

Control input design: detecting non influential inputs while satisfying a reachability specification^{*}

Riccardo Vignali^{*} Luca Deori^{*} Maria Prandini^{*}

^{*} *Dipartimento di Elettronica, Informazione e Bioingegneria,
Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano,
Italy. E-mail: {vignali,deori,prandini}@elet.polimi.it*

Abstract: We address the problem of designing the control input for a discrete time dynamical system so as to make its state reach some target set in finite time. Among the feasible solutions to the reachability problem, we look for those where only few input variables need to be set to some specific value, whereas the others can take an arbitrary value within their admissible range without compromising the desired reachability condition. This input design problem is not standard and the optimality criterion cannot be easily expressed in terms of some performance index to be optimized. Here, we propose a solution that rests on an appropriate parametrization of the input variables as set-valued signals, and rephrase the input design problem as a robust optimization program. In turn, if the target set is a polytope, the optimization problem reduces to a linear program for linear system, and to a mixed integer linear program for mixed logical dynamical systems. Some numerical examples show the efficacy of the approach.

1. INTRODUCTION

In this paper we address the problem of designing the control input for a discrete time dynamical system so as to make its state reach a target set in some finite time. This problem is well-studied in the literature and effective solutions have been proposed for various classes of systems, see e.g. Bemporad et al. [2000], Mitchell and Tomlin [2000], Tomlin et al. [2003], Chutinan and Krogh [2003], Girard et al. [2006], Kurzhanskiy and Varaiya [2007], Guernic and Girard [2010] to name a few. However, here we have an additional requirement: we look for a solution where only few input variables need to be set to some specific value, whereas the others can take an arbitrary value within their admissible range without compromising the desired reachability condition. For ease of reference, we shall call the former input variables *influential*, and the latter ones *non influential*. Interestingly, the non influential input variables remain free to be set and can eventually be used to satisfy further objectives besides reachability (*multi-objective optimization*).

The considered problem originates from a verification context, where, in order to test the correct functioning of a system S composed of multiple subsystems, one needs to apply some appropriate excitation signal to a subsystem \tilde{S} . In the case when \tilde{S} is affected by the inputs of S only indirectly through some state variables, the problem becomes that of imposing some desired behavior to those state variables, which can then be translated into a reachability condition for an enlarged system that embeds the system specification (see e.g. Fainekos et al. [2006], Fainekos et al. [2009]). Finding a solution to the reachability problem while maximizing the number of

non influential input variables can effectively simplify the testing phase, since having to set only a limited number of inputs can save time and reduce the effort.

Note that the described input design problem is not standard and the optimality criterion cannot be expressed easily in terms of some performance index to be optimized. In this paper, we propose a solution that rests on an appropriate parametrization of the input variables as set-valued signals, and show that this parametrization allows to reformulate the input design problem as a robust optimization program. In turn, if the target set is a polytope, the robust optimization program reduces to a Linear Programming (LP) problem for linear systems, and to a Mixed Integer LP (MILP) problem for the class of Mixed Logical Dynamical (MLD) systems introduced in Bemporad and Morari [1999], hence solutions can be effectively computed with LP and MILP solvers.

The rest of the paper unfolds as follows. In Section 2, we formulate the problem for a discrete time linear system and rephrase it as a robust optimization problem. A resolution methodology is then proposed that leads to an LP problem to be solved. Section 3 generalizes the approach to the class of MLD systems, leading to a MILP solution. Section 4 presents some numerical examples. Finally, some concluding remarks are drawn in Section 5.

2. PROBLEM FORMULATION AND RESOLUTION FOR A LINEAR SYSTEM

Consider a discrete time linear system

$$x(k+1) = Ax(k) + B_1 u_1(k) + \dots + B_m u_m(k), \quad (1)$$

where the evolution of the state $x \in \mathbb{R}^n$ is affected by m scalar control inputs u_i , $i = 1, \dots, m$, taking values within the intervals $[\underline{u}_i, \bar{u}_i]$, $i = 1, \dots, m$.

^{*} Research supported by the European Commission under the MoVeS project, FP7-ICT-2009-257005.

Our goal is to design the inputs u_i , $i = 1, \dots, m$ so as to steer the state x of the system to some convex set $\mathcal{X}_f \subset \mathbb{R}^n$ at time T , starting from a given initial condition $x(0) = x_0$. We look for a solution where only a limited number $p \leq m$ of inputs have to be set to some specific value in order to reach the target set \mathcal{X}_f at time T (*influential* inputs), whilst the other $m - p$ inputs can take arbitrarily values within their whole admissible range (*non influential* inputs).

To this purpose, we treat each input u_i as a set-valued signal whose range has to be maximized while satisfying the reachability specification.

More precisely, for each $i = 1, \dots, m$, we parameterize u_i as follows

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

$k = 0, \dots, T - 1$, where w_i is a set-valued auxiliary signal taking values in $[-1, 1]$, whereas β_i and \tilde{u}_i are optimization variables to be set. In particular, $\beta_i \in [0, 1]$ is a scalar parameter and \tilde{u}_i is a single-valued signal taking values in $[\underline{u}_i, \bar{u}_i]$.

Note that the range $I_i(k) \subseteq [\underline{u}_i, \bar{u}_i]$ for $u_i(k)$ is jointly determined by β_i and $\tilde{u}_i(k)$ as follows

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

and collapses to the singleton $[\tilde{u}_i(k), \tilde{u}_i(k)]$ when $\beta_i = 0$, whereas it coincides with the whole interval $[\underline{u}_i, \bar{u}_i]$ when $\beta_i = 1$. Figure 1 depicts the case when $\beta_i = 0.5$.

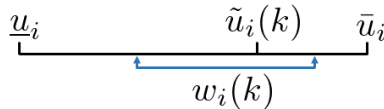


Fig. 1. Range $I_i(k)$ for $u_i(k)$ for a given $\tilde{u}_i(k)$ when $\beta_i = 0.5$.

The goal of identify the minimum number p of influential inputs, while setting an appropriate value for them at the same time, can be pursued by solving the following robust optimization problem:

$$\max_{\{\beta_i \in [0, 1], \tilde{u}_i(k) \in [\underline{u}_i, \bar{u}_i], k=0, \dots, T-1\}_{i=1}^m} |\{i : \beta_i = 1\}| \quad (3)$$

subject to:

$$x(T) \in \mathcal{X}_f$$

$$x(k+1) = Ax(k) + B_1 u_1(k) + \dots + B_m u_m(k)$$

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

$$\forall w_i(k) \in [-1, 1], \quad i = 1, \dots, m, \quad k = 0, \dots, T - 1$$

where $|\mathcal{C}|$ denotes the cardinality of set \mathcal{C} and we are actually maximizing the number of β_i 's that are equal to 1.

Due to the non-convex nature of the cost function and the bilinear term appearing in the parametrization of the input u_i (see (2)), computing a solution to the robust optimization problem (3) is difficult, in general.

We next propose an approach to reduce (3) to a robust convex optimization problem that can be solved efficiently if \mathcal{X}_f is a polytope described by a set of linear inequalities:

$$\mathcal{X}_f = \{x \in \mathbb{R}^n : H_a x \leq H_b\}. \quad (4)$$

The same approach can be applied if an inner approximation of \mathcal{X}_f via a polytope is adopted.

We first reparameterize u_i in (2) by replacing the bilinear term $(1 - \beta_i)\tilde{u}_i(k)$ with $u_{\beta,i}(k)$ taking values in $[(1 - \beta_i)\underline{u}_i, (1 - \beta_i)\bar{u}_i]$, thus obtaining

$$u_i(k) = u_{\beta,i}(k) + \frac{\underline{u}_i + \bar{u}_i}{2}\beta_i + \frac{\bar{u}_i - \underline{u}_i}{2}\beta_i w_i(k). \quad (5)$$

Accordingly, the range $I(k)$ of $u_i(k)$ can be expressed as

$$I_i(k) = [u_{\beta,i}(k) + \beta_i \underline{u}_i, u_{\beta,i}(k) + \beta_i \bar{u}_i], \quad (6)$$

and $\tilde{u}_i(k)$ can be recovered through:

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

As for the cost function, maximizing $|\{i : \beta_i = 1\}|$ is equivalent to enhancing the sparsity of vector $\gamma = [\gamma_1 \ \gamma_2 \ \dots \ \gamma_m]'$ with $\gamma_i = 1 - \beta_i$ by minimizing its ℓ_0 -norm, i.e., the number of its non-zero elements. Since the ℓ_0 -norm is non-convex, we shall replace it with the ℓ_1 -norm $\|\gamma\|_1 = \sum_{i=1}^m |1 - \beta_i|$, as suggested in Candes et al. [2008] where a discussion on possible variants with a reweighting of the ℓ_1 -norm to better approximate the ℓ_0 -norm is also provided. Given that $\beta_i \in [0, 1]$, then, $\|\gamma\|_1 = m - \sum_{i=1}^m \beta_i$, and minimizing $\|\gamma\|_1$ is equivalent to maximizing the function $\sum_{i=1}^m \beta_i$, which is convex.

This leads to the robust optimization program

$$\max_{\{\beta_i \in [0, 1], u_{\beta,i}(k), k=0, \dots, T-1\}_{i=1}^m} \sum_{i=1}^m \beta_i \quad (7)$$

subject to:

$$x(T) \in \mathcal{X}_f$$

$$x(k+1) = Ax(k) + B_1 u_1(k) + \dots + B_m u_m(k)$$

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

$$(1 - \beta_i)\underline{u}_i \leq u_{\beta,i}(k) \leq (1 - \beta_i)\bar{u}_i$$

$$\forall w_i(k) \in [-1, 1], \quad i = 1, \dots, m, \quad k = 0, \dots, T - 1,$$

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

We next show how (7) can be solved efficiently when \mathcal{X}_f is a polytope (see (4)). We first need to introduce some notations.

Let us define vectors

$$u(k) = [u_1(k), u_2(k), \dots, u_m(k)]',$$

$$u_{\beta}(k) = [u_{\beta,1}(k), u_{\beta,2}(k), \dots, u_{\beta,m}(k)]',$$

$$w(k) = [w_1(k), w_2(k), \dots, w_m(k)]',$$

$$\beta = [\beta_1, \beta_2, \dots, \beta_m]'$$

and

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

Then, the state at time T can be expressed as

$$x(T) = A^T x_0 + \mathbf{B}_T U, \quad (8)$$

where

$$\mathbf{B}_T = [A^{T-1}B \ A^{T-2}B \ \dots \ B] \quad (9)$$

with $B = [B_1 \ B_2 \ \dots \ B_m]$. Also, equation (5) can be written in the following compact form:

$$U = U_\beta + (D_W + S)\beta, \quad (10)$$

where we set

$$D_W = \begin{bmatrix} \text{diag}([d_1 w_1(0), \dots, d_m w_m(0)]) \\ \vdots \\ \text{diag}([d_1 w_1(T-1), \dots, d_m w_m(T-1)]) \end{bmatrix},$$

$$S = \begin{bmatrix} \text{diag}([s_1, \dots, s_m]) \\ \vdots \\ \text{diag}([s_1, \dots, s_m]) \end{bmatrix},$$

with $d_i = \frac{\bar{u}_i - u_i}{2}$ and $s_i = \frac{\bar{u}_i + u_i}{2}$, $i = 1, \dots, m$.

By plugging (10) into (8), we finally get

$$x(T) = A^T x_0 + \mathbf{B}_T U_\beta + \mathbf{B}_T (D_W + S)\beta. \quad (11)$$

As a result, the constraint $x(T) \in \mathcal{X}_f$ with \mathcal{X}_f given in (4) becomes:

$$[H_a \mathbf{B}_T \ H_a \mathbf{B}_T (D_W + S)] \begin{bmatrix} U_\beta \\ \beta \end{bmatrix} \leq H_b - H_a A^T x_0 \quad (12)$$

and the optimization problem (7) rewrites as

$$\max_{\{\underline{U}(1-\beta) \leq U_\beta \leq \bar{U}(1-\beta), \beta \in [0,1]^m\}} c' \beta \quad (13)$$

subject to:

$$[H_a \mathbf{B}_T \ H_a \mathbf{B}_T (D_W + S)] \begin{bmatrix} U_\beta \\ \beta \end{bmatrix} \leq H_b - H_a A^T x_0,$$

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

where $c = [1 \ 1 \ \dots \ 1]'$,

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

Note that the optimization problem (13) is linear but semi-infinite since the number of optimization variables is finite but the number of constraints is infinite given that constraint (12) has to be satisfied for any realization of the set-valued signal W . Finding a solution to a semi-infinite optimization problem is difficult in general, see e.g. Boyd and Vandenberghe [2004], Ben-Tal and Nemirovski [1998], Ben-Tal and Nemirovski [1999], Ben-Tal and Nemirovski [2002]. However, we can find the exact optimal solution of (13) with a limited computational effort thanks to its particular structure. This is explained in the following proposition.

Proposition 1. The solution of (13) can be obtained by solving the following LP program

$$\max_{\{\underline{U}(1-\beta) \leq U_\beta \leq \bar{U}(1-\beta), \beta \in [0,1]^m\}} c' \beta \quad (14)$$

subject to:

$$[H_a \mathbf{B}_T \ H_a \mathbf{B}_T S + \Xi] \begin{bmatrix} U_\beta \\ \beta \end{bmatrix} \leq H_b - H_a A^T x_0,$$

where

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

Proof

Consider the i -th row of constraint (12) given by

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

Note that the set-valued signal W enters only the left-hand-side of this inequality through the term $(H_a \mathbf{B}_T D_W)_i \beta$. The worst case is then achieved for those values of W that maximize the term $(H_a \mathbf{B}_T D_W)_i \beta$. Observe also that β is non-negative, and that each entry $(H_a \mathbf{B}_T D_W)_{ij}$ depends on $w_j(0), w_j(1), \dots, w_j(T-1)$ only, i.e., each j -th entry of the same row i depends only on the values within $[0, T-1]$ of the j -th component of w . Exploiting these two facts, we can just focus on independently maximizing each j -th entry of $(H_a \mathbf{B}_T D_W)_i$. In turn, since $(H_a \mathbf{B}_T D_W)_{ij}$ depends linearly on $w_j(0), w_j(1), \dots, w_j(T-1)$, we can write

$$(H_a \mathbf{B}_T D_W)_{ij} = \xi'_{ij} \begin{bmatrix} w_j(0) \\ \vdots \\ w_j(T-1) \end{bmatrix},$$

where

$\xi_{ij} = d_j [(H_a \mathbf{B}_T)_{ij}, (H_a \mathbf{B}_T)_{ij+m}, \dots, (H_a \mathbf{B}_T)_{ij+m(T-1)}]'$. Recalling that $w_j(k) \in [-1, 1], \forall k = 0, \dots, T-1$, we finally have

$$\max_{W \in [-1, 1]^{mT}} (H_a \mathbf{B}_T D_W)_{ij} = \|\xi_{ij}\|_1,$$

which concludes the proof. \square

Problem (14) is a standard LP program with a finite number of constraints, and, hence, can be easily solved by means of standard solvers like CPLEX [2007].

3. EXTENSION TO MIXED LOGICAL DYNAMICAL SYSTEMS

In this section we consider Mixed Logical Dynamical (MLD) systems of the form:

$$x(k+1) = Ax(k) + B_u u(k) + B_\delta \delta(k) + B_z z(k) \quad (15a)$$

$$E_2 \delta(k) + E_3 z(k) \leq E_1 u(k) + E_4 x(k) + E_5 \quad (15b)$$

where $x \in \mathbb{R}^{n_c} \times \{0, 1\}^{n_l}$ represents the state of the system and is composed of both continuous and binary variables, $u = [u_1, u_2, \dots, u_m]'$ $\in [\underline{u}_1, \bar{u}_1] \times [\underline{u}_2, \bar{u}_2] \times \dots \times [\underline{u}_m, \bar{u}_m]$ are the control inputs, whereas $\delta \in \{0, 1\}^{r_l}$ and $z \in \mathbb{R}^{r_c}$ are auxiliary variables.

By adopting the parametrization for u_i in (2) and following the same approach as in Section 2, we can reformulate the problem of designing the inputs so as to reach some target set \mathcal{X}_f at time T while maximizing the number of non influential inputs as the following robust optimization problem:

$$\max_{\{\underline{U}(1-\beta) \leq U_\beta \leq \bar{U}(1-\beta), \beta \in [0,1]^m, \Delta \in \{0,1\}^{r_l T}, Z\}} c' \beta \quad (16)$$

subject to:

$$\begin{bmatrix} H_a \mathbf{B}_u T & H_a \mathbf{B}_\delta T & H_a \mathbf{B}_z T \\ -\mathbf{E}_1 - \mathbf{E}_4 \mathbf{B}_u & \mathbf{E}_2 - \mathbf{E}_4 \mathbf{B}_\delta & \mathbf{E}_3 - \mathbf{E}_4 \mathbf{B}_z \end{bmatrix}$$

$$\begin{bmatrix} H_a \mathbf{B}_u T (D_W + S) \\ -(\mathbf{E}_1 + \mathbf{E}_4 \mathbf{B}_u) (D_W + S) \end{bmatrix} \begin{bmatrix} U_\beta \\ \Delta \\ Z \\ \beta \end{bmatrix} \leq \begin{bmatrix} H_b - H_a A^T x_0 \\ \mathbf{E}_4 \mathbf{A} x_0 + \mathbf{E}_5 \end{bmatrix}$$

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

Matrices $\mathbf{B}_u T$, $\mathbf{B}_\delta T$, and $\mathbf{B}_z T$ have the same structure as matrix (9) with B replaced by B_u , B_δ , and B_z ,

respectively, $\mathbf{E}_i = \text{diag}([E_i, \dots, E_i])$, $i = 1, 2, 3, 4$, $\mathbf{E}_5 = [E_5, E_5, \dots, E_5]'$, and \mathbf{A} , \mathbf{B}_u , \mathbf{B}_δ , \mathbf{B}_z , Δ , Z are defined as follow:

$$\mathbf{A} = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^{T-1} \end{bmatrix}, \quad \mathbf{B}_i = \begin{bmatrix} 0 & \dots & 0 \\ B_i & 0 & \\ AB_i & B_i & \vdots \\ \vdots & \ddots & \\ A^{T-2}B_i & \dots & B_i & 0 \end{bmatrix}, i = u, \delta, z,$$

$$\Delta = \begin{bmatrix} \delta(0) \\ \delta(1) \\ \vdots \\ \delta(T-1) \end{bmatrix}, \quad Z = \begin{bmatrix} z(0) \\ z(1) \\ \vdots \\ z(T-1) \end{bmatrix}.$$

Whilst (13) is a LP, problem (16) is a MILP because of the presence of the binary variables δ 's, and is semi-infinite because constraints have to be satisfied for every possible realization of W within $[-1, 1]^{mT}$. Note that in problem (16) there is an additional constraint deriving from (15b). Constraints in (16) have the same form of those in (13) and thus they can be dealt with by means of the same approach of Section 2. Indeed, note that only the terms $H_a \mathbf{B}_u D_W$ and $(-\mathbf{E}_1 - \mathbf{E}_4 \mathbf{B}_u) D_W$ depend on W , and each ij -th entry of these terms depends linearly on the j -th component of w only. Hence, adopting the same approach of Section 2 we can find the optimal solution of (16) by solving the following MILP problem with a finite number of constraints:

$$\begin{aligned} & \max_{\substack{\mathbf{U}(1-\beta) \leq U_\beta \leq \bar{\mathbf{U}}(1-\beta), \beta \in [0,1]^m, \Delta \in \{0,1\}^{r_1 T}, Z}} c' \beta \quad (17) \\ & \text{subject to:} \\ & \begin{bmatrix} H_a \mathbf{B}_u D_W & H_a \mathbf{B}_\delta D_W & H_a \mathbf{B}_z D_W \\ -\mathbf{E}_1 - \mathbf{E}_4 \mathbf{B}_u & \mathbf{E}_2 - \mathbf{E}_4 \mathbf{B}_\delta & \mathbf{E}_3 - \mathbf{E}_4 \mathbf{B}_z \\ H_a \mathbf{B}_u D_W S + \Xi \\ -(\mathbf{E}_1 + \mathbf{E}_4 \mathbf{B}_u) S + \Psi \end{bmatrix} \begin{bmatrix} U_\beta \\ \Delta \\ Z \\ \beta \end{bmatrix} \leq \begin{bmatrix} H_b - H_a A^T x_0 \\ \mathbf{E}_4 \mathbf{A} x_0 + \mathbf{E}_5 \end{bmatrix} \end{aligned}$$

where

$$\begin{aligned} (\Xi)_{ij} &= \|d_j [(H_a \mathbf{B}_u D_W)_{ij}, (H_a \mathbf{B}_u D_W)_{i, j+m}, \dots \\ & \quad \dots, (H_a \mathbf{B}_u D_W)_{i, j+m(T-1)}]\|_1 \\ (\Psi)_{ij} &= \|d_j [(-\mathbf{E}_1 - \mathbf{E}_4 \mathbf{B}_u)_{ij}, (-\mathbf{E}_1 - \mathbf{E}_4 \mathbf{B}_u)_{i, j+m}, \dots \\ & \quad \dots, (-\mathbf{E}_1 - \mathbf{E}_4 \mathbf{B}_u)_{i, j+m(T-1)}]\|_1. \end{aligned}$$

Problem (17) is a MILP with a finite number of constraints and, hence, can be solved by means of standard solvers like CPLEX [2007].

Note that the optimal value taken by the auxiliary variables Z^* and Δ^* does not change for any choice for the input $u(k)$ in the range defined by (U_β^*, β^*) according to (6).

4. NUMERICAL EXAMPLES

Consider the following linear system:

$$x(k+1) = \begin{bmatrix} 0.7 & 0.9 & 0.5 \\ -0.2 & 0.8 & -0.1 \\ 0 & 0.2 & 0.1 \end{bmatrix} x(k) + \begin{bmatrix} 5 \\ 1 \\ 0.1 \end{bmatrix} u_1(k) + \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} u_2(k) \quad (18)$$

where $x \in \mathbb{R}^3$, $u_1 \in [0.5, 1.5]$, and $u_2 \in [-1, 1]$.

Suppose that we want to steer the state x so as to reach the target set \mathcal{X}_f at time $T = 5$ starting from $x(0) = [1 \ 1 \ 1]'$. The target set \mathcal{X}_f is defined by the linear inequalities:

$$\begin{bmatrix} I_3 \\ -I_3 \end{bmatrix} x \leq [30 \ 0 \ 1 \ -5 \ 7 \ 0]'. \quad (19)$$

In order to solve problem (14) we use YALMIP Löfberg [2004] with CPLEX [2007] as LP solver, and obtain the following results:

$$\begin{aligned} \beta_1^* &= 1, & \beta_2^* &= 0.042 \\ u_{\beta,1}^*(k) &= 0, & k &= 0, \dots, 4 \\ u_{\beta,2}^*(k) &= \begin{cases} -0.957, & k = 0, 4 \\ 0.957 & k = 1, 2, 3. \end{cases} \end{aligned}$$

We obtain that the first input is non influential ($\beta_1^* = 1$) while the second input is influential and can vary in a quite small range being its corresponding β_2^* close to 0 ($\beta_2^* = 0.042$).

Figure 2 (a) shows the target set \mathcal{X}_f together with an approximation of the set that can be reach at time T by applying all input sequences in the set \mathcal{U}^T starting from $x(0) = [111]'$ (maximal reachable set). The approximation is computed by gridding \mathcal{U}^T and computing $x(T)$ for each point of the grid. Note that just a small portion of the reachable set is contained in the target set, and thus it is necessary to limit the admissible range of at least one of the inputs, as highlighted by the achieved solution. Indeed, we can compute the admissible ranges for the inputs

$$\begin{aligned} u_1(k) &\in [0.5, 1.5] \quad \forall k = 0, \dots, 4, \\ u_2(k) &\in \begin{cases} [-1, -0.915] & k = 0, 4 \\ [0.915, 1] & k = 1, 2, 3 \end{cases} \end{aligned}$$

using (6) with $\beta_1 = \beta_1^*$, $\beta_2 = \beta_2^*$, $u_{\beta,1} = u_{\beta,1}^*$, and $u_{\beta,2} = u_{\beta,2}^*$, and verify that the set of values for $x(T)$ reached from $x(0) = [1 \ 1 \ 1]'$ when applying all input sequences in these ranges is contained within \mathcal{X}_f .

The more the constraints defining the target set become tighter, the more the inputs become influential. This is shown in Figure 2 (b), which refers to a smaller target set described by:

$$\begin{bmatrix} I_3 \\ -I_3 \end{bmatrix} x \leq [20 \ -4 \ 1 \ -5 \ 7 \ 0]'. \quad (20)$$

In this case, by solving problem (13) we get $\beta_1^* = 0.441$ and $\beta_2^* = 0$, so that both the inputs are influential.

We consider now a different linear system, described by:

$$\begin{aligned} x(k+1) &= \begin{bmatrix} 0.2 & 0.9 & 0.5 \\ -0.2 & 0.6 & -0.1 \\ 0.8 & -0.2 & 0.7 \end{bmatrix} x(k) + \begin{bmatrix} 0.7 \\ 0.1 \\ -0.7 \end{bmatrix} u_1(k) \\ &+ \begin{bmatrix} -0.1 \\ 0.4 \\ -0.9 \end{bmatrix} u_2(k) \end{aligned} \quad (21)$$

where $x \in \mathbb{R}^3$, $u_1 \in [0.5, 1.5]$, and $u_2 \in [-1, 1]$, and the goal is reaching some target set \mathcal{X}_f at $T = 4$ from $x_0 = [000]'$. The purpose of this second example is to show that, depending on \mathcal{X}_f , an input can be either influential or non influential. As a matter of fact, if we choose \mathcal{X}_f as

$$\begin{bmatrix} I_3 \\ -I_3 \end{bmatrix} x \leq [1.5 \ 2 \ 3.5 \ -0.5 \ 1.5 \ 5]'. \quad (22)$$

5. CONCLUSIONS

In this paper we have introduced an input design problem that originates from the system verification context. We have proposed a solution that rests on an appropriate parametrization of the input variables as set-valued signals and on the reformulation of the design problem as a robust optimization problem. Computability of the solution has been shown in the case of linear systems and mixed logical dynamical systems with continuous inputs.

Future work includes extending the approach to the case of logic inputs and applying it to a realistic case study. This will eventually prompt the need for improving the scalability of the approach, possibly adopting some decomposition technique.

REFERENCES

- A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3): 407–427, 1999.
- A. Bemporad, G. Ferrari-Trecate, and M. Morari. Observability and Controllability of Piecewise Affine and Hybrid Systems. *IEEE Transactions on Automatic Control*, 45(10):1864–1876, 2000.
- A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.
- A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations research letters*, 25(1):1–13, 1999.
- A. Ben-Tal and A. Nemirovski. On tractable approximations of uncertain linear matrix inequalities affected by interval uncertainty. *SIAM Journal on Optimization*, 12(3):811–833, 2002.
- S.P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- E.J. Candès, M.B. Wakin, and S.P. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications*, 14(5-6):877–905, 2008.
- A. Chutinan and B.H. Krogh. Computational techniques for hybrid system verification. *IEEE Transactions on Automatic Control*, 48(1):64–75, 2003.
- ILOG CPLEX. 11.0 users manual. *ILOG SA, Gentilly, France*, 2007.
- G. E. Fainekos, A. Girard, H. Kress-Gazit, and G.J. Pappas. Temporal logic motion planning for dynamic robots. *Automatica*, 45(2):343 – 352, 2009.
- G.E. Fainekos, S.G. Loizou, and G.J. Pappas. Translating temporal logic to controller specifications. In *45th IEEE Conference on Decision and Control*, pages 899–904, 2006.
- A. Girard, C. Le Guernic, and O. Maler. Efficient computation of reachable sets of linear time-invariant systems with inputs. In J. Hespanha and A. Tiwari, editors, *Hybrid Systems: Computation and Control*, volume 3927 of *Lecture Notes in Computer Science*, pages 257–271. Springer, Berlin, 2006.
- C. Le Guernic and A. Girard. Reachability analysis of linear systems using support functions. *Nonlinear Analysis: Hybrid Systems*, 4(2):250 – 262, 2010.
- A.A. Kurzhanskiy and P. Varaiya. Ellipsoidal techniques for reachability analysis of discrete-time linear systems.

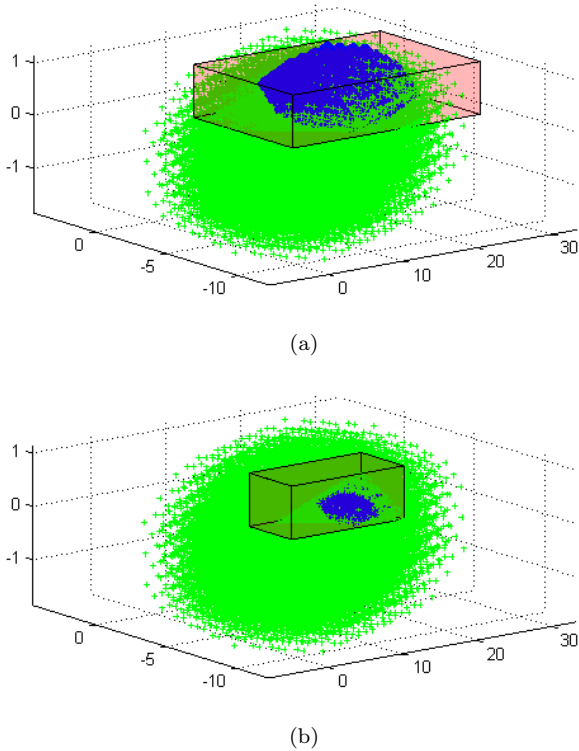


Fig. 2. Maximal reachable set (green), target set \mathcal{X}_f (red), reachable set corresponding to the designed input ranges (blue).

we obtain $\beta_1^* = 0.08$ and $\beta_2^* = 1$ so that input u_2 is non influential (Figure 3 (a)), while, if we choose \mathcal{X}_f as

$$\begin{bmatrix} I_3 \\ -I_3 \end{bmatrix} x \leq [0.8 \ 1.2 \ -2 \ 0.2 \ -0.8 \ 4]'$$
 (23)

we obtain $\beta_1^* = 1$ and $\beta_2^* = 0.097$ so that it is now the first input to be non influential (Figure 3 (b)).

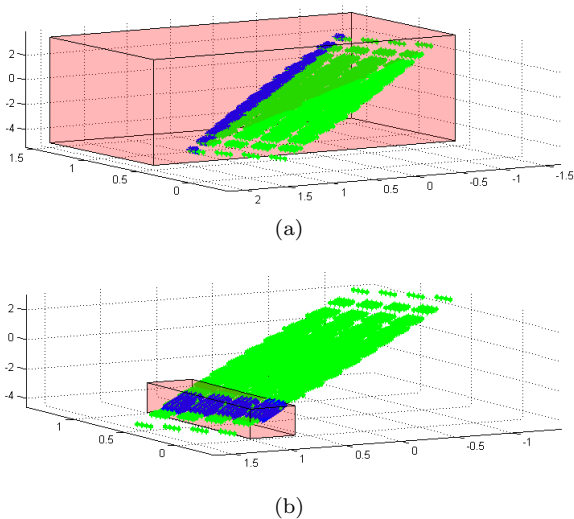


Fig. 3. Maximal reachable set (green), target set \mathcal{X}_f (red), reachable set corresponding to the designed input ranges (blue).

- IEEE Transactions on Automatic Control*, 52(1):26–38, 2007.
- J. Löfberg. Yalmip: a toolbox for modeling and optimization in MATLAB. In *Proceedings of 13th IEEE Symposium on Computer Aided Control System Design*, 2004. URL <http://control.ee.ethz.ch/~joloef/yalmip.php>.
- I. Mitchell and C. Tomlin. Level set methods in for computation in hybrid systems. In *Hybrid Systems: Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, pages 310–323. Springer Verlag, 2000.
- C.J. Tomlin, I. Mitchell, A.M. Bayen, and M. Oishi. Computational techniques for the verification of hybrid systems. *Proceedings of the IEEE*, 91(7):986–1001, 2003.