D} = \{Ed \mid d \in \mathcal{D}\}$. If a system with control inputs as in (1) is considered, the set in (6) can be computed by

$$\mathrm{Pre}(\mathcal{X}) = ((\mathcal{X} \ominus (E \circ \mathcal{D})) \oplus (-B \circ \mathcal{U})) \circ A, \tag{8}$$

[2] I_{n_x} is the identity matrix of dimension $n_x \times n_x$.

[3] MPT also allows to use unbounded polyhedra on top of bounded polytopes. However, to keep the theoretical exposition simple, we restrict ourselves to bounded polytopic sets.