Synthesis of separable controlled invariant sets for modular local control design

Petter Nilsson and Necmiye Ozay

Abstract—Many correct-by-construction control synthesis methods suffer from the curse of dimensionality. Motivated by this challenge, we seek to reduce a correct-by-construction control synthesis problem to subproblems of more modest dimension. As a step towards this goal, in this paper we consider the problem of synthesizing decoupled robustly controlled invariant sets for dynamically coupled linear subsystems with state and input constraints. Our approach, which gives sufficient conditions for decoupled invariance, is based on optimization over linear matrix inequalities which are obtained using slack variable identities. We illustrate the applicability of our method on several examples, including one where we solve local control synthesis problems in a compositional manner.

I. INTRODUCTION

Distributed embedded control systems are already integral parts of many safety-critical systems, including those in avionics [1], [2], automotive [3], electricity generation and distribution [4], and medical equipment [5]. Due to increasing complexity and integration of a large number of components and subsystems, certification of safety and performance properties of such systems constitutes a bottleneck in terms of design time. This burden can be partially alleviated by adopting formal methods-based verification and correct-by-construction control synthesis techniques [6].

A fundamental property related to safety is invariance [7]. In this paper, we consider the problem of synthesizing polytopic controlled invariant sets for distributed systems consisting of a set of linear constrained subsystems. We assume that each subsystem has its own controller that has some local sensing capabilities and that is required to achieve a local safety specification, while there is coupling between subsystems through the dynamics. Our goal is to synthesize a separable controlled invariant set, which essentially is a cross-product of local invariant sets for the subsystems. The behaviors of the subsystems can then be decoupled as long as their states are constrained to these sets. As such, it is possible to compositionally synthesize more advanced controllers (for instance, from temporal logic specifications, or using model predictive control) within these sets.

Compositional approaches have attracted considerable attention in recent years in the context of verification of stability [8], safety [9], and performance [10] specifications of dynamical systems. The results on compositional synthesis are mostly limited to linear systems. Our work also falls into this category and is tightly related to the compositional synthesis approaches proposed for finding ellipsoidal controlled invariant sets for linear systems to be used as terminal sets in model predictive control [11], [12], [13], [14]. Instead of searching for ellipsoidal invariant sets, we search for polytopic invariant sets with tunable complexity. Polytopic sets are more commonly used in temporal logic-based control synthesis techniques [15], [16], [17], [18]. Therefore, our approach constitutes a first step in extending some of these techniques to the distributed setting.

Our main contribution is to demonstrate how local invariant sets can be used to separately synthesize local controllers to achieve local tasks while still guaranteeing correctness when these controllers are composed together. This is achieved by (i) synthesizing feedback controllers that render a subset of the state-space of each subsystem robustly invariant, (ii) extracting the set of allowable inputs that will keep these sets invariant even when the feedback controllers are discarded, and (iii) solving a local synthesis problem with complex task specifications, for instance, expressed in temporal logics, with the new state and input constraints. For the first step we build on recent results on low complexity polytopic invariant set computation. In particular, we extend the results in [19] from the centralized setting to the decentralized setting and allow synthesis of local invariant sets in the form of arbitrary zonotopes, where the number of generators is an input to our method, as opposed to linearly transformed hyper boxes as in [19]. Although conceptually simple, the computation of these local invariant sets is far from trivial, as it requires solving a non-convex feasibility problem. In order to overcome this difficulty, we resort to slack variable based relaxations and provide linear matrix inequality (LMI)-based sufficient conditions for computing these invariant sets.

We have structured this paper in the following way. After giving preliminaries on notation and three key results from [19] in Section II, we formally introduce the decoupled invariance problem in Section III. Next, we present the main results of this paper in Section IV and discuss where conservativeness is introduced. We give some examples in Section V to illustrate the advantages of our method, before concluding the paper in Section VI.

II. PRELIMINARIES

In this paper we will make use of the following notation; \( I_n \) denotes the identity matrix of size \( n \times n \). When the dimension is apparent from the context, the subscript will be dropped. We denote by \( e_i \) the \( i \)-th Euclidean standard basis column vector, while \( 1 \) is a column vector where all entries are 1. Furthermore, given a one-dimensional set of matrices \( \{ A_i \}_i = \{ A_1, A_2, \ldots \} \), we denote by \( \text{blkdiag}(\{ A_i \}_i) \)
the block-diagonal matrix formed by that set. Similarly, for a two-dimensional set of matrices \( \{A_{i,j}\}_{i,j} \), we write fullmat(\( \{A_{i,j}\}_{i,j} \)) to indicate the block matrix whose \((i,j)\)th block is \( A_{i,j} \). When we write \( \Pi_iX_i \) we mean the cross product between sets, i.e. \( \Pi_{i=1}X_i = X_1 \times X_2 = \{(x_1,x_2) : x_1 \in X_1, x_2 \in X_2\} \). For two sets \( X \) and \( Y \), we denote the Minkowski sum by \( X \oplus Y = \{x + y : x \in X, y \in Y\} \).

When we introduce matrix variables, we will use “(full)”, “(sym)”, or “(diag)” to indicate if this matrix variable is a full matrix, or restricted to being symmetric or diagonal. In the case of symmetric matrices, we will write \( * \) for entries whose values follow from symmetry.

We now present three lemmas that are crucial to the developments in later sections. First, we recall the following result from [19] which we state without proof.

Lemma 1: Let \( R \) (sym), \( Z \) (sym), \( A \) (full) and \( B \) (full) be arbitrary matrices. Then the following conditions are equivalent

1) \[
\begin{bmatrix} R & AB \\ * & Z \end{bmatrix} \succ 0. \tag{1}
\]

2) \[
\exists X \text{ (sym)} : \begin{bmatrix} R & A \\ * & X^{-1} \end{bmatrix} \succ 0, \quad \begin{bmatrix} X & B \\ * & Z \end{bmatrix} \succ 0 \tag{2}
\]

The usefulness of this lemma is to separate the matrix product \( AB \) in the upper right entry of (1). However, the introduced matrix variable \( X \) appears both as itself and its inverse.

The next two lemmas are inspired by the same paper, but we have altered the formulation to our purposes. For completeness, we provide proofs in the Appendix.

Lemma 2: The following two statements are equivalent

1) \[
\begin{bmatrix} C^TXC & Y \\ * & Z \end{bmatrix} \succ 0. \tag{3}
\]

2) \( C \) is non-singular, \( X \succ 0 \), and there exists \( \Psi \) such that
\[
\begin{bmatrix} C \Psi + \Psi^TC^{-1}X^{-1} & \Psi^TY \\ * & Z \end{bmatrix} \succ 0. \tag{4}
\]

Lemma 3: If there exist \( \Theta \) (full), \( \Gamma \) (sym) and \( \Xi \) (sym) such that
\[
\Delta = \begin{bmatrix} \Gamma & \Xi \\ * & \Theta \end{bmatrix} \succ 0, \quad \begin{bmatrix} Z + \Xi^TV^T & \Theta^T \Theta - \Delta & V \\ * & * & 0 \\ * & * & 0 \end{bmatrix} \succ 0, \tag{5}
\]

then
\[
\begin{bmatrix} Z + XY^TV^T & X^TV^T & V \\ * & * & W \end{bmatrix} \succ 0. \tag{6}
\]

Remark 1: As is evident from the proof, if \( X \Gamma X^T \) is added to the top left entry of the right hand matrix in (5), the reverse implication also holds.

III. PROBLEM FORMULATION

As alluded to in Section 3 as a first step in modular local control design, we seek to find a separable invariant set for a system, where the splitting into subsystems is given beforehand. Specifically, we consider a discrete time system of the form
\[
x_i(t+1) = A_{i,i}x_i(t) + \sum_{j \neq i} A_{i,j}x_j(t) + B_iu_i(t) + E_i d_i(t), \tag{7}
\]
for \( i = 1, \ldots, d \), where \( x_i \in \mathbb{R}^{n_i} \) is the state of subsystem \( i \).

As can be seen, each subsystem is associated with an input \( u_i \in \mathbb{R}^{m_i} \) and a disturbance \( d_i \in \mathbb{R}^{p_i} \), both of which do not affect other subsystems. We assume that the inputs of (7) are bounded as
\[
u_i(t) \in U_i = \{ u_i : H_{i}^iu_i \leq h_i \}, \quad H_{i}^i \in \mathbb{R}^{n_i \times m_i}, \quad \text{for all } i, \tag{8}
\]
and that there are bounds on disturbance given by
\[
d_i(t) \in D_i = \{ d_i : -1 \leq H_i^dd_i \leq 1 \}, \quad \text{for all } i. \tag{9}
\]

where \( H_i^j \in \mathbb{R}^{p_i \times p_i} \) is a non-singular square matrix. We also consider state constraints of the form
\[
x_i \in \mathcal{S}_i = \{ x_i : H_i^ix_i \leq h_i \}, \quad \text{for all } i. \tag{10}
\]

We are now ready to state the first problem we seek to solve.

Problem 1: Given a set of dynamically coupled linear subsystems (7) together with input constraints, disturbance bounds and state constraints of the forms (8) - (10), find for all \( i = 1, \ldots, d \), sets \( \mathcal{X}_i \subset \mathcal{S}_i \) such that
\[
\forall x_i(t) \in \mathcal{X}_i, \exists u_i(t) \in U_i \quad \forall x_j(t) \in \mathcal{X}_j \quad \forall d_i(t) \in D_i, \quad x_i(t+1) \in \mathcal{X}_i, \quad \text{for all } i. \tag{11}
\]

This problem exhibits some interesting features due to its circular nature. A solution is basically an assume-guarantee protocol [20] where each subsystem guarantees that it will affect other subsystems. We assume that the inputs of (7) are affect other subsystems. We assume that the inputs of (7) are
IV. MAIN RESULTS

A. Computation of separable invariant sets

In order to attack Problem 1 we restrict the description in two ways. Firstly, we do not consider arbitrary nonlinear controllers but search for sets that can be rendered invariant by local feedback controllers $u_i = K_i x_i$. Secondly, we restrict the sets $X_i$ to be symmetric zonotopes with a fixed number of generators. Specifically, we search for invariant sets $X_i$ of the form

$$X_i = \{ x_i : -1 \leq Z_i H^T x_i \leq 1 \},$$

where $Z_i \in \mathbb{R}^{N_i \times n_i}$ is an arbitrary given matrix and $H_i \in \mathbb{R}^{n_i \times n_i}$ is non-singular. $N_i$ gives the number of generators of the zonotope describing the set $X_i$, whereas $H_i$ is a variable that represents an unknown linear transformation. This description allows us to tune the complexity of the invariant sets we search for.

Before stating the main result, we concatenate the indexed variables introduced above to obtain a compact representation of the overall system. Specifically, let $A = \text{fullmat}(\{A_{ij}\}_{i,j}), B = \text{blkdiag}(\{B_i\}_{i}), K = \text{blkdiag}(\{K_i\}_{i}), E = \text{blkdiag}(\{E_i\}_{i}), Z = \text{blkdiag}(\{Z_i\}_{i}), H_x = \text{blkdiag}(\{H^x_{i}\}_{i}), H_u = \text{blkdiag}(\{H^u_{i}\}_{i}), H_s = \text{blkdiag}(\{H^s_{i}\}_{i}), H_d = \text{blkdiag}(\{H^d_{i}\}_{i}), h_s = [(h^s_1)^T, \ldots, (h^s_{N_s})^T]^T, h_d = [(h^d_1)^T, \ldots, (h^d_{N_d})^T]^T, x = [x^T \ldots x^T] = x \in \mathbb{R}^n, u = [u^T \ldots u^T] = u \in \mathbb{R}^d$ and $d = [d^T \ldots d^T] = d \in \mathbb{R}^d$. The dimensions of these composed variables become $n = \sum n_i$ and similarly for $m$ and $p$. The number of rows of $Z, H_u, H_d$ are $N = \sum_i N_i^z, N_u = \sum_i N_u^i$ and $N_s = \sum_i N_s^i$, respectively. For $A_K = A + BK$, the closed loop dynamics then become

$$x(t + 1) = A_K x(t) + E d(t).$$

The control and state constraints are given by $X = \bigcap_i X_i$ and $D = \bigcap_i D_i$, which can be compactly represented as

$$X = \{ x : -1 \leq Z x \leq 1 \},$$

$$D = \{ d : -1 \leq H_d d \leq 1 \}.$$  

The conditions in Problem 1 can now be stated as follows using set-theoretic constructs.

1) Separable invariance of closed loop system:

$$A_K X \oplus E D \subset X.$$  

2) State constraints:

$$X \subset \bigcap_i S_i = \{ x : H x \leq h \}.$$  

3) Control constraints:

$$B K X \subset \bigcap_i U_i = \{ u : H_u u \leq h_u \}.$$  

**Theorem 1:** If there exists $\Lambda = \text{blkdiag}(\{I_{n_i}, \lambda_i\}_{i=1}^d) > 0$, and for all $j = 1, \ldots, N_j$ there exist matrix variables $D^j_2 > 0$ (diag), $\Phi_j (\text{sym})$, $\Gamma_j (\text{sym}), \Xi_j (\text{sym}), \Psi_j (\text{full}), \Omega_j^1 (\text{full}), \Omega_j^2 (\text{full})$, and for all $k = 1, \ldots, N_k$ matrix variables $D^k_2 > 0$ (diag), and for all $d = 1, \ldots, N_d$ matrix variables $D^d_2 > 0$ (diag), such that LMI’s (13) and (19) - (21) are satisfied, then the block diagonal pair $(H_x, K H_x)$ gives rise to three additional LMI’s on the same forms as (13), (19) and (20), which are shown to be convex. This allows us to state the following.

**Remark 3:** For a given $\Lambda$, $X_i$ is a property of $X_i$, whereas $H_i$ is a variable that represents an unknown linear transformation. This description allows us to tune the complexity of the invariant sets we search for.

**Remark 4:** The feedback matrix $K$ was defined above to be block diagonal, to only allow local state information to influence the control signal computation. If more information is available locally (i.e., the states of neighboring subsystems are available to the controller), this restriction can be relaxed by allowing additional non-zero blocks in $K$. We illustrate this with an example in Section V-A where neighboring subsystems are allowed to interchange state information.

**Remark 5:** A natural way to select $Z_i$ is to pick (randomly or evenly spaced) unit vectors from $S^{n_i - 1}_+ = \{ (g_1, \ldots, g_{n_i}) \in S^{n_i - 1} : g_i \geq 0 \}$, where $S^n$ is the $n$-dimensional unit sphere.

**Remark 6:** The sizes of the matrices in (13) and (19) - (21) are $4n \times 4n, 6n + 1 \times 6n + 1, 4n \times 4n, m + n + 1 \times m + n + 1$, respectively. When there is no external disturbance, there is no need to introduce the variables $D^d_2$ and the sizes of the first three matrices reduce to $2n \times 2n, 3n + 1 \times 3n + 1$ and $2n \times 2n$, which is a computationally easier problem.

B. Using separable invariant sets for compositional synthesis

In general, $K_i x_i$ is not the unique control signal that renders $X_i$ invariant. By performing polytopic reachability computations, the envelope of invariant-enforcing controls can be extracted for a given point. For such a point $x_i^j \in \mathbb{R}^{n_i}$
\(\mathcal{X}_i\), the set of all inputs that both satisfy the local control constraints and guarantees invariance of \(\mathcal{X}_i\) is the set of \(u_i\)’s that satisfy

\[
\begin{bmatrix}
H_i^u^T \\ Z_i H_i^u B
\end{bmatrix} \leq \begin{bmatrix}
1 - H_i^u \delta_0 - \sum_{j \neq i, j \in \mathcal{X}_i} \max_{x_j} H_i^u A_{ij} x_j \\
-1 + H_i^u \delta_0 + \sum_{j \neq i, j \in \mathcal{X}_i} \min_{x_j} H_i^u A_{ij} x_j
\end{bmatrix},
\]  

(22)

where the inequality should hold element-wise. By construction this set will be non-empty as long as \(x_i^0 \in \mathcal{X}_i\). The \(\max\) and \(\min\) terms in this expression can be found by solving linear programs over the other invariant sets \(\mathcal{X}_j\), or by enumerating their vertices. This flexibility in control signal selection can be exploited to design local controllers separately in a compositional manner, for instance with the goal of performing local control tasks.

In what follows we assume that the local task specifications are given in terms of linear temporal logic (LTL) over atomic propositions defined on the state-spaces of individual subsystems. To summarize, by construction we have the following result which enables local controller synthesis with global guarantees.

**Proposition 1:** For a given system in the form (7) - (10), let \(\mathcal{X}_i\) be invariant sets that satisfy the requirements of Problem 1 and let \(\varphi_i\) be a local LTL specification for subsystem \(i\) for \(i = 1, \ldots, d\). Furthermore, for \(i = 1, \ldots, d\), let \(u_i : \mathcal{X}_i \rightarrow \mathcal{U}_i\) be local controllers that generate closed loop trajectories that satisfy \(\varphi_i\), while also satisfying the (state-dependent) input constraint (22). Then, the composed control \(u \doteq [u_1, \ldots, u_d] : \prod_i \mathcal{X}_i \rightarrow \prod_i \mathcal{U}_i\) generates closed loop behaviors of the overall system that satisfy \(\bigwedge_i \varphi_i\).

**C. Discussion about conservativeness**

For the separable invariant set computation, conservativeness enters the proof of Theorem 1 at two places. First, a positive term \(\Lambda^{-1} \Gamma_i^{-1} \Gamma_j \Gamma_j^{-1} \Lambda^{-1}\) is thrown away when Lemma 3 is used on \(\Theta_j\). Secondly, instead of allowing an arbitrary \(\Theta_j\) in (45), the upper two blocks of \(\Theta_j\) is restricted to be equal to \(\Lambda\) to make the resulting inequality linear. While it is easy to post to evaluate the effect of ignoring the positive term by looking at its magnitude, it is more subtle how much conservativeness is introduced by restricting \(\Theta_j\). Once an initial solution is obtained, it can be iteratively updated as in [19] by using the values of LMI variables \(\Phi_j^0\) and \(H_j^0\) from a previous iteration. The idea is to use a different matrix inverse identity than the one employed in the proof Lemma 3. The result is that the only term that needs to be discarded, and thus introduces conservativeness, is a quadratic term that is small if \(\Phi_j^0\) and \(\Phi_j\), and \(\Gamma_j\) and \(\Gamma_j^0\), are close. Therefore a solution can be iteratively updated without much conservativeness by performing small steps.

For the overall synthesis problem, first computing the local invariant sets and then trying to solve local synthesis problems in each invariant set is clearly more conservative than trying to synthesize local controllers by taking into account all the interactions. However, there are trade-offs between conservativeness, modularity of the local control design and computational complexity. Correct-by-construction control synthesis from temporal logic specifications is computationally challenging for high dimensional systems in general [6], [16]. Therefore, decomposing the problem into smaller subproblems per subsystem improves scalability [2]. Moreover, the local robust controlled invariant sets provide a modular framework in that it is possible to replace an existing local controller for a subsystem, for instance in order to pursue a different local control objective, without needing to resynthesize the rest of the local controllers.

**V. Examples**

We illustrate the applicability of this approach on a few examples. First, in Section V-A we compute separable invariant sets for two different systems. Thanks to the increased flexibility in our set description, we are able to solve problems where the geometry is not compatible with linearly transformed hyper boxes. Then, in Section V-B we look at a system of connected mobile robots and show how our framework allows local control tasks to be performed while guaranteeing overall safety.

**A. Finding invariant sets**

The dynamics of our first example are as follows. For each subsystem \(x_i \in \mathbb{R}^2\), the evolution consists of a rotational part and disturbance coming both from the other subsystems and from an additive term.

\[
x_i(t + 1) = \alpha_i R(\theta) x_i(t) + u_i(t) + \sum_{i \neq j} \beta_{ij} x_j + d_i(t). \tag{23}
\]

Here \(R(\theta) \in SO(2)\) is the (counter-clockwise) rotation matrix. The input \(u_i \in \mathbb{R}^2\) and disturbance \(d_i \in \mathbb{R}^2\) are bounded by \(\|u_i\|_\infty \leq u_{\text{max}}\) and \(\|d_i\|_\infty \leq d_{\text{max}}\).

We solved the LMI’s in Theorem 1 for a system with three subsystems and parameters \(\theta = \pi/4\), \(\alpha_i = 0.8\), \(\beta_{ij} = 0.1\) for \(|i - j| = 1\) and 0 otherwise, \(u_{\text{max}} = 0.65\), \(d_{\text{max}} = 0.4\) for all \(i, j\), with state constraints \(\|x_i\|_\infty \leq 1\), and for \(Z = [z_1, z_2, \ldots, z_8]^T\), where \(z_k\) are randomly chosen unit vectors in \(S^1\). The resulting robustly controlled invariant sets that are depicted in Fig. 1 were computed in 11 seconds.

Due to the rotational geometry of this problem, the additional set flexibility introduced by the \(Z\) matrix is crucial in order to achieve feasibility. Indeed, we were not able to find decoupled invariant sets consisting of linearly transformed hyper boxes using our implementation of the previous work [19].

Next, we present another example of finding invariant sets, this time for an array of \(N\) undisturbed inverted pendulums connected by springs and dampers, an example taken from [13]. A pendulum at position \(i\) in the interior of the array (i.e. \(i \notin \{1, N\}\)) is described by the states \((\theta_i, \dot{\theta}_i)\) and has the linearized dynamics

\[
\begin{bmatrix}
\delta_1 \\
\delta_\dot{1}
\end{bmatrix} = \begin{bmatrix}
0 & 1 \\
-2k & -2c
\end{bmatrix} \begin{bmatrix}
\delta_1 \\
\delta_\dot{1}
\end{bmatrix} + \begin{bmatrix}
u_i \end{bmatrix} + \begin{bmatrix}
0 \ c \\
0 \ k
\end{bmatrix} \begin{bmatrix}
\delta_{i+1} + \delta_{i-1} \\
\dot{\delta}_{i+1} + \dot{\delta}_{i-1}
\end{bmatrix}. \tag{24}
\]
imposed input bounds of \( l_u \) at position 2 and has the dynamics positions 2 and 1, respectively.

The pendulum at position 1 is only connected to the pendulum at position 2 and has the dynamics

\[
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_1
\end{bmatrix} = \begin{bmatrix} 0 & 1 \\ k & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\
\theta_1
\end{bmatrix} + \begin{bmatrix} 0 \\
k \end{bmatrix} \begin{bmatrix} u_1 \\
u_1
\end{bmatrix},
\]

and symmetrically for the pendulum at position \( N \). We used the same parameter values as in the cited paper which were as follows. The spring and damper parameters are set to \( k = c = 3 \), and we discretized the continuous time dynamics using a time step \( \Delta t = 0.1 \) s using Euler forward. We imposed input bounds of \( |u_i| < 10 \) for all \( i \) and state constraints \( \max(|\theta_i|, |\dot{\theta}_i|) \leq 1 \) and allowed the input of pendulum \( i \) to depend on the states of pendulums \( i-1, i \) and \( i+1 \) (i.e. relaxing \( K \) to be a tri-block diagonal matrix).

For \( N = 5 \), and 6 generators per subsystem, we obtained the invariant sets depicted in Fig. 2 after finding an initial solution from the LMI’s in Theorem 2 and iteratively updating it as described in Section IV.C. The total computation time for initial solution and 5 iterations was 28 s.

**B. Local synthesis inside invariant sets**

Next, we consider a scenario involving a tethered UAV and a ground vehicle, where each vehicle is given a surveillance task that requires visiting certain regions infinitely often. The tether can be used to power the UAV to significantly increase the duration it is airborne, however it induces dynamic coupling between the UAV and the ground vehicle. This coupling is modeled as a spring. For simplicity, the vehicles are modeled as double integrators and their motion is constrained to one dimension:

\[
\begin{bmatrix}
x^{1(t+1)} \\
v^{1(t+1)}
\end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\
k & 1 & k & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
-k & 0 & k & 1 & 0 \\
-0 & 0 & 0 & 1 & 0
\end{bmatrix} \begin{bmatrix}
x^{1(t)} \\
v^{1(t)}
\end{bmatrix} + \begin{bmatrix} 0 \\
k \\
0 \\
k \\
0
\end{bmatrix} \begin{bmatrix} u^1 \\
u^1
\end{bmatrix}.
\]

Our objective is to find separable invariant sets for the two subsystems \((x^1, v^1)^2\) and \((x^2, v^2)^5\) and then let each system perform additional control objectives while still guaranteeing overall invariance. Using a small coupling \( k = 0.1 \) and a bound \( ||u_i||_{\infty} \leq 0.3 \), we could in 1.7 s compute the invariant sets depicted in Fig. 3 that consist of 5 zonotope generators each.

We now illustrate how additional control objectives can be modularly realized inside these sets. Let \( X_1, X_2 \) be the invariant set pertaining to the subsystem \((x^1, v^1)^2\). We consider a surveillance-like task where the ground robot is required to visit the goal sets \( G_i^1, G_i^2 \subset X_i \) infinitely often; whereas the UAV is required to visit \( G_i^2, G_i^2 \subset X_i \) infinitely often, where the goal sets are taken to be \( G_i^1 = \{x \in [0, 0.35] : x^1 \in [0.2, 0.35] \} \) and \( G_i^2 = \{x \in [0.05, 0.18] : x^1 \in [0.2, 0.35] \} \). The local objectives can be expressed in LTL as \( \varphi_i = \Box G_i^1 \land \Box G_i^2 \) for \( i = 1, 2 \).

We use reachability computations similar to [18] to synthesize the local controllers, but any LTL synthesis method can be used in this step. Fig. 4 depicts \( G_i^1 \), together with robust (with respect to “disturbance” induced from the other subsystem) backwards reachable sets of \( G_i^2 \) contained inside \( X_i \) in lighter green (the backwards reachable sets from \( G_i^1 \) are symmetric). Since the sets from where \( G_i^1 \) is reachable eventually cover \( G_i^1 \), together with the fact that \( G_i^1 \) is reachable from \( G_i^2 \) by symmetry, the specification is realizable. Fig. 5 shows trajectories of a simulation where both systems satisfy their control objectives, while simultaneously counteracting the “disturbance” they cause each other.

Now, assume the task specification for the UAV changes to \( \varphi_2' = \Box G_2' \land \Box G_2' \), with \( G_2' = \{x^2 \in [-0.18, -0.05] \} \) and \( G_2'' = \{x^2 \in [0.18, 0.33] \} \). By construction of the invariant sets, if a new controller can be synthesized for the UAV, the new specification \( \varphi_1 \land \varphi_2' \) is guaranteed to be satisfied without making any changes to the controller of the ground robot, despite the potentially different “disturbance” inputs it gets from the UAV. Fig. 6 shows the trajectories of a simulation of this new scenario where both systems satisfy their control objectives.

An interesting feature of this problem is that if the invariant sets are increased in size, these control objectives
to achieve local tasks. By construction, this approach guarantees correctness when the local controllers are composed together. We proposed LMI based sufficient conditions for computation of the separable controlled invariant set. The proposed invariant set computation scheme is quite flexible as it allows for (i) incorporating external disturbances, (ii) handling different communication constraints (i.e., which states each local controller has access to), and (iii) tuning the complexity of the invariant set description. As demonstrated by examples, this flexibility allows us to find invariant sets even when known polytopic invariant set construction schemes fail.

There are several directions for future work. On the theoretical side, we are interested in characterizing systems that admit separable invariant sets. There are recent results on characterizations of systems with separable Lyapunov functions [24]. We are working on extending some of these results to the controlled setting considered in this paper. Automatically decomposing a complex system into subsystems that admit local invariant sets is another direction. On the computational side, we used off-the-shelf convex optimization solvers to compute the invariant sets. It is possible to exploit the structure in the LMI formulation to increase the efficiency of the computation. In particular, we are currently working on implementing our solution using alternating direction method of multipliers [25], which will allow the invariant set computation to be performed in a distributed manner as well.

ACKNOWLEDGMENTS

The authors would like to thank Jessy Grizzle for helpful discussions. The work of PN was supported by NSF grant CNS-1239037. The work of NO was partly supported by NSF grant CNS-1446298.

APPENDIX

A. Proof of Lemma 2

We prove that the following two statements are equivalent

1) \[
\begin{bmatrix}
X & Y \\
Z & \mathbf{0}
\end{bmatrix} \succ 0
\]

2) \[
X \succ 0 \text{ and there exists } \Psi (\text{full}) \text{ such that } \begin{bmatrix}
\Psi^T + \Psi - X^{-1} \Psi^T Y \\
Z
\end{bmatrix} \succ 0.
\]

Then Lemma 2 follows by replacing \(X \rightarrow C^T X C\) in (28), the congruency transform blockdiag \((\{C, I\})\) and re-defining \(C\Psi^T \rightarrow \Psi^T\).

(1 \rightarrow 2): First note that (27) implies that \(X \succ 0\) by a Schur complement argument. By a congruency transform, (27) is equivalent to

\[
\forall \Psi \text{ non-singular, } \begin{bmatrix}
\Psi^T X \Psi & \Psi^T Y \\
Z
\end{bmatrix} \succ 0.
\]

Since \(X\) is invertible we can write \(\Psi^T X \Psi = \Psi^T + \Psi - X^{-1} + (\Psi - X)^T X^{-1} (\Psi - X)\). Substituting this expression in (29) and choosing \(\Psi\) as the (non-singular) matrix \(X\) which eliminates the last term shows that there exists \(\Psi\) such that (28) holds.
Similarly, this, we express the left hand side of (34) in terms of the

\begin{equation}
\exists \Psi \text{ s.t. } [\Psi^T X \Psi \Psi^T Y] > 0.
\end{equation}

A contradiction argument shows that \( \Psi \) is non-singular, so we can apply a congruency transform that eliminates \( \Psi \).

**B. Proof of Lemma 3**

Adding the positive definite terms \( X \Gamma X^T \) to the top left block and \( (\Theta - \Delta)^T \Delta^{-1} (\Theta - \Delta) \) to the middle block of (35) preserves positive definiteness. Using the identity \( \Theta^T \Delta^{-1} \Theta = \Theta + \Theta^T - \Delta + (\Theta - \Delta)^T \Delta^{-1} (\Theta - \Delta) \) then implies that

\begin{equation}
\begin{pmatrix}
Z + ZH_x X + X^T Z \Theta & V \Theta \\
W^T \Theta & 0
\end{pmatrix} > 0.
\end{equation}

A contradiction argument shows that \( \Theta \) is non-singular. The congruency transform

\begin{equation}
\begin{pmatrix}
I & 0 \\
0 & -I
\end{pmatrix}
\end{equation}

then gives

\begin{equation}
\begin{pmatrix}
Z + ZH_x X + X^T Z & V \\
W & 0
\end{pmatrix} > 0.
\end{equation}

Applying a Schur complement finally implies that

\begin{equation}
0 < \begin{cases}
Z + ZH_x X + X^T Z & V \\
W & 0
\end{cases} > 0 \text{ for all } x \in \mathcal{X} \text{ if and only if } -x \in \mathcal{X} \text{ and } W = \Delta^{-1}.
\end{equation}

C. Proof of Theorem 1

We verify that the satisfaction of the LMI’s given in the theorem statement guarantee (16) - (18), starting with (16). Because of symmetry, \( x \in \mathcal{X} \) if and only if \( -x \in \mathcal{X} \). Therefore it follows that (16) holds if and only if

\begin{equation}
e^T Z H_x (A_k x + E_d) - 1 \leq 0 \text{ for all } x \in \mathcal{X}, d \in D \text{ and for all } j = 1, \ldots, N_x.
\end{equation}

Furthermore note that \( x \in \mathcal{X} \) if and only if for all diagonal \( D_x > 0 \)

\begin{equation}
(1 - Z H_x x)^T D_x (1 + Z H_x x) \geq 0.
\end{equation}

Similarly, \( d \in D \) if and only if for all diagonal \( D_d > 0 \)

\begin{equation}
(1 - H_d d)^T D_d (1 + H_d d) \geq 0.
\end{equation}

The next step is to employ the S Procedure. To prepare for this, we express the left hand side of (34) in terms of the quadratic forms in (33)-(38) and an additional quadratic term:

\begin{align}
e^T Z H_x A_k x + e^T Z H_x E_d - 1 \\
= - (1 - Z H_x x)^T D_x (1 + Z H_x x) \\
- (1 - H_d d)^T D_d (1 + H_d d) \\
- [x^T d^T 1] L_x^T (D_x, D_d) [x^T d^T 1]^T,
\end{align}

for \( D_x > 0 \) (diag), \( D_d > 0 \) (diag) and

\begin{equation}
L_x^T (D_x, D_d) = \begin{bmatrix}
H_x^T z^T D_x H_x & 0 & -\frac{1}{2} A_k x^T \Psi^* e_j \\
0 & H_x^T D_x H_d & 0 \\
-\frac{1}{2} \Psi^* e_j H_x^T D_d & 0 & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix}.
\end{equation}

An application of the S procedure [26] shows that (16) holds if and only if for all \( j = 1, \ldots, N_x \) there exist \( D_x > 0 \) (diag) and \( D_d > 0 \) (diag), \( \Phi_j \text{ (sym)} \) such that

\begin{equation}
M_1 \begin{bmatrix}
H_x^T z^T D_x H_x & 0 & -\frac{1}{2} A_k x^T \Psi^* e_j \\
0 & H_x^T D_x H_d & 0 \\
-\frac{1}{2} \Psi^* e_j H_x^T D_d & 0 & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix} > 0,
\end{equation}

\begin{equation}
M_2 \begin{bmatrix}
H_x^T z^T & Z e_j & Z e_j \\
Z e_j & 1 - 1^T D_x 1 & 1 - 1^T D_d 1
\end{bmatrix} > 0.
\end{equation}

In the remaining part of the proof, these two matrix inequalities are turned into LMI’s.

**a) Treatment of M2:** Let \( H_x = \text{blkdiag} \{H_x, H_x\} \)

apply the congruency transform blkdiag \{\( H_x^{-T}, I \)\} on \( M_2 \) to obtain the equivalent condition

\begin{equation}
H_x^{-T} \Phi_j H_x^{-1} \begin{bmatrix}
Z e_j & Z e_j \\
Z e_j & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix} > 0.
\end{equation}

Multiply the matrix in (41) with a scalar \( \lambda_{i(j)} > 0 \), where \( i(j) \) is the index of the subsystem corresponding to the \( j \)-th inequality in \( Z \), and redefine \( \Phi_j = \lambda_{i(j)} \Phi_j \), \( D_x = \lambda_{i(j)} D_x \), \( D_d = \lambda_{i(j)} D_d \). Note that since \( Z^T \) is block diagonal, we have for \( \tilde{\Lambda} = \text{blkdiag} \{\tilde{\Lambda}_n, \lambda_{i(j)} \}_{i=1}^d \) that \( \Lambda Z^T e_j = \lambda_{i(j)} Z^T e_j \). This results in

\begin{equation}
\tilde{H}_x^{-T} \Phi_j \tilde{H}_x^{-1} \begin{bmatrix}
\Lambda Z^T e_j & \Lambda Z^T e_j \\
\Lambda Z^T e_j & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix} > 0.
\end{equation}

For \( \tilde{\Lambda} = \text{blkdiag} \{\{\Lambda, \Lambda\}\} \) apply the congruency transform blkdiag \{\( \tilde{\Lambda}^{-T}, I \)\}:

\begin{equation}
\tilde{\Lambda}^{-T} \tilde{H}_x^{-T} \Phi_j \tilde{H}_x^{-1} \begin{bmatrix}
\tilde{Z} e_j & \tilde{Z} e_j \\
\tilde{Z} e_j & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix} > 0.
\end{equation}

We now apply Lemma 2 which implies that (40) holds if and only if there exist \( \Phi_j \text{ (sym)} \) and \( \Psi_j \text{ (full)} \) such that

\begin{align}
\Psi_j^T \tilde{H}_x^{-T} & + \tilde{H}_x^{-T} \Phi_j \tilde{H}_x^{-1} - \tilde{Z} e_j \\
& + \Psi_j^T \tilde{Z} e_j
\end{align}

\begin{align}
+ \Psi_j^T \tilde{Z} e_j & + \tilde{Z} e_j \\
= \lambda_{i(j)} \tilde{Z} e_j & + \lambda_{i(j)} \tilde{Z} e_j
\end{align}

\begin{align}
+ \lambda_{i(j)} \tilde{Z} e_j & + \tilde{Z} e_j
\end{align}

\begin{align}
= \lambda_{i(j)} \tilde{Z} e_j & + \lambda_{i(j)} \tilde{Z} e_j
\end{align}

\begin{align}
+ \lambda_{i(j)} \tilde{Z} e_j & + \tilde{Z} e_j
\end{align}

Finally, we use Lemma 3 to obtain the necessary condition

\begin{align}
\Delta_j \begin{bmatrix}
\gamma_j & \gamma_j \\
\gamma_j & \gamma_j
\end{bmatrix} > 0,
\end{align}

\begin{align}
-\gamma_j \begin{bmatrix}
\tilde{H}_x^T & \tilde{Z} e_j \\
\tilde{Z} e_j & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix} > 0.
\end{align}

By further restricting to \( \Theta_j = \begin{bmatrix}
\Delta_j & \Delta_j \\
\Delta_j & \Delta_j
\end{bmatrix} \), we obtain the LMI’s in (13).

**b) Treatment of M1:** Apply the congruency transform blkdiag \{\( \lambda_{i(j)} H_x^{-T}, \lambda_{i(j)} H_x^{-T}, I \)\} to obtain

\begin{align}
\begin{bmatrix}
Z e_j & \lambda_{i(j)} D_d \\
\lambda_{i(j)} D_d & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix} > 0.
\end{align}

\begin{align}
\begin{bmatrix}
\lambda_{i(j)} D_d & 0 \\
0 & 1 - 1^T D_x 1 - 1^T D_d 1
\end{bmatrix} > 0.
\end{align}


Note that $H_\alpha^{-T}A_k^T = ((A + BK)H_\alpha^{-1})^T = (AH_\alpha^{-1} + BK)^T$, for $K = KH_x$. Thus after the same re-definitions as above for $D_k^1$, $D_k^2$ and $\bar{Y}_j$, we get the LMI (19).

We move on to finding an expression that ensures (17). As before, we write for $k = 1, \ldots, N_x$

$$c_k^T(H_x x - h_a) = -[(1 - ZH_x x)^T D_k^1 (1 + ZH_x x)] - [x^T L_k(D_k^2) [x^T 1]^T,$$

for

$$L_k(D_k^2) = \begin{bmatrix} H_x^T Z^T D_k^1 ZH_x - \frac{1}{e} H_x^T e_k & e_k^T h_a - e_k^T D_k^1 \end{bmatrix}.$$ (48)

The S procedure applies in the same way as before and by the congruence transform blkdiag ($\{H_x^{-T}, I\}$) we get the LMI (20) which represents necessary and sufficient conditions for (17).

Finally, the input constraints are handled similarly, that is, condition (18) is satisfied if and only if for all $l = 1, \ldots, N_u$, $c_l^T (H_u K x - h_a) \leq 0$ (49) for all $x \in \mathcal{X}$. By the S procedure this can be translated into the positive definiteness of the matrix

$$\begin{bmatrix} H_x^T Z^T D_l^1 ZH_x - \frac{1}{e} H_x^T e_l & e_l^T h_a - e_l^T D_l^1 \\ e_l^T h_a - e_l^T D_l^1 & e_l^T h_a - e_l^T D_l^1 \end{bmatrix} > 0. \quad (50)$$

Applying the congruence transform blkdiag ($H_x^{-T}, I$) gives (21) for the same $K$ as before.

We now argue that $(H_x, KH_x)$ that satisfy (13) and (19) - (21) are indeed a solution to Problem 1. By definition, both $H_x$ and $KH_x$ are block diagonal in the sizes of the subsystems, so we can extract sets $\mathcal{X}_i$ (from blocks of $H_x$) and local feedback controllers $K_i$ (from blocks of $KH_x$) that render $\mathcal{X}_i$ invariant. The feedback controller provides for each $x_i \in \mathcal{X}_i$ a control $K_i x_i$ that enforces invariance of $\mathcal{X}_i$. This shows that (11) is satisfied.

REFERENCES

[1] G. Tallant, P. Bose, J. Buffington, V. Crum, R. Hull, T. Johnson, B. Krog, and P. Prasanth, “Validation & verification of intelligent and adaptive control systems,” in IEEE International Conference, 2005.

[2] N. Ozay, U. Topcu, and R. Murray, “Distributed power allocation for vehicle management systems,” in Proc. of the IEEE CDC and ECC, 2011, pp. 4841–4848.

[3] R. K. Jurgen, Distributed Automotive Embedded Systems. SAE International, 2007.

[4] A. Massoud and B. F. Wollenberg, “Toward a smart grid: power delivery for the 21st century,” IEEE Power and Energy Magazine, vol. 3, no. 5, pp. 34–41, 2005.

[5] I. Lee, O. Sokolsky, S. Chen, J. Hatcliff, E. Jee, B. Kim, A. King, M. Mullen-Fortino, S. Park, A. Roederer, and K. Venkatakrishnan, “Challenges and research directions in medical cyber–physical systems,” Proc. of the IEEE, vol. 100, no. 1, pp. 75–90, 2012.

[6] P. Tabuada, Verification and control of hybrid systems: a symbolic approach. Springer Science & Business Media, 2009.

[7] F. Blanchini, “Set invariance in control,” Automatica, vol. 35, no. 11, pp. 1747–1767, 1999.

[8] U. Topcu, A. C. Packard, and R. M. Murray, “Compositional stability analysis based on dual decomposition,” in Proc. of IEEE CDC and CCC, 2009, pp. 1175–1180.

[9] C. Sloth, G. J. Pappas, and R. Wisniewski, “Compositional safety analysis using barrier certificates,” in Proc. of the ACM HSSE, 2012, pp. 15–24.

[10] C. Meissen, L. Lessard, M. Arcak, and A. Packard, “Performance certification of interconnected nonlinear systems using admm,” in Proc. of the IEEE CDC, 2014, pp. 5131–5136.

[11] A. Jokic and M. Lazar, “On decentralized stabilization of discrete-time nonlinear systems,” in Proc. of ACC, 2009, pp. 5777–5782.

[12] S. V. Rakovic, B. Kern, and R. Findeisen, “Practical set invariance for decentralized discrete time systems,” in Proc. of the IEEE CDC, 2010, pp. 3283–3288.

[13] A. Jokic and M. Lazar, “On feasibility, stability and performance in distributed model predictable control,” Automatic Control, IEEE Transactions on, vol. 59, no. 4, pp. 1031–1036, 2014.

[14] M. Kloetzer and C. Belta, “A fully automated framework for control of linear systems from temporal logic specifications,” IEEE Trans. Autom. Control, vol. 53, no. 1, pp. 287–297, 2008.

[15] T. Wongpiromsarn, U. Topcu, and R. M. Murray, “Receding horizon temporal logic planning,” IEEE Trans. Autom. Control, vol. 57, no. 11, pp. 2817–2830, 2012.

[16] F. Tahir and I. Jaimoukha, “Low-complexity polytopic invariant sets for linear systems subject to norm-bounded uncertainty,” IEEE Trans. Autom. Control, vol. 60, no. 5, pp. 1416–1421, May 2015.

[17] P. Fehse, Z. Han, and B. Krog, “Assume-guarantee reasoning for hybrid i/o-automata by over-approximation of continuous interaction,” in Proc. of the IEEE CDC, 2014, pp. 816–823.

[18] A. Jokic and M. Lazar, “On decentralized stabilization of discrete-time nonlinear systems,” in Proc. of ACC, 2009, pp. 5777–5782.

[19] C. Meissen, L. Lessard, M. Arcak, and A. Packard, “Performance certification of interconnected nonlinear systems using admm,” in Proc. of the IEEE CDC, 2014, pp. 5131–5136.

[20] A. Jokic and M. Lazar, “On feasibility, stability and performance in distributed model predictable control,” Automatic Control, IEEE Transactions on, vol. 59, no. 4, pp. 1031–1036, 2014.

[21] M. Kloetzer and C. Belta, “A fully automated framework for control of linear systems from temporal logic specifications,” IEEE Trans. Autom. Control, vol. 53, no. 1, pp. 287–297, 2008.

[22] F. Tahir and I. Jaimoukha, “Low-complexity polytopic invariant sets for linear systems subject to norm-bounded uncertainty,” IEEE Trans. Autom. Control, vol. 60, no. 5, pp. 1416–1421, May 2015.

[23] P. Fehse, Z. Han, and B. Krog, “Assume-guarantee reasoning for hybrid i/o-automata by over-approximation of continuous interaction,” in Proc. of the IEEE CDC, 2014, pp. 816–823.

[24] A. Jokic and M. Lazar, “On feasibility, stability and performance in distributed model predictable control,” Automatic Control, IEEE Transactions on, vol. 59, no. 4, pp. 1031–1036, 2014.

[25] M. Kloetzer and C. Belta, “A fully automated framework for control of linear systems from temporal logic specifications,” IEEE Trans. Autom. Control, vol. 53, no. 1, pp. 287–297, 2008.

[26] F. Tahir and I. Jaimoukha, “Low-complexity polytopic invariant sets for linear systems subject to norm-bounded uncertainty,” IEEE Trans. Autom. Control, vol. 60, no. 5, pp. 1416–1421, May 2015.

[27] P. Fehse, Z. Han, and B. Krog, “Assume-guarantee reasoning for hybrid i/o-automata by over-approximation of continuous interaction,” in Proc. of the IEEE CDC, 2014, pp. 816–823.