Minimum resource commitment for reachability specifications in a discrete time linear setting

Riccardo Vignali and Maria Prandini

***Abstract*—This paper addresses control input design for a discrete time linear system. The goal is to satisfy a reachability specification and, at the same time, minimize the number of inputs that need to be set (influential inputs). To this purpose, we introduce an appropriate input parametrization so that, depending on the parameter values, some of the inputs act as control variables, while the others are treated as disturbances and can take an arbitrary value in their range. We then enforce the specification while maximizing the number of disturbance inputs. Two approaches are developed: one based on an open loop scheme and one based on a compensation scheme. In the former, we end up solving a linear program. In the latter, the parametrization is extended so as to allow the influential inputs to depend on the non-influential ones, and the problem is reduced to a mixed integer linear program. A comparison between the two approaches is carried out, showing the superiority of the latter. Possible applications to system design and security of networked control systems are briefly discussed in the introduction.**

I. Introduction

The goal of system verification is to evaluate if a system behaves as desired. Commonly, the desired behavior is referred to as a *specification* and is associated to the evolution in time of some variables of the system (state or output). In particular, *reachability* and *safety* verification problems have been extensively studied in the literature. They consist of checking if the state can reach a desired *target* set (reachability) or can keep evolving within some given *safe* set (safety), and possibly designing the control input enforcing such behavior.

In this work, we address a finite horizon reachability verification problem for a system with multiple control inputs. Among the possible solutions to the problem, we look for the one that corresponds to the maximal number of *non-influential inputs*, i.e., inputs that can take an arbitrary value in their range without compromising the satisfaction of the specification. The presence of this requirement makes the verification problem non standard and calls for novel techniques to tackle it.

Detecting non-influential inputs can be particularly useful in system design, when one has to verify the correct functioning of some complex system, subject to multiple inputs. In this context, it is common to model an undesired behavior of the system in terms of a specification and then to detect if there is some assignment of the inputs that causes that behavior. Based on the identified assignment (the witness of the malfunctioning), one can make a diagnosis of the faulty behavior and appropriately redesign the system. A possible approach to identify a witness is *model checking*, [1], which relies on a model of the system to explore in some efficient way all its possible evolutions. For complex systems with many inputs, it is typically the case that only a limited number of inputs is influential and hence the number of witnesses is infinite. Model checking techniques would then provide one of the many witnesses, chosen out of the pool without any explicit criterion, which makes it difficult to understand the actual cause of the undesired behavior and to improve the design. To ease the diagnosis of the system misbehavior, it is hence useful to identify the inputs that are actually influential. This is the goal of our paper, which offers a novel approach with respect to model checking to the detection of inputs assignments satisfying a finite horizon reachability specification, in that it contemporarily fulfills the requirement of identifying the influential inputs.

The problem of designing the inputs of a dynamical system so as to make its evolution satisfy some specification has been extensively treated in literature, and effective solutions have been developed for a variety of classes of systems. Many of these techniques are simple derivation of the ones developed in the model checking context. For a large class of specifications, the problem of verifying if a discrete system S satisfies a specification can be reduced to a reachability test on the enlarged system obtained by making S interact with the *test automaton* (see [2], [1], [3]) that translates that specification. In the case of finite state space the resulting reachability test can then be efficiently tackled by means of model checkers, like SPIN [4], UPPAAL [5], NuSMV [6], to name a few. For continuous state systems, these model checking techniques are not directly applicable, due to the uncountable number of values that the state can take. For this reason, alternative methods have been proposed in literature. Some of them rely on the computation (either exact or approximated) of the reachable set of the system under test, so as to consider in a compact way all its infinite possible evolutions (see [7] [8], [9], [10], [11], [12], [13]). Other common approaches are based on abstracting the continuous system to a discrete one (by means, for example, of a *bisimulation* [14]), and then analyzing the latter via model checking techniques (see [15], [16], [17]). Many of these methods can be extended to the class of *hybrid* systems, i.e., systems with interacting discrete and continuous dynamics (see e.g. [18], [9], [19]).

Our resolution approach for the detection of non-influential inputs has been strongly influenced by the optimization-based works in [9] and [13]. Indeed, optimization appears to be the most suitable approach to address the requirement of

R. Vignali and M. Prandini are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milano, 20133 Italy (e-mail: {riccardomaria.vignali, maria.prandini}@polimi.it)

This work is partially supported by the European Commission under the project UnCoVerCPS with grant number 643921.

maximizing the number of non-influential inputs. We focus on the class of linear systems and formulate an approach that rests on an appropriate parametrization of the inputs so that the influential inputs are treated as actual control variables and the non-influential ones as disturbances taking an arbitrary value in their range. We then enforce the reachability specification while maximizing the number of disturbance inputs. Related work can be found in [20] and [21], where the aim is to appropriately set the range of some pre-specified inputs so as to make them non-influential.

We start by presenting a technique based on the optimization of an open loop control scheme, which leads to a linear program to be solved. This technique was proposed in [22], and extended in [23] to a class of hybrid systems with a cascading structure that allows decomposability. The novel part of this paper consists in the introduction of an alternative technique inspired by the *disturbance compensation* scheme introduced in [24], where we allow the influential inputs to depend on the non-influential ones, and in a comparative analysis between the two techniques. Differently from [24], however, the disturbance here is not an exogenous signal, but a subset of the control inputs whose cardinality has to be maximized. We then show that the problem of designing the compensation scheme that maximizes the number of non-influential inputs while satisfying a reachability specification can be solved by means of robust optimization, that, in the case of polytopic target sets, reduces to a Mixed Integer Linear Programming (MILP) problem. The comparison between the open loop scheme and the compensation scheme shows that both of them can be easily extended to further specifications than reachability and that the latter scheme outperforms the former in terms of number of non-influential inputs that are detected.

Interestingly, the compensation scheme has some potential for addressing security of networked control systems. In a typical networked control system the plant is connected to the controller through a communication network, that carries the control input and the system output signals. Because the data channels are usually unprotected, the control system is vulnerable to threats (see [25], [26]). A typical approach to minimize the risk is to protect the actuators data channels via encryption (see [27]), so that the attacker can no longer have access to the communication channel. Since carrying out the encryption of many data channels can be expensive, one could minimize the number of signals that need to be encrypted by identifying the influential inputs that need to be protected so as to guarantee the safe operation of the controlled system. The rest of the paper is structured into two main parts related to the open loop scheme (Section II) and the compensation scheme (Section III), which include problem formulation and resolution. Extensions of both schemes are presented in Section IV. Section V is devoted to some numerical examples. Section VI concludes the paper with some final remarks.

II. Open loop scheme

In this section, we propose a solution to the considered input design problem that rests on an appropriate parametrization of the input variables as set-valued signals, and show that this parametrization allows to reformulate the problem as a robust optimization program. In turn, if the target set is a polytope, the robust optimization program reduces to an LP problem.

A. Problem formulation and resolution

Our goal is to steer the state $x \in \mathbb{R}^n$ of a system from $x(0) = x_0$ to a given convex set $\mathcal{X}_f \subset \mathbb{R}^n$ at some time T, i.e., $x(T) \in \mathcal{X}_f$. We suppose that x evolves affected by m scalar control inputs u_i, $i = 1, \dots, m$, according to the discrete-time linear dynamics

$$x(k+1) = Ax(k) + B_1 u_1(k) + \cdots + B_m u_m(k). \qquad (1)$$

Inputs u_i, $i = 1, \dots, m$, take values in the intervals $[\underline{u}_i, \overline{u}_i]$, $i = 1, \dots, m$, and we aim at appropriately design them so as to satisfy the reachability condition $x(T) \in \mathcal{X}_f$. Among all the admissible solutions, we look for the one where the number of inputs that have to be set to some specific value is minimized, while the others are non-influential and can be set to an arbitrary value within their range.

We next formulate the input design problem as an optimization problem, where each input u_i is treated as a set-valued signal whose range is maximized while imposing the reachability specification. To this purpose, let us introduce the optimization variables β_i and $\tilde{u}_i$, which are respectively a scalar parameter taking values in $[0, 1]$ (defining the amplitude of the range of values for input u_i) and a single-valued signal taking values in $[\underline{u}_i, \overline{u}_i]$ (defining the reference value for u_i). For each $i = 1, ..., m$ and $k = 0, ..., T-1$, the input u_i at time k is expressed as follows

$$u_i(k) = (1-\beta_i)\tilde{u}_i(k) + \frac{\underline{u}_i + \overline{u}_i}{2}\beta_i + \frac{\overline{u}_i - \underline{u}_i}{2}\beta_i w_i(k), \qquad (2)$$

where w_i is a set-valued auxiliary signal taking values in $[-1, 1]$. The resulting range for $u_i(k)$ is given by

$$R_i(k) = [\tilde{u}_i(k) + \beta_i(\underline{u}_i - \tilde{u}_i(k)),\, \tilde{u}_i(k) + \beta_i(\overline{u}_i - \tilde{u}_i(k))],$$

which entails that $u_i(k) = \tilde{u}_i(k)$ when $\beta_i = 0$, whereas, at the opposite extreme, $u_i(k) \in [\underline{u}_i, \overline{u}_i]$ when $\beta_i = 1$.

Let $|\mathcal{C}|$ denote the cardinality of some set $\mathcal{C}$. Then, the problem of determining the minimum number of influential inputs and set them to an appropriate value for satisfying the reachability condition can be rephrased as the following robust optimization program:

$$\begin{aligned} \max_{\{\beta_i \in [0,1], \tilde{u}_i(k) \in [\underline{u}_i, \overline{u}_i], k=0,\dots,T-1\}_{i=1}^m} \quad & |\{i : \beta_i = 1\}| \qquad (3) \\ & x(T) \in \mathcal{X}_f \\ & x(k+1) = Ax(k) + B_1 u_1(k) + \cdots + B_m u_m(k) \\ & u_i(k) = (1-\beta_i)\tilde{u}_i(k) + \frac{\underline{u}_i + \overline{u}_i}{2}\beta_i + \frac{\overline{u}_i - \underline{u}_i}{2}\beta_i w_i(k) \\ & \forall w_i(k) \in [-1, 1],\ i = 1, \dots m,\ k = 0, \dots, T-1 \end{aligned}$$

where the number of β_i's that are set to 1 is maximized, subject to some constraints representing the reachability condition, the linear state evolution, and the set-valued parametrization of the control inputs.

Problem (3) is hard to solve since the cost function and the bilinear term in the parametrization (2) of input u_i make it non-convex. We can however reduce it to a robust convex optimization problem. We first need to reparameterize u_i in (2) as follows

$$u_i(k) = u_{\beta,i}(k) + \frac{\underline{u}_i + \overline{u}_i}{2}\beta_i + \frac{\overline{u}_i - \underline{u}_i}{2}\beta_i w_i(k), \tag{4}$$

where $u_{\beta,i}(k) = (1-\beta_i)\tilde{u}_i(k) \in [(1-\beta_i)\underline{u}_i,\ (1-\beta_i)\overline{u}_i]$. Then, $\tilde{u}_i(k)$ can be recovered from $u_{\beta,i}(k)$ and β_i:

$$\tilde{u}_i(k) = \begin{cases} \frac{u_{\beta,i}(k)}{1-\beta_i} & \text{if } \beta_i \in [0,1) \\ 0 & \text{if } \beta_i = 1. \end{cases}$$

Moreover, the range $R_i(k)$ of $u_i(k)$ can be expressed as $R_i(k) = [u_{\beta,i}(k) + \beta_i\underline{u}_i,\ u_{\beta,i}(k) + \beta_i\overline{u}_i]$. As for the cost function, maximizing the cardinality of set $\{i : \beta_i = 1\}$ coincides with considering vector $\gamma = [\gamma_1\ \gamma_2 \dots \gamma_m]'$ with $\gamma_i = 1 - \beta_i$ and enhancing its sparsity. The sparsity of γ can be maximized by minimizing the number of its non-zero elements, i.e., its ℓ_0-norm. Given that the ℓ_0-norm is non-convex, we minimize the ℓ_1-norm $\|\gamma\|_1 = \sum_{i=1}^m |1-\beta_i|$ in place of it. Since $\beta_i \in [0,1]$ and, hence, $\|\gamma\|_1 = m - \sum_{i=1}^m \beta_i$, minimizing $\|\gamma\|_1$ is equivalent to maximizing $\sum_{i=1}^m \beta_i$, which is a convex function of β_i's.

Remark 1. *The idea of approximating the ℓ_0 norm with the ℓ_1 norm is not new, and it is commonly adopted when looking for a sparse solution to a system of linear equations with fewer equations than unknowns for, e.g., sparse signal recovery, image processing, statistical estimation, compressive sensing, to name a few applications. In general, the ℓ_1 norm is only an approximation to the ℓ_0 norm, and in fact reweighted versions of the ℓ_1 norm that better approximate the ℓ_0-norm are presented in [28]. The interested reader is referred to [28] and the references therein for further details.*

We can now formulate the robust optimization program

$$\begin{aligned} \max_{\{\beta_i\in[0,1],u_{\beta,i}(k),k=0,\dots,T-1\}_{i=1}^m} & \sum_{i=1}^m \beta_i \qquad (5) \\ & x(T) \in \mathcal{X}_f \\ & x(k+1) = Ax(k) + B_1u_1(k) + \dots + B_mu_m(k) \\ & u_i(k) = u_{\beta,i}(k) + \frac{\underline{u}_i + \overline{u}_i}{2}\beta_i + \frac{\overline{u}_i - \underline{u}_i}{2}\beta_i w_i(k) \\ & (1-\beta_i)\underline{u}_i \le u_{\beta,i}(k) \le (1-\beta_i)\overline{u}_i \\ & \forall w_i(k) \in [-1,1],\ i = 1,\dots m,\ k = 0,\dots,T-1, \end{aligned}$$

which is convex since $x(T)$ is linear as a function of the optimization variables β_i and $u_{\beta,i}$, $i = 1,\dots,m$, and $\mathcal{X}_f$ is convex.

We next show that (5) reduces to an LP problem when $\mathcal{X}_f$ is a polytope or inner-approximated by a polytope, i.e.,

$$\{x \in \mathbb{R}^n :\ H_a x \le H_b\} \subseteq \mathcal{X}_f. \tag{6}$$

To this purpose we introduce some compact notations, i.e.,

$$U = \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(T-1) \end{bmatrix},\ U_\beta = \begin{bmatrix} u_\beta(0) \\ u_\beta(1) \\ \vdots \\ u_\beta(T-1) \end{bmatrix},\ W = \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(T-1) \end{bmatrix},$$

where $\beta = [\beta_1, \beta_2, \dots, \beta_m]'$, $u = [u_1, u_2, \dots, u_m]'$, $u_\beta = [u_{\beta,1}, u_{\beta,2}, \dots, u_{\beta,m}]'$, $w = [w_1, w_2, \dots, w_m]'$.
Then, $x(T)$ can be written as

$$x(T) = A^T x_0 + \mathbf{B}_T U, \tag{7}$$

where $\mathbf{B}_T$ is obtained by extracting the last n rows of matrix

$$\mathbf{B} = \begin{bmatrix} B & 0 & 0 & 0 \\ AB & B & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ A^{T-1}B & A^{T-2}B & \dots & B \end{bmatrix}, \tag{8}$$

with $B = [B_1\ B_2\ \dots\ B_m]$. Also, (4) can be expressed as

$$U = U_\beta + (D_W + M)\beta, \tag{9}$$

where

$$D_W = \begin{bmatrix} \operatorname{diag}([a_1w_1(0), \dots, a_mw_m(0)]) \\ \vdots \\ \operatorname{diag}([a_1w_1(T-1), \dots, a_mw_m(T-1)]) \end{bmatrix},$$

$$M = \begin{bmatrix} \operatorname{diag}([\mu_1, \dots, \mu_m]) \\ \vdots \\ \operatorname{diag}([\mu_1, \dots, \mu_m]) \end{bmatrix},$$

with $a_i = \frac{\overline{u}_i - \underline{u}_i}{2}$ and $\mu_i = \frac{\overline{u}_i + \underline{u}_i}{2}$, $i = 1, \dots, m$.
By plugging expression (9) into (7), and using the inner-approximation (6), the optimization problem (5) can be rewritten as the following robust LP problem

$$\max_{\{\underline{\mathbf{U}}(1-\beta)\le U_\beta\le\overline{\mathbf{U}}(1-\beta),\ \beta\in[0,1]^m\}} \mathbf{1}'_m\beta \tag{10}$$

$$\begin{bmatrix} H_a\mathbf{B}_T & H_a\mathbf{B}_T(D_W + M) \end{bmatrix} \begin{bmatrix} U_\beta \\ \beta \end{bmatrix} \le H_b - H_aA^Tx_0, \tag{11}$$

$$\forall W \in [-1,1]^{mT}$$

where $\mathbf{1}_m$ is a column vector of m ones, and

$$\underline{\mathbf{U}} = \begin{bmatrix} \operatorname{diag}([\underline{u}_1, \dots, \underline{u}_m]) \\ \vdots \\ \operatorname{diag}([\underline{u}_1, \dots, \underline{u}_m]) \end{bmatrix},\quad \overline{\mathbf{U}} = \begin{bmatrix} \operatorname{diag}([\overline{u}_1, \dots, \overline{u}_m]) \\ \vdots \\ \operatorname{diag}([\overline{u}_1, \dots, \overline{u}_m]) \end{bmatrix}.$$

Problem (10) is a semi-infinite linear program with a finite number of optimization variables but an infinite number of constraints due to the fact that constraint (11) has to hold for all realizations of the set-valued signal W. Solving semi-infinite optimization problems is generally difficult, [29], [30], [31], [32]. In the problem at hand, however, this is not the case since it can be reduced to a finite LP problem, which can be efficiently solved by means of standard LP solvers like CPLEX [33].

Proposition II.1. *The semi-infinite linear optimization program* (10) *is equivalent to the following finite LP problem*

$$\max_{\{\underline{\mathbf{U}}(1-\beta)\le U_\beta\le\overline{\mathbf{U}}(1-\beta),\ \beta\in[0,1]^m\}} \mathbf{1}'_m\beta \tag{12}$$

$$\begin{bmatrix} H_a\mathbf{B}_T & H_a\mathbf{B}_TM + \Xi \end{bmatrix} \begin{bmatrix} U_\beta \\ \beta \end{bmatrix} \le H_b - H_aA^Tx_0,$$

where

$$(\Xi)_{ij} = \|a_j[(H_a\mathbf{B}_T)_{ij}, (H_a\mathbf{B}_T)_{i\,j+m}, \dots, (H_a\mathbf{B}_T)_{i\,j+m(T-1)}]'\|_1.$$

Proof: Let us consider the i-th row of constraint (11), i.e.,

$$(H_a\mathbf{B}_T)_i U_\beta + (H_a\mathbf{B}_T D_W)_i \beta + (H_a\mathbf{B}_T M)_i \beta \\ \leq (H_b - H_a A^T x_0)_i.$$

The set-valued signal W appears only in the term $(H_a\mathbf{B}_T D_W)_i\ \beta$ of the left-hand-side of such an inequality. We then just need to impose that the inequality is satisfied for those values of W that maximize $(H_a\mathbf{B}_T D_W)_i\beta$. Given that β is non-negative, and that element $(H_a\mathbf{B}_T D_W)_{ij}$ of $(H_a\mathbf{B}_T D_W)_i$ depends only on the values $w_j(k)$, $k = 0, 1, \ldots, T-1$, of the j-th component of w, we can independently maximize each j-th entry of $(H_a\mathbf{B}_T D_W)_i$. Since $(H_a\mathbf{B}_T D_W)_{ij} = \xi_{ij}' \left[w_j(0) \quad \ldots \quad w_j(T-1)\right]'$, where $\xi_{ij} = a_j[(H_a\mathbf{B}_T)_{ij}, (H_a\mathbf{B}_T)_{i\,j+m}, \ldots, (H_a\mathbf{B}_T)_{i\,j+m(T-1)}]'$, we obtain $\max_{W\in[-1,1]^{mT}}(H_a\mathbf{B}_T D_W)_{ij} = \|\xi_{ij}\|_1$, which finally leads to the constraint in (12). ■

III. Compensation scheme

In this section, inspired by the *disturbance compensation* scheme in [24], we propose an alternative approach to the input design problem under investigation where the influential inputs are allowed to depend on the non-influential ones. We show that the problem of maximizing the number of non-influential inputs while satisfying a reachability specification can be solved by means of robust optimization, that reduces to a MILP problem in the case of a polytopic target set. The integer component in the MILP problem is represented by the β's parameters which are now binary variables setting the range of an input either to a singleton ($\beta = 0$) or to the full admissible interval ($\beta = 1$). The obtained non-influential inputs are free to be set and can be chosen so as to optimize some performance criterion. This way, one can obtain a multi-objective control scheme with prioritized goals: reachability first, then performance.

The rest of the section is organized as follows. In Section III-A we formulate the problem for a discrete time linear system and show how to rephrase it as a MILP problem. In Section III-B we present a comparative analysis with the open loop with no compensation scheme proposed in Section II.

A. Problem formulation and resolution

Consider the discrete time linear system (1), where $x \in \mathbb{R}^n$ is the state vector and $u_i \in \mathbb{R}$, $i = 1, \ldots, m$, are m scalar control inputs taking values in a bounded set: $u_i \in [\underline{u}_i,\ \overline{u}_i] \subset \mathbb{R}$, $i = 1, \ldots, m$. System (1) can be rewritten in the compact form:

$$x(k+1) = Ax(k) + Bu(k) \tag{13}$$

where $u(k) \in \mathcal{U} = [\underline{u},\ \overline{u}] \subset \mathbb{R}^m$ with $\underline{u} = [\underline{u}_1, \underline{u}_2, \ldots, \underline{u}_m]'$ and $\overline{u} = [\overline{u}_1, \overline{u}_2, \ldots, \overline{u}_m]'$.

As in Section II, our goal is to design the system inputs so as to steer the state of the system into a target set $\mathcal{X}_f \subset \mathbb{R}^n$, in some finite time T, while maximizing the number of non-influential inputs that can take an arbitrary value in their range while guaranteeing that the reachability specification is satisfied. Differently from the open-loop scheme in Section Section II, however, influential inputs are allowed to depend on non-influential inputs, which entails that the former (control) inputs can eventually compensate for the latter (disturbance) inputs. Let $\mathcal{M}$ denote the set of all input indexes, i.e., $\mathcal{M} = \{1, \ldots, m\}$. The goal is to maximize the cardinality of the set $\mathcal{N} \subseteq \mathcal{M}$ of indexes of the non-influential inputs, while allowing the influential ones to depend on the past values of the non-influential ones. The problem can be formally stated as follows:

$$\max_{\mathcal{N}\in 2^{\mathcal{M}},\ \{g_{i,k}:|\mathcal{N}|\times k\to\mathbb{R}\}_{i\in\mathcal{M}\setminus\mathcal{N}}^{k\in\{0,\ldots T-1\}}} \quad |\mathcal{N}| \tag{14}$$

$$\begin{aligned}
&x(T) \in \mathcal{X}_f,\\
&x(k+1) = Ax(k) + Bu(k),\\
&u_i(k) = g_{i,k}(u_{j_1}(0), \ldots, u_{j_{|\mathcal{N}|}}(0), \ldots, u_{j_1}(k-1), \ldots, u_{j_{|\mathcal{N}|}}(k-1)),\\
&i \in \mathcal{M}\setminus\mathcal{N},\ \forall u_{j_h} \in \left[\underline{u}_{j_h},\ \overline{u}_{j_h}\right],\ j_h \in \mathcal{N},\ h = 1, 2, \ldots |\mathcal{N}|,\\
&k = 0, \ldots, T-1,
\end{aligned}$$

where $2^{\mathcal{C}}$ denotes the power set of set $\mathcal{C}$.

Note that problem (14) reduces to the open loop scheme in Section II if all functions $g_{i,k} : |\mathcal{N}| \times k \to \mathbb{R}$, $i \in \mathcal{M}\setminus\mathcal{N}$, $k \in \{0, \ldots T-1\}$, are constant and independent of non-influential inputs. In the general case, problem (14) is intractable because the optimization has to be performed over all subsets $\mathcal{N}$ of $\mathcal{M}$ and over all functions $g_{i,k} : |\mathcal{N}| \times k \to \mathbb{R}$, $i \in \mathcal{M}\setminus\mathcal{N}$ and $k \in \{0, \ldots T-1\}$. Moreover, problem (14) is also semi-infinite, since the constraint on reaching the target set has to be satisfied for any possible realization of the non-influential inputs. We next propose a tractable formulation of problem (14), which rests on an appropriate linear parametrization of the $g_{i,k}$ functions.

1) Input parametrization: In order to partition the inputs into two sets, one grouping together the influential inputs and the other one the non-influential inputs, we introduce vector $\beta = [\beta_1, \ldots, \beta_m]' \in \{0,1\}^m$ of boolean variables β_i, $i = 1, 2, \ldots, m$, that identify the set to which each input belongs. In particular, β_i is set equal to 1 if input u_i is non-influential and it is set equal to 0 otherwise. The i-th input is composed of an open loop and a compensation term, and is parameterized as follows:

$$u_i(k) = \gamma_{k,i}(\beta) + \sum_{t=0}^{k}\sum_{j=0}^{m} \theta_{k,i}^{t,j}(\beta)\ u_j(t), \tag{15}$$

where $\gamma_{k,i}(\beta) \in \mathbb{R}$ is a constant that represents the open loop component, whereas $\theta_{k,i}^{t,j}(\beta) \in \mathbb{R}$, $t = 0, \ldots, k$, $j = 1, \ldots, m$, are the compensation coefficients that define the dependency of $u_i(k)$ on the inputs samples $u_j(t)$, $t = 0, \ldots, k$, $j = 1, \ldots, m$.

Remark 2. *The input parametrization* (15) *recalls the one proposed in [24]. There is however a main difference: here the input does not depend on an exogenous signal acting on the system, but instead on a subset of the inputs – the set composed of the non-influential inputs – which is not known a priori and whose cardinality has to be maximized.*

Note that in (15) both the compensation and open loop terms depend on β. This is needed to force input u_i to either compensate the non-influential inputs (if $\beta_i = 0$) or be independent of them (if $\beta_i = 1$). In particular, we introduce the following constraints on the dependence of the compensation coefficients $\theta_{k,i}^{t,j}$ and the open loop coefficient $\gamma_{k,i}$ from β:

- $t < k$

$$\left.\begin{array}{l} -(1-\beta_i)V \leq \theta_{k,i}^{t,j} \leq (1-\beta_i)V \\ -\beta_j V \leq \theta_{k,i}^{t,j} \leq \beta_j V \end{array}\right\} \quad \text{if} \quad j \neq i, \tag{16}$$

$$\theta_{k,i}^{t,j} = 0 \qquad\qquad\qquad\qquad \text{if} \quad j = i$$

- $t = k$

$$\theta_{k,i}^{t,j} = 0 \qquad \text{if} \quad j \neq i, \tag{17}$$

$$\theta_{k,i}^{t,j} = \beta_i \qquad \text{if} \quad j = i$$

$$-(1-\beta_i)V \leq \gamma_{k,i} \leq (1-\beta_i)V, \tag{18}$$

where V is a constant large enough, which can be set based on the fact that all involved inputs are bounded (see [34]).
It is easy to verify that constraints (16), (17) and (18) jointly guarantee that: *i)* if input u_i is non-influential ($\beta_i = 1$), then, the compensation term and the open loop term in (15) vanish (see constraints (16) and (18)); *ii)* if input u_i is influential ($\beta_i = 0$), its value is determined by a linear combination of the past values of the non-influential inputs (those inputs u_j with $\beta_j = 1$), and the open loop term. None of the influential inputs depends either on any other influential input or on a synchronous value of the non-influential inputs; and *iii)* if all inputs are influential, i.e. $\beta_i = 0$, $i = 1, \dots, m$, no compensation will be performed. Finally, additional constraints are introduced to enforce the parameterized inputs to belong to their corresponding bounded interval:

$$\begin{aligned} &\underline{u}_i \leq \gamma_{k,i}(\beta) + \sum_{t=0}^{k}\sum_{j=0}^{m} \theta_{k,i}^{t,j}(\beta) u_j(t) \leq \overline{u}_i, \\ &k = 0, \dots, T-1, \; i = 1, \dots, m. \end{aligned} \tag{19}$$

Note that all the constraints introduced so far are linear in the design parameters β, $\theta_{k,i}^{t,j}$, and $\gamma_{k,i}$.

2) Reformulation as a MILP problem: Problem (14) can be formulated as:

$$\begin{aligned} &\max_{\{\beta_i,\ \gamma_{k,i},\ \theta_{k,i}^{t,j}\}_{i,j\in\{1,\dots,m\}}^{k,t\in\{0,\dots,T-1\}}} \sum_{i=0}^{m} \beta_i \qquad (20) \\ &H_a x(T) \leq H_b \\ &x(k+1) = Ax(k) + Bu(k) \\ &u_i(k) = \gamma_{k,i}(\beta) + \sum_{t=0}^{k}\sum_{j=0}^{m} \theta_{k,i}^{t,j}(\beta)\ u_j(t) \\ &\text{Constraints} \quad (16),\ (17),\ (18),\ (19) \\ &\forall u(k) \in \mathcal{U}, \quad k = 0, \dots, T-1, \end{aligned}$$

where we use the polytopic description or inner approximation (6) of the target set $\mathcal{X}_f$, and the input parametrization in (15).

The optimization problem (20) is a robust MILP problem, since $x(T)$ appearing in the reachability constraint and the constraints (16), (17), (18), (19) are linear as a function of the optimization variables, and all constraints have to hold for any possible value taken by the inputs. Note that imposing the satisfaction of the constraints for any possible value taken by all the inputs, both the influential and the non-influential ones, is consistent with the original problem formulation (14), since the input parametrization is constructed in such a way so that only the non-influential inputs are compensated for and hence appear in the definition of the robust constraints.

Problem (20) is a semi-infinite problem and hence it is in general not easy to solve. However, we can exploit the particular structure of (20) to find a solution with a limited computational effort. To this end, observe first that the input parametrization in (15) can be written in the compact form

$$U = \Theta U + \Gamma, \tag{21}$$

where $\Gamma = [\gamma'(0), \dots, \gamma'(T-1)]'$, and matrix

$$\Theta = \begin{bmatrix} \theta_0^0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ \theta_{T-1}^0 & \dots & \dots & \theta_{T-1}^{T-1} \end{bmatrix} \text{with } \theta_k^t = \begin{bmatrix} \theta_{k,1}^{t,1} & \dots & 0 \\ \vdots & \dots & \vdots \\ \theta_{k,m}^{t,1} & \dots & \theta_{k,m}^{t,m} \end{bmatrix}$$

has a lower triangular structure. Constraints in (19) can then be rewritten as $\underline{U} \leq \Theta U + \Gamma \leq \bar{U}$, where $\underline{U} = [\underline{u}', \dots, \underline{u}']'$ and $\bar{U} = [\overline{u}', \dots, \overline{u}']'$. In turn, by plugging (21) in (7) we obtain $x(T) = A^T x_0 + \mathbf{B}_T(\Theta U + \Gamma)$.
As a result, by properly rearranging terms, problem (20) can be rewritten as follows:

$$\max_{\beta,\ \Theta,\ \Gamma} \quad \mathbf{1}'_m \beta \tag{22}$$

$$Z_a U \leq Z_b, \qquad \forall U \in [\underline{U}, \bar{U}] \tag{23}$$

$$\text{Constraints} \quad (16),\ (17),\ (18)$$

where Z_a, Z_b have the following form:

$$Z_a = \begin{bmatrix} H_a \mathbf{B}_T \Theta \\ \Theta \\ -\Theta \end{bmatrix}, \; Z_b = \begin{bmatrix} H_b - H_a A^T x_0 - H_a \mathbf{B}_T \Gamma \\ \bar{U} - \Gamma \\ -\underline{U} + \Gamma \end{bmatrix}. \tag{24}$$

We can now prove the following proposition:

Proposition III.1. *Let* $M = \frac{\bar{U}+\underline{U}}{2}$ *and* $F = \frac{\bar{U}-\underline{U}}{2}$.
The robust MILP problem (22) *is equivalent to the standard MILP problem:*

$$\begin{aligned} &\max_{\beta,\ \Theta,\ \Gamma} \quad \mathbf{1}'_m \beta \\ &\Psi + Z_a M \leq Z_b \qquad (25) \\ &\textit{Constraints} \quad (16),\ (17),\ (18) \end{aligned}$$

where Ψ *is defined as a vector with* i*-th element given by the* F*-weighted* ℓ_1*-norm of the* i*-th row of* Z_a*, that is* $\Psi_i = \sum_{j=1}^{mT} \left|Z_{a_{i,j}} F_j\right|$.

Proof: By introducing an auxiliary variable $W \in \mathcal{W} = [-F,\ F]$ and setting $U = M + W$, the robust constraint (23) can be equivalently rewritten as:

$$\max_{W\in[-F,\ F]} Z_a W + Z_a M \leq Z_b. \tag{26}$$

The worst case value of the term on the left-hand-side in (26) is attained on a vertex of $\mathcal{W}$ and, since each element of F is positive, its value is given row-by-row by:

$$\max_{W\in[-F,\ F]} [Z_a]_i\, W = \|\, [Z_a]_i \operatorname{diag}(F)\|_1 = \|[Z_a]_i\|_{1,F} = \Psi_i,$$

where $\mathrm{diag}(F)$ is the square diagonal matrix with vector F on the diagonal. ■

Note that with respect to the approach described in Section II, β is a binary-valued vector and the additional information on the size of the range on which an input is influential is lost. Indeed, the parametrization (15) determine a unique value for the influential inputs, which are interpreted as set-valued signals as in Section II. However, by introducing the compensation scheme we can obtain more non-influential inputs than that with the open loop term only. This is shown in Section **??**, where the compensation and open loop schemes are compared on a benchmark numerical example.

B. Comparison with the open loop scheme

As for the number of detected non-influential inputs, the technique proposed in this section leads to better results than the one proposed in Section II. Given the same specification, denote with J_2^* the optimal solution of (25), and with J_1^* the optimal solution of (12) obtained by restricting the β variables to be binary so that J_2^* and J_1^* represent the number of non-influential inputs detected by the compensation scheme approach, and the open loop scheme approach, respectively.

Proposition III.2. $J_2^* \geq J_1^*$

Proof: If we restrict the β variables in (12) to be binary, then problems (25) and (12) share the same constraints and cost function, and differ just in terms of the adopted input parametrization. The proposition is then easily proved by noticing that the input parametrization adopted in (25) is richer than the one in (12), so that the optimal solution of the latter is always feasible for the former. ■

Note that the compensation scheme gives better performances than the open loop scheme but to the expense of solving a MILP instead of an LP problem, which is generally more computationally intensive.

IV. Possible extensions

Both the techniques proposed in Sections II and III easily extend to the case when system (1) is time-varying. It suffices to replace matrices A^T and $\mathbf{B}_T$ in (12) and (24) with $\prod_{k=0}^{T-1} A^{(k)}$ and $\left[\prod_{k=1}^{T-1} A^{(k)} B^{(0)} \prod_{k=2}^{T-1} A^{(k)} B^{(1)} \dots B^{(T-1)}\right]$, where $A^{(i)}$ and $B^{(i)}$ denote the matrices of the system at time i.
Interestingly, other specifications besides reachability can be handled. In particular, the same methodology can be used in the case of safety specifications where the entire state evolution of the system has to be confined within some possibly time-varying polyhedral safe set described by $H_{a,k}x(k) \leq H_{b,k}$, $k = 1, \dots, T$. In this case, one just needs to replace $\mathbf{B}_T$ with $\mathbf{B}$, A^T with $[A', \dots, A^{T'}]'$, and H_a with $\mathrm{diag}([H_{a,1}, \dots, H_{a,T}])$ in (20) and (12), and all results retain their validity. Clearly, analogous slight modifications can be used if the specification involves some output or some linear combination of state and output. Moreover, the approach can be applied to any specification given by some propositional logic formula composed by linear clauses on the state or the output (like, e.g., a reachability specification with two distinct target sets). To this purpose, one just need to translate the formula into a set of mixed integer inequalities (see [34]).
Finally, the approach can handle the case when the initial condition is uncertain and belongs to a box. One has simply to add a set-valued signal to account for its variability and incorporate this signal into the robust optimization problem.

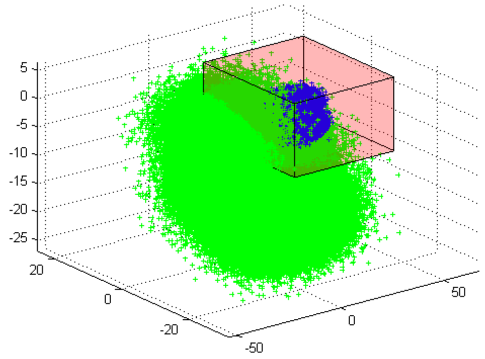

Fig. 1. Open loop scheme: target set $\mathcal{X}_f$ (red) and estimates of maximal reachable set (green) and reachable set with the designed input ranges (blue).

V. Numerical examples

We next present two examples and compare the performance achieved with the techniques presented in Sections II and III. In the both cases, we used CPLEX [33] over YALMIP [35].

1) Synthetic example: Consider system (13) where $x \in \mathbb{R}^3$ is subject to six scalar inputs ($m = 6$) taking values in the following intervals: $[\underline{u}_1,\ \overline{u}_1] = [0.5,\ 1.5]$, $[\underline{u}_2,\ \overline{u}_2] = [-1,\ 1]$, $[\underline{u}_3,\ \overline{u}_3] = [-1,\ -0.5]$, $[\underline{u}_4,\ \overline{u}_4] = [-0.5,\ 0.2]$, $[\underline{u}_5,\ \overline{u}_5] = [-2,\ 3]$, $[\underline{u}_6,\ \overline{u}_6] = [1,\ 4]$, and matrices A and B are given by:

$$A = \begin{bmatrix} 0.7 & 0.9 & 0.5 \\ -0.2 & 0.8 & -0.1 \\ 0 & 0.2 & 0.1 \end{bmatrix},\ B = \begin{bmatrix} 5 & 1 & 1 & 1 & 2 & 1 \\ 1 & 2 & -3 & 2 & -1 & -1 \\ 0.1 & 0 & 1 & 4 & -1 & -3 \end{bmatrix}.$$

Let I_k denote the identity matrix of dimension k. State x has to be steered to the target set

$$\mathcal{X}_f = \left\{x \in \mathbb{R}^3 : \begin{bmatrix} I_3 \\ -I_3 \end{bmatrix} x \leq \begin{bmatrix} 50 & 10 & 5 & -2 & 17 & 8 \end{bmatrix}' \right\},$$

at time $T = 50$, starting from $x(0) = [1\ 1\ 1]'$.

The open loop scheme of Section II has been applied by solving (12), and the following optimal values for the β's parameters have been obtained: $\beta_1^* = 0.33$, $\beta_2^* = 0$, $\beta_3^* = 1$, $\beta_4^* = 1$, $\beta_5^* = 0$, $\beta_6^* = 0$. This means that u_3 and u_4 are non-influential inputs ($\beta_3^* = \beta_4^* = 1$), whereas inputs u_2, u_5 and u_6 have to be precisely set to the corresponding optimal sequences $\{u_{\beta,i}^*(k)\}_{k=0}^{T-1}$, $i = 2,\ 5,\ 6$ since $\beta_2^* = \beta_5^* = \beta_6^* = 0$. As for input u_1, it can be set on a range $R_1^*(k)$, $k = 0, \dots, T-1$, whose size is one third of its entire admissible range ($\beta_1^* = 0.33$) and is specified by $\{u_{\beta,1}^*(k)\}_{k=0}^{T-1}$ and β_1^*. Values are not reported here, due to space limitations.

Figure 1 represents the target set $\mathcal{X}_f$ and an approximation of the maximal reachable set, i.e., the set of points that can be reached at time T by applying all input sequences in $\mathcal{U}^T$ from $x(0) = [1\ 1\ 1]'$. The approximation is obtained by first gridding $\mathcal{U}^T$ and then determining $x(T)$ for some randomly chosen grid points. Indeed, considering all points of the input grid would lead to too many points plotted and would not add any additional insight. The plot shows that just a small subset of the maximal reachable set is within $\mathcal{X}_f$, and in fact one has to limit the admissible range of some of the inputs so as to satisfy the reachability condition. As expected, the set of values $x(T)$ reached from $x(0) = [1\ 1\ 1]'$ when applying all input sequences in the computed optimal ranges is contained within $\mathcal{X}_f$ (see Figure 1). Not surprisingly, the more the target set become tighter, the more the inputs become influential. This is confirmed by the results obtained with the (smaller) target set

$$\mathcal{X}_f = \left\{x \in \mathbb{R}^3 : \begin{bmatrix} I_3 \\ -I_3 \end{bmatrix} x \leq \begin{bmatrix}50 & 10 & 5 & -2 & 17 & 0\end{bmatrix}'\right\}.$$

In this case, the solution to (12) is $\beta_1^* = 0.35$, $\beta_2^* = 0$, $\beta_3^* = 1$, $\beta_4^* = 0$, $\beta_5^* = 0$, $\beta_6^* = 0$, so that input u_4 becomes now influential. Figure 2 represents the set of values $x(T)$ reached from $x(0) = [1\ 1\ 1]'$ when applying the admissible input sequences of the computed optimal solution. Computation were performed on a laptop equipped with an Intel Core i5 2.4 GHz processor and took less than 1 second for both cases. In this example, the compensation scheme returns the same number of non-influential inputs as the open loop scheme.

2) Benchmark example: In [36], the non-linear fourth order model

$$\begin{aligned}
\dot{h}_1(t) &= -\frac{a_1}{A_1}\sqrt{2gh_1(t)} + \frac{a_3}{A_1}\sqrt{2gh_3(t)} + \frac{\gamma_1 k_1}{A_1}v_1(t) \\
\dot{h}_2(t) &= -\frac{a_2}{A_2}\sqrt{2gh_2(t)} + \frac{a_4}{A_2}\sqrt{2gh_4(t)} + \frac{\gamma_2 k_2}{A_2}v_2(t) \\
\dot{h}_3(t) &= -\frac{a_3}{A_3}\sqrt{2gh_3(t)} + \frac{(1-\gamma_2)k_2}{A_3}v_2(t) \\
\dot{h}_4(t) &= -\frac{a_4}{A_4}\sqrt{2gh_4(t)} + \frac{(1-\gamma_1)k_1}{A_4}v_1(t)
\end{aligned} \tag{27}$$

of a quadruple tank system is considered, and the problem of keeping the tanks levels $h = [h_1\ h_2\ h_3\ h_4]'$ below a given threshold value by acting on two pumps through the input $v = [v_1\ v_2]'$ is addressed. The model is linearized in correspondence of two operating points P_- and P_+ obtained by feeding the non-linear system (27) with the constant inputs $\bar{v}_- = [3,\ 3]'$ and $\bar{v}_+ = [3.15,\ 3.15]'$, respectively. In the linearized dynamics $\dot{x}(t) = A_c x(t) + B_c u(t)$, x and $u = [u_1\ u_2]'$ represent the difference between h, v and their corresponding equilibrium values, and matrices A_c and B_c are given by

$$A_c = \begin{bmatrix} -\frac{1}{\tau_1} & 0 & \frac{a_3}{a_1 \tau_3} & 0 \\ 0 & -\frac{1}{\tau_2} & 0 & \frac{a_4}{a_2 \tau_4} \\ 0 & 0 & -\frac{1}{\tau_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau_4} \end{bmatrix} \quad B_c = \begin{bmatrix} \frac{\gamma_1 k_1}{a_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{a_2} \\ 0 & \frac{(1-\gamma_2)k_2}{a_3} \\ \frac{(1-\gamma_1)k_1}{a_4} & 0 \end{bmatrix},$$

whose coefficients are reported in Table I for both the operating points P_- and P_+.

The linearized system is discretized in time by applying a constant input over each sampling interval of duration $T_s = 4$.

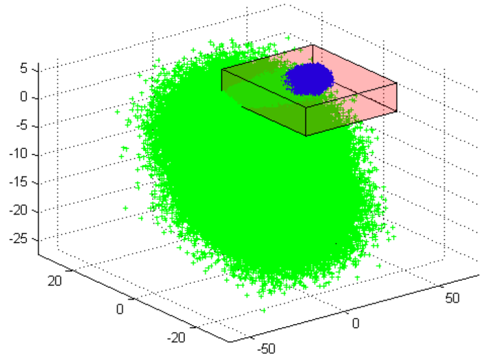

Fig. 2. Tighter specification: target set $\mathcal{X}_f$ (red) and estimates of maximal reachable set (green) and reachable set with the designed input ranges (blue).

	τ_1	τ_2	τ_3	τ_4	$a_{1,3}$	$a_{2,4}$	γ_1	γ_2	k_1	k_2
P_-	62	90	23	30	28	32	0.7	0.6	3.33	3.35
P_+	63	91	39	56	28	32	0.43	0.34	3.14	3.29

TABLE I
COEFFICIENTS OF THE LINEARIZED MODELS

The safety specification is given by $x(k) \in \mathcal{X}_f$, $k = 0, \ldots, T$, where $\mathcal{X}_f := \left\{x \in \mathbb{R}^4 : \|x\|_\infty \leq 5\right\}$ and $T = 50$. Both the inputs of the linearized system are bounded in the range $[-1.5,\ 1.5]$. The open loop scheme for non-influential inputs detection described in Section II provides $\beta^* = [0,\ 0]'$ as solution in both the P_- and P_+ cases, so that both u_1 and u_2 are influential. This means that there exists no open loop sequence for u_1 that keeps the state in the safe set $\mathcal{X}_f$ within $[0, T]$ for any possible behavior of u_2, and vice versa. Instead, the compensation scheme in Section III leads to $\beta^* = [1,\ 0]'$ for both the linearized models, meaning that if we set u_2 to the value given by the optimal parametrization Θ^* and Γ^*, u_1 is non-influential. This is shown in Figure 3 for the operating point P_- where we set input u_1 to a specific realization that would steer the state of the system outside the safe set if the open loop solution for u_2 were applied, as shown in Figure 4. Computations took 1330 seconds (1155 to write the constraints in YALMIP and 175 to solve the MILP via CPLEX).

VI. Conclusions

We have addressed the problem of detecting non-influential inputs, i.e. those inputs that do not need to be properly set and can take an arbitrarily value in their range while guaranteeing the satisfaction of some reachability/safety specification. Effective solutions have been developed for the class of linear systems. Two different methods have been developed, which rely on different input parameterizations. In the first one, an open loop sequence for the influential inputs that makes the other inputs non-influential is designed. In the second one, influential inputs are allowed to depend on the non-influential ones, so as to partially compensate for them. The compensation method is amenable for application to networked control systems security. This requires further investigation.

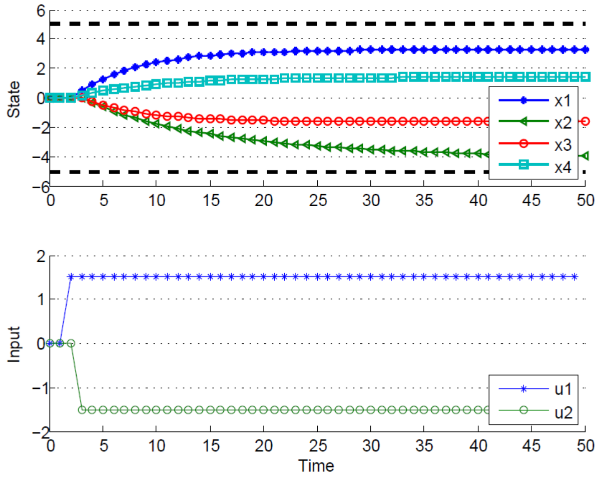

Fig. 3. Operating point P_{-}: input u_2 compensates for input u_1.

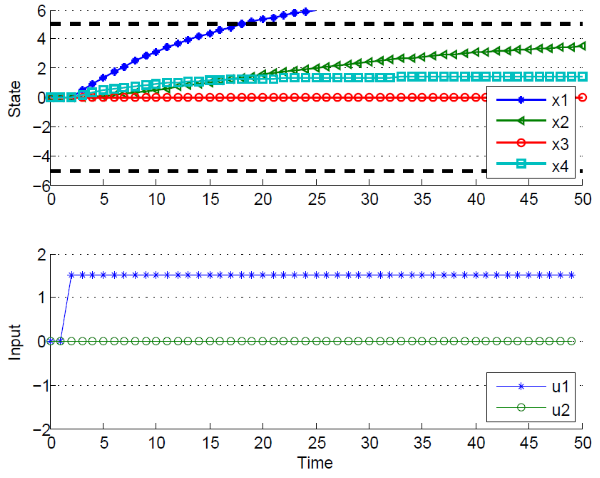

Fig. 4. Operating point P_{-}: input u_1 steering the state of the system outside the safe set when input u_2 is set equal to zero.

References

[1] C. Baier and J.-P. Katoen, *Principles of model checking*. MIT press Cambridge, 2008.

[2] L. Aceto, A. Burgueno, and K. Larsen, *Model checking via reachability testing for timed automata*. Springer, 1998.

[3] E. Emerson, "Temporal and modal logic." *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, vol. 995, no. 1072, p. 5, 1990.

[4] G. Holzmann, "The model checker SPIN," *IEEE Transactions on software engineering*, no. 5, pp. 279–295, 1997.

[5] K. G. Larsen, P. Pettersson, and W. Yi, "UPPAAL in a nutshell," *International Journal on Software Tools for Technology Transfer (STTT)*, vol. 1, no. 1, pp. 134–152, 1997.

[6] A. Cimatti, E. Clarke, F. Giunchiglia, and M. Roveri, "NuSMV: A new symbolic model verifier," in *Computer Aided Verification*. Springer, 1999, pp. 495–499.

[7] G. Frehse, C. Le Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler, "Spaceex: Scalable verification of hybrid systems," in *Proc. 23rd International Conference on Computer Aided Verification (CAV)*, ser. LNCS, S. Q. Ganesh Gopalakrishnan, Ed. Springer, 2011.

[8] E. Asarin, O. Bournez, T. Dang, and O. Maler, "Approximate reachability analysis of piecewise-linear dynamical systems," in *Hybrid Systems: Computation and Control*. Springer, 2000, pp. 20–31.

[9] A. Bemporad and M. Morari, "Verification of hybrid systems via mathematical programming," in *Hybrid Systems: Computation and Control*. Springer, 1999, pp. 31–45.

[10] A. Girard, C. Le Guernic, and O. Maler, "Efficient computation of reachable sets of linear time-invariant systems with inputs," in *Hybrid Systems: Computation and Control*. Springer, 2006, pp. 257–271.

[11] A. B. Kurzhanski and P. Varaiya, "Ellipsoidal techniques for reachability analysis: internal approximation," *Systems & control letters*, vol. 41, no. 3, pp. 201–211, 2000.

[12] I. Mitchell and C. Tomlin, "Level set methods for computation in hybrid systems," in *Hybrid Systems: Computation and Control*. Springer, 2000, pp. 310–323.

[13] F. Torrisi, "Modeling and reach-set computation for analysis and optimal control of discrete hybrid automata," Ph.D. dissertation, Diss., Technische Wissenschaften ETH Zürich, Nr. 15064, 2003, 2003.

[14] R. Alur, T. Henzinger, G. Lafferriere, and G. Pappas, "Discrete abstractions of hybrid systems," *Proceedings of the IEEE*, vol. 88, no. 7, pp. 971–984, 2000.

[15] A. Girard and G. Pappas, "Approximation metrics for discrete and continuous systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 782–798, 2007.

[16] H. Kress-Gazit, G. Fainekos, and G. Pappas, "Temporal-logic-based reactive mission and motion planning," *IEEE Transactions on Robotics*, vol. 25, no. 6, pp. 1370–1381, 2009.

[17] P. Tabuada and G. Pappas, "Linear time logic control of discrete-time linear systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 12, pp. 1862–1877, 2006.

[18] R. Alur, "Formal verification of hybrid systems," in *IEEE International Conference on Embedded Software*, 2011, pp. 273–278.

[19] G. Fainekos, S. Loizou, and G. Pappas, "Translating temporal logic to controller specifications," in *45th IEEE Conference on Decision and Control*, 2006, pp. 899–904.

[20] A. Bitlislioglu, T. T. Gorecki, and C. Jones, "Robust Tracking Commitment with Application to Demand Response," EPFL, Tech. Rep., 2015.

[21] X. Zhang, M. Kamgarpour, P. Goulart, and J. Lygeros, "Selling robustness margins: A framework for optimizing reserve capacities for linear systems," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 6419–6424.

[22] R. Vignali, L. Deori, and M. Prandini, "Control input design: detecting non influential inputs while satisfying a reachability specification," in *19th IFAC World Congress*, 2014.

[23] R. Vignali and M. Prandini, "Input design for a cascading system: An approach based on system decomposition and non-influential input detection," in *IEEE Multi-Conference on Systems and Control*, 2014.

[24] P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski, "Optimization over state feedback policies for robust control with constraints," *Automatica*, vol. 42, no. 4, pp. 523–533, 2006.

[25] A. A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems." in *HotSec*, 2008.

[26] Z. Huang, Y. Wang, S. Mitra, and G. E. Dullerud, "Controller synthesis for linear time-varying systems with adversaries," *CoRR*, vol. abs/1501.04925, 2015. [Online]. Available: http://arxiv.org/abs/1501.04925

[27] A. Teixeira, K. Sou, H. Sandberg, and K. Johansson, "Secure control systems: A quantitative risk management approach," *Control Systems, IEEE*, vol. 35, no. 1, pp. 24–45, 2015.

[28] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 877–905, 2008.

[29] A. Ben-Tal and A. Nemirovski, "Robust convex optimization," *Mathematics of Operations Research*, vol. 23, no. 4, pp. 769–805, 1998.

[30] ——, "Robust solutions of uncertain linear programs," *Operations research letters*, vol. 25, no. 1, pp. 1–13, 1999.

[31] ——, "On tractable approximations of uncertain linear matrix inequalities affected by interval uncertainty," *SIAM Journal on Optimization*, vol. 12, no. 3, pp. 811–833, 2002.

[32] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[33] I. ILOG, *11.0 CPLEX User's Manual*, 2007.

[34] A. Bemporad, G. Ferrari-Trecate, and M. Morari, "Observability and Controllability of Piecewise Affine and Hybrid Systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 10, pp. 1864–1876, 2000.

[35] J. Löfberg, "YALMIP: a toolbox for modeling and optimization in MATLAB," in *Proceedings of 13th IEEE Symposium on Computer Aided Control System Design*, 2004.

[36] K. H. Johansson, "The quadruple-tank process: A multivariable laboratory process with an adjustable zero," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 3, pp. 456–465, 2000.