Automatica 74 (2016) 107–117

Contents lists available at ScienceDirect

Automatica

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

Closed-loop input design for guaranteed fault diagnosis using set-valued observers☆

Davide M. Raimondo [a], Giuseppe Roberto Marseglia [a], Richard D. Braatz [b], Joseph K. Scott [c]

[a] Identification and Control of Dynamic Systems Laboratory, University of Pavia, Italy
[b] Massachusetts Institute of Technology, Cambridge, MA, USA
[c] Department of Chemical and Biomolecular Engineering, Clemson University, Clemson, SC, USA

ARTICLE INFO

Article history:
Received 14 December 2015
Accepted 17 June 2016
Available online 1 October 2016

Keywords:
Fault detection
Set-based estimation
Zonotopes
Robust control

ABSTRACT

Active fault diagnosis (AFD) can be used to improve the diagnosability of faults by injecting a suitably designed input into a process. When faults are described as discrete switches between linear systems with uncertainties bounded within zonotopes, an optimal open-loop input guaranteeing diagnosis within a specified time horizon can be computed efficiently by solving a Mixed Integer Quadratic Program (MIQP). In this article, the constrained zonotope (CZ) set representation recently developed by the authors is used to extend the MIQP approach to general polytopic uncertainties without sacrificing efficiency. Next, this approach is combined with a CZ-based set-valued observer in a moving horizon framework to achieve rigorous closed-loop AFD. This method can greatly accelerate diagnosis relative to the open-loop approach, but requires online optimization. To reduce the online cost, we propose a method for solving the open-loop problem explicitly with respect to past measurements and inputs, which requires only observability of the nominal and faulty models. The effectiveness of the proposed approaches is demonstrated through several numerical examples.

1. Introduction

The detection and diagnosis of malfunctions and other abnormal events (i.e., faults) is an essential control task for engineered systems in the chemical, power, aerospace, and mechanical domains (Gao, Cecati, & Ding, 2015; Tchakoua et al., 2014; Yu, Woradechjumroen, & Yu, 2014). Without corrective action, faults can lead to performance degradation and potentially critical situations. However, fault detection and diagnosis are challenging due to the presence of disturbances, measurement noises, and the actions of feedback controllers. Approaches to automatic fault diagnosis can be classified as either active or passive. In the passive approach, input–output data are collected in real-time and faults are diagnosed based on comparisons with a process model or historical data. In contrast, the active approach involves injecting a signal into the system to improve the diagnosability of potential faults with minimal impact on the nominal system (Gao et al., 2015).

This article considers input design for active fault diagnosis of linear systems subject to bounded process and measurement noise, and faults modeled by discrete changes in the system matrices. Several works have addressed this problem using inputs that excite specially designed residual signals in the presence of faults (Keresteciogl & Cetin, 2004; Niemann, 2006). A multi-model stochastic formulation is considered in Blackmore, Rajamanoharan, and Williams (2008) and Cheong and Manchester (2015), where inputs are designed to minimize the probability of incorrect diagnosis. A similar approach for nonlinear systems is given in Streif, Petzke, Mesbah, Findeisen, and Braatz (2014). Several multi-model formulations with deterministic bounds on the measurement and process noises have also been proposed. Interestingly, these either provide an input that is guaranteed to identify the correct model within a specified time horizon, or conclude that no such input exists. The article (Nikoukhah & Campbell, 2006) considers noises that are energy-bounded within ellipsoids. Pointwise-in-time polytopic bounds are considered in Nikoukhah (1998), but costly computations with high-dimensional

☆ The material in this paper was presented at the 12th European Control Conference, July 17–19, 2013, Zurich, Switzerland. This paper was recommended for publication in revised form by Associate Editor Brett Ninness under the direction of Editor Torsten Söderström.

E-mail addresses: davide.raimondo@unipv.it (D.M. Raimondo), groberto.marseglia@gmail.com (G. Roberto Marseglia), braatz@mit.edu (R.D. Braatz), jks9@clemson.edu (J.K. Scott).

http://dx.doi.org/10.1016/j.automatica.2016.07.033
0005-1098/

polytopes are required. More recently, an efficient method using pointwise bounds described by zonotopes was developed, making it tractable solve diagnosis problems with high dimension and/or multiple fault models (Scott, Findeisen, Braatz, & Raimondo, 2014). Extensions of the preceding approaches include methods for nonlinear systems (Andjelkovic, Sweetingham, & Campbell, 2008; Paulson, Raimondo, Braatz, Findeisen, & Streif, 2014), hybrid stochastic–deterministic approaches (Marseglia, Scott, Magni, Braatz, & Raimondo, 2014; Scott, Marseglia, Magni, Braatz, & Raimondo, 2013), methods with input and robust state constraints (Andjelkovic & Campbell, 2011; Scott et al., 2014), and robust MPC with diagnosis constraints (Raimondo, Marseglia, Braatz, & Scott, 2013).

The above approaches are all open-loop in the sense that the computed active input is applied with no online modification. The design of exogenous active inputs for closed-loop systems has been studied (Ashari, Nikoukhah, & Campbell, 2012a,b). However, the feedback law was given *a priori*, not designed for fault diagnosis. In Stoican, Olaru, Seron, and De Doná (2012), the authors propose a fault tolerant control method which relies on the computation of invariant sets and a reference governor scheme to isolate faults. While the feedback law was also given a priori, the reference was suitably chosen to guarantee the separation of the residual sets for the healthy and faulty dynamics. In Niemann, Stoustrup, and Poulsen (2014), a given feedback controller is temporarily modified to make a residual more sensitive to faults. A closed-loop approach for stochastic models is presented in Puncochar, Siroky, and Simandl (2015) and Simandl and Puncochar (2009), where the input minimizes nominal control objectives and risks associated with incorrect diagnosis. Finally, a deterministic closed-loop approach using polytopes is described in Tabatabaeipour (2015).

In this context, the present article makes three main contributions. First, the open-loop input design method in Scott et al. (2014) is generalized. This method provides guaranteed diagnosis for linear multi-model systems with initial conditions, disturbances, and measurement noises bounded pointwise by zonotopes. Here, the *constrained zonotope* computations recently proposed in Scott, Marseglia, Raimondo, and Braatz (2016) are used to extend this approach to general polytopic uncertainties, while maintaining the efficiency of the original approach (see Section 3). Second, a new closed-loop input design method is developed by applying the open-loop method of Section 3 within a moving horizon framework, where online measurements are incorporated through set-valued observers (Section 4). This method potentially provides much less conservative active inputs on average (e.g., reduced length, norm), while maintaining the guarantee of fault diagnosis within a given time horizon. Among existing closed-loop approaches, such a guarantee is only provided by the method in Tabatabaeipour (2015). However, that method uses polytope projection operations that scale exponentially in the system dimension (Althoff, Stursberg, & Buss, 2010; Fukuda, 2004). Numerical experiments in Scott, Findeisen, Braatz, and Raimondo (2013) clearly show that such projections are intractable for systems with more than 2 or 3 states. In contrast, our use of constrained zonotopes here avoids this computation completely. Our third contribution is a method for computing an explicit feedback law off-line for cases where computing open-loop inputs online is prohibitive (Section 5). This is enabled by the use of *finite-memory* set-valued observers, at the cost of some additional conservatism. Compared to our preliminary results in Raimondo, Braatz, and Scott (2013), the closed-loop approaches here use more effective observers (enabled by the developments of Section 3), and the explicit method is generalized to address the case of incomplete state measurements.

1.1. Problem formulation

Consider a discrete-time system with time k, state $\mathbf{x}_k \in \mathbb{R}^{n_x}$, output $\mathbf{y}_k \in \mathbb{R}^{n_y}$, input $\mathbf{u}_k \in \mathbb{R}^{n_u}$, disturbance $\mathbf{w}_k \in \mathbb{R}^{n_w}$, and measurement error $\mathbf{v}_k \in \mathbb{R}^{n_v}$. In each interval $[k, k+1]$, $k = 0, 1, \ldots$, the system evolves according to one of n_m possible linear models. The matrices of these models are distinguished by the argument $i \in \mathbb{I} \equiv \{1, \ldots, n_m\}$:

$$\mathbf{x}_{k+1} = \mathbf{A}(i)\mathbf{x}_k + \mathbf{B}(i)\mathbf{u}_k + \mathbf{r}(i) + \mathbf{B}_w(i)\mathbf{w}_k, \tag{1}$$

$$\mathbf{y}_k = \mathbf{C}(i)\mathbf{x}_k + \mathbf{s}(i) + \mathbf{D}_v(i)\mathbf{v}_k. \tag{2}$$

The model $i = 1$ is nominal, and the rest are faulty. Models representing multiple, simultaneous faults can be included in $\mathbb{I}$ if desired (Scott et al., 2014). The constant vectors $\mathbf{r}(i)$ and $\mathbf{s}(i)$ are used to model additive faults such as sensor and actuator bias. Let $\mathbf{x}_0 \in X_0(i)$ represent any information known about $\mathbf{x}_0$ prior to $k = 0$, given that model $i \in \mathbb{I}$ is active. $X_0(i)$ can depend on i if, e.g., it has been constructed from previous measurements through (2). We assume that $(\mathbf{w}_k, \mathbf{v}_k) \in W \times V$, $\forall k \in \mathbb{N}$, and that W, V, and $X_0(i)$ are bounded convex polytopes. Our objective is to design input sequences that guarantee fault diagnosis over a finite horizon N. Specifically, assuming that one model $i^* \in \mathbb{I}$ is active on $[0, N]$ (i.e., the test interval), we aim to design an input that can identify i^* with certainty, while simultaneously satisfying convex polytopic constraints $\mathbf{u}_k \in U$, $\forall k \in \mathbb{N}$, and minimizing a quadratic cost function. The proposed methods are appropriate for designing short test signals that are applied periodically, or after a fault has been detected but not diagnosed.

2. Preliminaries

2.1. Constrained zonotopes and set operations

The new methods in this article are largely enabled by computations with *constrained zonotopes*, a new class of sets introduced in Scott et al. (2016) as an extension of the zonotopes.

Definition 1. A set $Z \subset \mathbb{R}^n$ is a *constrained zonotope* if there exists $(\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}) \in \mathbb{R}^{n \times n_g} \times \mathbb{R}^n \times \mathbb{R}^{n_c \times n_g} \times \mathbb{R}^{n_c}$ such that

$$Z = \{\mathbf{G}\boldsymbol{\xi} + \mathbf{c} : \|\boldsymbol{\xi}\|_\infty \leq 1,\ \mathbf{A}\boldsymbol{\xi} = \mathbf{b}\}. \tag{3}$$

In contrast to standard zonotopes, Definition 1 permits linear equality constraints on $\boldsymbol{\xi}$. The columns of $\mathbf{G}$ are called the *generators*, $\mathbf{c}$ is the *center*, and $\mathbf{A}\boldsymbol{\xi} = \mathbf{b}$ are the *constraints*. We use the shorthand $Z = \{\mathbf{G}, \mathbf{c}, \mathbf{A}, \mathbf{b}\}$ and $Z = \{\mathbf{G}, \mathbf{c}\}$ for constrained and standard zonotopes, respectively.

Constrained zonotopes are substantially more flexible than zonotopes. Indeed, a central result in Scott et al. (2016) is that Z is a constrained zonotope iff it is a convex polytope; i.e., iff Z is bounded and $\exists(\mathbf{H}, \mathbf{k}) \in \mathbb{R}^{n_h \times n} \times \mathbb{R}^n$ such that Z can be written in the halfspace representation (H-rep) $Z = \{\mathbf{z} \in \mathbb{R}^n : \mathbf{H}\mathbf{z} \leq \mathbf{k}\}$. We refer to (3) as the *constrained generator representation* (CG-rep) of Z. Converting from H- to CG-rep is simple, and while the converse is difficult in general, it is never required in the proposed methods (Scott et al., 2016).

The CG-rep has two primary advantages compared to the H-rep. First, it trivializes the computation of some important set operations. Let $Z, W \subset \mathbb{R}^n$, $Y \subset \mathbb{R}^k$, $\mathbf{R} \in \mathbb{R}^{k \times n}$, and define

$$\mathbf{R}Z \equiv \{\mathbf{R}\mathbf{z} : \mathbf{z} \in Z\}, \tag{4}$$

$$Z \oplus W \equiv \{\mathbf{z} + \mathbf{w} : \mathbf{z} \in Z,\ \mathbf{w} \in W\}, \tag{5}$$

$$Z \cap_{\mathbf{R}} Y \equiv \{\mathbf{z} \in Z : \mathbf{R}\mathbf{z} \in Y\}. \tag{6}$$

Eq. (4) is a linear mapping of Z, (5) is the *Minkowski sum*, and (6) is a generalized intersection that arises in state estimation (see

Section 4.1). With Z, W, and Y in CG-rep, (4)–(6) are given in CG-rep by

$$\mathbf{R}Z = \{\mathbf{R}\mathbf{G}_z, \mathbf{R}\mathbf{c}_z, \mathbf{A}_z, \mathbf{b}_z\}, \tag{7}$$

$$Z \oplus W = \left\{ [\mathbf{G}_z \ \mathbf{G}_w], \mathbf{c}_z + \mathbf{c}_w, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_w \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_w \end{bmatrix} \right\}, \tag{8}$$

$$Z \cap_{\mathbf{R}} Y = \left\{ [\mathbf{G}_z \ \mathbf{0}], \mathbf{c}_z, \begin{bmatrix} \mathbf{A}_z & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_y \\ \mathbf{R}\mathbf{G}_z & -\mathbf{G}_y \end{bmatrix}, \begin{bmatrix} \mathbf{b}_z \\ \mathbf{b}_y \\ \mathbf{c}_y - \mathbf{R}\mathbf{c}_z \end{bmatrix} \right\}. \tag{9}$$

Clearly, these computations can be done efficiently and robustly, even in high dimensions. In contrast, with Z and Y in H-rep, Minkowski sums and linear mappings with singular $\mathbf{R}$ (e.g., polytope projection) both become extremely computationally demanding and numerically unstable in dimensions greater than about 10 (Althoff et al., 2010; Fukuda, 2004). For zonotopes, (4)–(5) can again be computed trivially, but (6) is not a zonotope and its approximation is very difficult (Bravo, Alamo, & Camacho, 2006).

The second advantage of the CG-rep is that efficient and effective methods are available for enclosing a given constrained zonotope within another of lower complexity (i.e., with fewer generators and constraints) (Scott et al., 2016). This availability is essential because the results of (7)–(9) can be more complex than the arguments. Similar reduction techniques are also available for zonotopes, but effective methods for polytopes in H-rep are computationally demanding (Broman & Shensa, 1990). Thus, with the CG-rep limited through reduction techniques, the constrained zonotopes constitute a new class of sets with the ability to balance accuracy and efficiency very effectively.

2.2. Reachable set notation and computations

In this section, we demonstrate the computation of reachable sets for (1)–(2) using constrained zonotopes. Below, a tilde designates a sequence associated with (1)–(2). Specifically, $\tilde{\mathbf{y}} = (\mathbf{y}_0, \dots, \mathbf{y}_k) \in \mathbb{R}^{(k+1)n_y}$, with $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{v}}$ defined similarly, and $\tilde{\mathbf{u}} = (\mathbf{u}_0, \dots, \mathbf{u}_{k-1}) \in \mathbb{R}^{kn_u}$, with $\tilde{\mathbf{w}}$ defined similarly. Moreover, we index subsequences by $\tilde{\mathbf{y}}_{\ell:k} = (\mathbf{y}_\ell, \dots, \mathbf{y}_k)$ for $0 \le \ell \le k$. Define the solution mappings

$$(\phi_k, \psi_k) : \mathbb{R}^{kn_u} \times \mathbb{I} \times \mathbb{R}^{n_x} \times \mathbb{R}^{kn_w} \times \mathbb{R}^{n_v} \to \mathbb{R}^{n_x} \times \mathbb{R}^{n_y}$$

so that $\phi_k(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \mathbf{v}_k)$ and $\psi_k(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \mathbf{v}_k)$ are the state and output of (1)–(2) at k, respectively, given the specified inputs. Strictly, ϕ_k does not depend on $\mathbf{v}_k$, but it is included for convenience. Let $\tilde{\phi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) = (\phi_\ell, \dots, \phi_k)$ and $\tilde{\psi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) = (\psi_\ell, \dots, \psi_k)$, where we have abbreviated $(\phi_j, \psi_j) = (\phi_j, \psi_j)(\tilde{\mathbf{u}}_{0:j-1}, i, \mathbf{x}_0, \tilde{\mathbf{w}}_{0:j-1}, \mathbf{v}_j)$, $\ell \le j \le k$.

For each $i \in \mathbb{I}$ and $\tilde{\mathbf{u}} \in \mathbb{R}^{kn_u}$, define the *reachable state and output sets* on $[\ell, k]$ by

$$\tilde{\mathcal{X}}_{\ell:k}(\tilde{\mathbf{u}}, i) \equiv \{\tilde{\phi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) : (\mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) \in X_0(i) \times \tilde{W} \times \tilde{V}\},$$

$$\tilde{\mathcal{Y}}_{\ell:k}(\tilde{\mathbf{u}}, i) \equiv \{\tilde{\psi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) : (\mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) \in X_0(i) \times \tilde{W} \times \tilde{V}\},$$

where $\tilde{W} = W \times \cdots \times W$ and $\tilde{V} = V \times \cdots \times V$ with k and $k+1$ products, respectively. Dependence on $(X_0(i), \tilde{W}, \tilde{V})$ is omitted for brevity. The reachable state and output sets *at* k are $\mathcal{X}_k(\tilde{\mathbf{u}}, i) \equiv \tilde{\mathcal{X}}_{k:k}(\tilde{\mathbf{u}}, i)$ and $\mathcal{Y}_k(\tilde{\mathbf{u}}, i) \equiv \tilde{\mathcal{Y}}_{k:k}(\tilde{\mathbf{u}}, i)$.

Note that (1)–(2) recursively define matrices $\tilde{\mathbf{A}}(i)$, $\tilde{\mathbf{B}}(i)$, etc., which depend on ℓ and k, such that

$$\tilde{\phi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) = \tilde{\mathbf{A}}(i)\mathbf{x}_0 + \tilde{\mathbf{B}}(i)\tilde{\mathbf{u}} + \tilde{\mathbf{r}}(i) + \tilde{\mathbf{B}}_w(i)\tilde{\mathbf{w}},$$

$$\tilde{\psi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) = \tilde{\mathbf{C}}(i)\tilde{\phi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) + \tilde{\mathbf{s}}(i) + \tilde{\mathbf{D}}_v(i)\tilde{\mathbf{v}}.$$

Using (4)–(6), it follows that

$$\tilde{\mathcal{X}}_{\ell:k}(\tilde{\mathbf{u}}, i) = \tilde{\mathbf{A}}(i)X_0(i) \oplus \tilde{\mathbf{B}}(i)\tilde{\mathbf{u}} \oplus \tilde{\mathbf{r}}(i) \oplus \tilde{\mathbf{B}}_w(i)\tilde{W}, \tag{10}$$

$$\tilde{\mathcal{Y}}_{\ell:k}(\tilde{\mathbf{u}}, i) = \tilde{\mathbf{C}}(i)\tilde{\mathcal{X}}_{\ell:k}(\tilde{\mathbf{u}}, i) \oplus \tilde{\mathbf{s}}(i) \oplus \tilde{\mathbf{D}}_v(i)\tilde{V}. \tag{11}$$

Now, supposing that $X_0(i)$, W, and V are constrained zonotopes, denote $X_0 = \{\mathbf{G}_0(i), \mathbf{c}_0(i), \mathbf{A}_0(i), \mathbf{b}_0(i)\}$, $W = \{\mathbf{G}_W, \mathbf{c}_W, \mathbf{A}_W, \mathbf{b}_W\}$, and $V = \{\mathbf{G}_V, \mathbf{c}_V, \mathbf{A}_V, \mathbf{b}_V\}$. It is readily shown that $\tilde{W}$ is also a constrained zonotope with $\mathbf{c}_{\tilde{W}} = (\mathbf{c}_W, \dots, \mathbf{c}_W)$, $\mathbf{b}_{\tilde{W}} = (\mathbf{b}_W, \dots, \mathbf{b}_W)$, $\mathbf{G}_{\tilde{W}} = \operatorname{diag}(\mathbf{G}_W, \dots, \mathbf{G}_W)$, and $\mathbf{A}_{\tilde{W}} = \operatorname{diag}(\mathbf{A}_W, \dots, \mathbf{A}_W)$, and $\tilde{V}$ is analogous. Thus, the above reachable sets can be computed efficiently using (7)–(8). Moreover, these sets take the form

$$\tilde{\mathcal{X}}_{\ell:k}(\tilde{\mathbf{u}}, i) = \{\mathbf{G}^{\mathcal{X}}_{\ell:k}(i), \overline{\phi}_{\ell:k}(\tilde{\mathbf{u}}, i), \mathbf{A}^{\mathcal{X}}_{\ell:k}(i), \mathbf{b}^{\mathcal{X}}_{\ell:k}(i)\}, \tag{12}$$

$$\tilde{\mathcal{Y}}_{\ell:k}(\tilde{\mathbf{u}}, i) = \{\mathbf{G}^{\mathcal{Y}}_{\ell:k}(i), \overline{\psi}_{\ell:k}(\tilde{\mathbf{u}}, i), \mathbf{A}^{\mathcal{Y}}_{\ell:k}(i), \mathbf{b}^{\mathcal{Y}}_{\ell:k}(i)\}, \tag{13}$$

with the generator matrices $\mathbf{G}^{\mathcal{X}}_{\ell:k}(i) = [\tilde{\mathbf{A}}(i)\mathbf{G}_0(i) \ \tilde{\mathbf{B}}_w(i)\mathbf{G}_{\tilde{W}}]$ and $\mathbf{G}^{\mathcal{Y}}_{\ell:k}(i) = [\tilde{\mathbf{C}}(i)\mathbf{G}^{\mathcal{X}}_{\ell:k}(i) \ \tilde{\mathbf{D}}_v(i)\mathbf{G}_{\tilde{V}}]$, the centers $\overline{\phi}_{\ell:k}(\tilde{\mathbf{u}}, i) = \tilde{\phi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{c}_0(i), \mathbf{c}_{\tilde{W}}, \mathbf{c}_{\tilde{V}})$ and $\overline{\psi}_{\ell:k}(\tilde{\mathbf{u}}, i) = \tilde{\psi}_{\ell:k}(\tilde{\mathbf{u}}, i, \mathbf{c}_0(i), \mathbf{c}_{\tilde{W}}, \mathbf{c}_{\tilde{V}})$, constraint matrices $\mathbf{A}^{\mathcal{X}}_{\ell:k} = \operatorname{diag}(\mathbf{A}_0(i), \mathbf{A}_{\tilde{W}})$ and $\mathbf{A}^{\mathcal{Y}}_{\ell:k} = \operatorname{diag}(\mathbf{A}^{\mathcal{X}}_{\ell:k}, \mathbf{A}_{\tilde{V}})$, and right-hand sides $\mathbf{b}^{\mathcal{X}}_{\ell:k} = (\mathbf{b}_0(i), \mathbf{b}_{\tilde{W}})$ and $\mathbf{b}^{\mathcal{Y}}_{\ell:k} = (\mathbf{b}^{\mathcal{X}}_{\ell:k}, \mathbf{b}_{\tilde{V}})$. Note that the sole $\tilde{\mathbf{u}}$-dependence in these sets is through the centers, which are the 'nominal' state and output vectors of (1)–(2) and are linear in $\tilde{\mathbf{u}}$. This fact is central to the input design procedure developed below.

3. Open-loop input design

This section considers the optimal design of open-loop active inputs for fault diagnosis when $(X_0(i), W, V)$ are constrained zonotopes. The results extend the passive method in Scott et al. (2016) and the active method for zonotopes in Scott et al. (2014). Note that if $(X_0(i), W, V)$ are specified as convex polytopes in H-rep, they can be easily converted to CG-rep as shown in Scott et al. (2016).

Consider an input $\tilde{\mathbf{u}} = (\mathbf{u}_0, \dots, \mathbf{u}_{N-1}) \in \mathbb{R}^{Nn_u}$ and the corresponding output $\tilde{\mathbf{y}} = (\mathbf{y}_0, \dots, \mathbf{y}_N) \in \mathbb{R}^{(N+1)n_y}$. We assume that $\tilde{\mathbf{u}}$ must be synthesized before $\mathbf{y}_0$ is known, although this can be relaxed when the cost of input design is negligible (see Remark 1). For a fixed horizon N, if model i is active on $[0, N]$, then $\tilde{\mathbf{y}}$ must be an element of the reachable output set for the ith model over the time horizon from 0 to N:

$$\tilde{\mathbf{y}} \in \tilde{\mathcal{Y}}_{0:N}(\tilde{\mathbf{u}}, i, X_0(i)). \tag{14}$$

Note that $\tilde{\mathcal{Y}}_{0:N}$ is written here with explicit dependence on $X_0(i)$ for clarity. Conversely, if (14) fails, then i was not active on $[0, N]$. Thus, (14) provides a sharp online test for ruling out models in $\mathbb{I}$. Accordingly, our objective is to find an input guaranteeing that (14) will hold for exactly one $i \in \mathbb{I}$. The required property of $\tilde{\mathbf{u}}$ is, $\forall i, j \in \mathbb{I}$ with $i \ne j$:

$$\tilde{\mathcal{Y}}_{0:N}(\tilde{\mathbf{u}}, j, X_0(j)) \cap \tilde{\mathcal{Y}}_{0:N}(\tilde{\mathbf{u}}, i, X_0(i)) = \emptyset. \tag{15}$$

We remark on the ease of satisfying (15) after Corollary 2.

Definition 2. Let $I \subset \mathbb{I}$ and $X_0^I \equiv \{X_0(i)\}_{i \in I}$. An input $\tilde{\mathbf{u}} \in \mathbb{R}^{Nn_u}$ is said to *separate* (I, X_0^I) *in* N *steps* if (15) holds $\forall i, j \in I$ with $i \ne j$. The set of all such inputs is $\mathcal{S}_N(I, X_0^I)$.

Inputs used for active fault diagnosis should be minimally invasive in some sense. Here, inputs are ranked according to their length first, and an objective function second. Let $\tilde{U}_N = U \times \cdots \times U$ with N products. The smallest integer N such that $\tilde{U}_N \cap \mathcal{S}_N(\mathbb{I}, X_0^{\mathbb{I}}) \ne \emptyset$ is called the *minimum separation horizon for* $(\mathbb{I}, X_0^{\mathbb{I}})$. Assuming $N < \infty$, an *optimal input separating* $(\mathbb{I}, X_0^{\mathbb{I}})$ *in* N *steps* is defined as any solution[1] of

$$\inf\{J_N(\tilde{\mathbf{u}}) : \tilde{\mathbf{u}} \in \tilde{U}_N \cap \mathcal{S}_N(\mathbb{I}, X_0^{\mathbb{I}})\}, \tag{16}$$

where $J_N(\tilde{\mathbf{u}}) = \sum_{\ell=1}^{N-1} \mathbf{u}_\ell^{\mathrm{T}} \mathbf{R} \mathbf{u}_\ell$ and $\mathbf{R}$ is positive semidefinite.

[1] It is clear from (15) that $\mathcal{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$ is open, so we should speak of ε-solutions having objective values within $\varepsilon > 0$ of the infimum.

Solving (16) is challenging because the constraint $\tilde{\mathbf{u}} \in \mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$ is nonconvex (Theorem 1). However, when W, V, and $X_0(i)$, $\forall i \in \mathbb{I}$, are zonotopes, (Scott et al., 2014) shows that (16) can be reformulated as a mixed-integer quadratic program (MIQP) that can be solved efficiently. A key step in that reformulation is to show that, using the notation of Section 2.2,

$$\mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}}) = \{\tilde{\mathbf{u}} : \mathbf{N}(i,j)\tilde{\mathbf{u}} \notin Z(i,j),\ \forall i,j \in \mathbb{I},\ j \neq i\}, \tag{17}$$

where $\mathbf{N}(i,j) \equiv \tilde{\mathbf{C}}(j)\tilde{\mathbf{B}}(j) - \tilde{\mathbf{C}}(i)\tilde{\mathbf{B}}(i)$ and

$$Z(i,j) \equiv \{[\,\mathbf{G}^{\Psi}_{0:N}(i)\ -\mathbf{G}^{\Psi}_{0:N}(j)\,], \overline{\psi}_{0:N}(\mathbf{0}, i) - \overline{\psi}_{0:N}(\mathbf{0}, j)\}. \tag{18}$$

Using the zonotopic structure of $Z(i,j)$, $\mathbf{N}(i,j)\tilde{\mathbf{u}} \notin Z(i,j)$ can be expressed as an inequality constraint on the optimal objective value of a linear program (LP), which converts (16) to a bilevel program that is finally reformulated as an MIQP (see Scott et al. (2014) for details).

Corollary 2 below shows that the form (17) can be recovered, albeit with different matrices $\mathbf{N}(i,j)$ and zonotopes $Z(i,j)$, in the case where W, V, and $X_0(i)$ are constrained zonotopes (see Appendix for proof). With this accomplished, (16) can be solved exactly as in Scott et al. (2014). We first prove a direct analogue of (17) with a constrained zonotope $\mathscr{Z}(i,j)$ appearing in place of $Z(i,j)$.

Theorem 1. *An input $\tilde{\mathbf{u}}$ belongs to $\mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$ iff*

$$\mathbf{N}(i,j)\tilde{\mathbf{u}} \notin \mathscr{Z}(i,j), \quad \forall i,j \in \mathbb{I},\ i \neq j, \tag{19}$$

where

$$\mathscr{Z}(i,j) \equiv \left\{ [\,\mathbf{G}^{\Psi}_{0:N}(i)\ -\mathbf{G}^{\Psi}_{0:N}(j)\,], \overline{\psi}_{0:N}(\mathbf{0}, i) - \overline{\psi}_{0:N}(\mathbf{0}, j), \begin{bmatrix} \mathbf{A}^{\Psi}_{0:N}(i) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^{\Psi}_{0:N}(j) \end{bmatrix}, \begin{bmatrix} \mathbf{b}^{\Psi}_{0:N}(i) \\ \mathbf{b}^{\Psi}_{0:N}(j) \end{bmatrix} \right\}. \tag{20}$$

We now recover the form (17) with higher dimensional zonotopes $Z^+(i,j)$ appearing in place of $Z(i,j)$. Denote $\mathscr{Z}(i,j) = \{\mathbf{G}^{\mathscr{Z}}(i,j), \mathbf{c}^{\mathscr{Z}}(i,j), \mathbf{A}^{\mathscr{Z}}(i,j), \mathbf{b}^{\mathscr{Z}}(i,j)\}$.

Corollary 2. *An input $\tilde{\mathbf{u}}$ belongs to $\mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$ iff*

$$\begin{bmatrix} \mathbf{N}(i,j) \\ \mathbf{0} \end{bmatrix} \tilde{\mathbf{u}} \notin Z^+(i,j) \equiv \left\{ \begin{bmatrix} \mathbf{G}^{\mathscr{Z}}(i,j) \\ \mathbf{A}^{\mathscr{Z}}(i,j) \end{bmatrix}, \begin{bmatrix} \mathbf{c}^{\mathscr{Z}}(i,j) \\ -\mathbf{b}^{\mathscr{Z}}(i,j) \end{bmatrix} \right\}, \tag{21}$$

for all $i,j \in \mathbb{I}$, $i \neq j$.

Note that, since $Z^+(i,j)$ is compact, there must exist $\tilde{\mathbf{u}} \in \mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$ provided only that $\mathbf{N}(i,j) \neq \mathbf{0}$; i.e., models i and j do not have identical output controllability matrices.

The optimization (16) can now be solved as follows. First, compute $\tilde{\mathbf{B}}(i)$, $\tilde{\mathbf{C}}(i)$, and $\tilde{\Psi}_{0:N}(\mathbf{0}, i, X_0(i))$ as in Section 2.2, which provides everything necessary to form $\mathbf{N}(i,j)$ and $\mathscr{Z}(i,j)$ for all $i,j \in \mathbb{I}$. Next, $Z^+(i,j)$ is formed from $\mathscr{Z}(i,j)$ via (21), which characterizes $\mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$ in the same mathematical form as (17), so that the algorithm in Scott et al. (2014) can be used to solve (16) as an MIQP. This algorithm can efficiently accommodate polytopic state constraints in (16) for each $i \in \mathbb{I}$, which are omitted here for brevity. Finally, this optimization can be repeated with N increasing from 1 until a feasible program is generated or some maximum N is exceeded.

A significant advantage of the above MIQP reformulation is that the number of binary variables is proportional to the total number of generators in the zonotopes defining $\mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$. When $(X_0(i), W, V)$ are zonotopes, complexity can be managed using existing methods for reducing the number of generators in each $Z(i,j)$ (Althoff et al., 2010). This approach overestimates $Z(i,j)$, leading to a restriction of (16), and hence an optimal input that is feasible in (16) but has a larger objective value than necessary for separation. Case studies in Scott, Findeisen et al. (2013) show that this approach achieves large efficiency gains with minor added conservatism. In the present case, a similar reduction in the number of binary variables could be achieved by applying zonotope reduction methods to each $Z^+(i,j)$. However, even reduction to a first-order zonotope (i.e., a parallelotope) leaves as many generators as the dimension of the set, which is much larger for $Z^+(i,j)$ than $Z(i,j)$; i.e., $((N+1)n_y + 2n_c)$ versus $(N+1)n_y$, where n_c is the number of constraints in each $\tilde{\Psi}_{0:N}(\mathbf{0}, i, X_0(i))$. A more flexible approach is to reduce $\mathscr{Z}(i,j)$ using constrained zonotope techniques that can reduce both the number of generators and constraints (Scott et al., 2016), and to form $Z^+(i,j)$ only subsequently. The most aggressive reduction possible with this scheme takes $\mathscr{Z}(i,j)$ to a first-order $(N+1)n_y$-dimensional zonotope. Remarkably, this results in an optimization problem with complexity that is independent of n_w, n_v, or n_x. Finally, (Scott et al., 2014) suggests several other complexity reduction methods that are readily extended, including tests to eliminate pairs of models satisfying (15) for all feasible $\tilde{\mathbf{u}}$ prior to optimization.

4. Closed-loop input design

In the previous section, it was observed that an open-loop input $\tilde{\mathbf{u}}$ will provide a complete fault diagnosis via the test (14) if and only if $\tilde{\mathbf{u}} \in \mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$. Moreover, a method was outlined for computing an optimal element of this set. This section considers the same input design problem in closed-loop; i.e., under the assumption that $\mathbf{u}_{K+1}$ can be synthesized during the time interval $[K, K+1]$ with full knowledge of the measurements $\tilde{\mathbf{y}}_{0:K}$. As before, the objective is to guarantee that (14) will hold for exactly one $i \in \mathbb{I}$. However, in the closed-loop setting, the requirement (15) is too strong because $\tilde{\mathbf{y}}_{0:K}$ provides additional information about the initial states, disturbances, and measurement noises consistent with each model. Thus, diagnosis can potentially be achieved by an input that is not an element of $\mathscr{S}_N(\mathbb{I}, X_0^{\mathbb{I}})$, and is better than any such element in terms of length and/or objective value.

The following development shows that, at each $0 \leq K < N$, the information provided by the measurements $\tilde{\mathbf{y}}_{0:K}$ can be expressed in terms of the states $\hat{X}_{K+1|K}(i)$ of a bank of set-valued observers. Consequently, a sufficient condition for an updated input $\tilde{\mathbf{u}}_{K+1:N-1}$ to ensure diagnosis via (14) can be expressed as the open-loop condition $\tilde{\mathbf{u}}_{K+1:N-1} \in \mathscr{S}_{N-K-1}(\mathbb{I}, \hat{X}^{\mathbb{I}}_{K+1|K})$, where $\hat{X}^{\mathbb{I}}_{K+1|K} \equiv \{\hat{X}_{K+1|K}(i)\}_{i\in\mathbb{I}}$. Note that set-valued observers have been used extensively for passive fault diagnosis (Ingimundarson, Bravo, Puig, Alamo, & Guerra, 2009; Rosa, Silvestre, Shamma, & Athans, 2010; Seron, Zhuo, De Dona, & Martinez, 2008), and in Scott et al. (2014) to reduce the cost of computing open-loop active inputs when N is large. However, the input was not updated online, so the approach here is distinct. Furthermore, this section extends the preliminary results in Raimondo, Braatz et al. (2013) by enabling the use of improved observers via the developments in Section 2.1 and Section 3.

4.1. Set-valued observers and the online update

The state of a set-valued observer at $K+1$ is a set $\hat{X}_{K+1|K}$ that contains all $\mathbf{x}_{K+1}$ consistent with (1)–(2), the constraints $\mathbf{x}_0 \in X_0(i)$ and $(\mathbf{w}_k, \mathbf{v}_k) \in W \times V$, and the measured outputs $\mathbf{y}_k$ for all $k \leq K$. The subscript $K+1|K$ indicates that the observer state uses all information available at time K, and distinguishes $\hat{X}_{K+1|K}$ from its refinement based on $\mathbf{y}_{K+1}$, $\hat{X}_{K+1|K+1}$. For each $i \in \mathbb{I}$, we

use a set-valued observer defined recursively through the relations (dropping the i's for brevity)

$$\hat{X}_{k|k} \supset \hat{X}_{k|k-1} \cap \{\mathbf{x}_k : \mathbf{C}\mathbf{x}_k \in (\mathbf{y}_k - \mathbf{s}) \oplus (-\mathbf{D}_v)V\}, \tag{22}$$

$$\hat{X}_{k+1|k} \supset \mathbf{A}\hat{X}_{k|k} \oplus \mathbf{B}\mathbf{u}_k \oplus \mathbf{r} \oplus \mathbf{B}_w W, \tag{23}$$

with $\hat{X}_{0|-1} \equiv X_0(i)$. We assume in Section 4 that constrained zonotopic enclosures $X_0(i)$ are available prior to $k = 0$, and address the case where no such information is available in Remark 2 after the developments of Section 5.1. Note that the sensitivity of the observer to $X_0(i)$ diminishes with time and depends on the size of the measurement error set V.

An observer is called *exact* if (22)–(23) hold with equality, and *conservative* if at least one superset relation is strict. In order to execute the required set operations, $\hat{X}_{k|k}$ and $\hat{X}_{k+1|k}$ are typically represented by simple classes of sets such as ellipsoids, parallelotopes, or zonotopes (Bravo et al., 2006; Chisci, Garulli, & Zappa, 1996; Rosa et al., 2010). Often, some of the required operations do not have exact representations within the class and must be overapproximated, leading to a conservative observer (e.g., zonotope intersection in Bravo et al., 2006).

Here, we use constrained zonotopic observers constructed by directly applying (7)–(9) to the operations in (22)–(23). Note that the intersection in (22) is exactly (9) with $\mathbf{R} = \mathbf{C}$ and $Y = (\mathbf{y}_k - \mathbf{s}) \oplus (-\mathbf{D}_v)V$. In principle, this observer is exact. In practice, however, the complexity of the exact observer states quickly becomes prohibitive. This fact severely limits the utility of existing exact observers using polytopes in H-rep (Rosa et al., 2010; Tabatabaeipour, 2015). Using the CG-rep, we avoid this issue by reducing $\hat{X}_{k+1|k}$ in each step to a target number of generators and constraints using the methods in Scott et al. (2016). Although this approach makes our observer conservative, comparisons in Scott et al. (2016) show that its tradeoff between cost and complexity is significantly better than existing methods.

Theorem 4 (see the Appendix for proof) provides the theoretical basis for updating an active input online at $0 \le K < N$ based on the states of set-valued observers, $\hat{X}_{K+1|K}(i)$, $\forall i \in \mathbb{I}$. Specifically, we prove that an updated input $\tilde{\mathbf{u}}_{K+1:N-1}$ ensures diagnosis via (14) if it separates $(\mathbb{I}, \hat{X}^{\mathbb{I}}_{K+1|K})$ in $N-K-1$ steps. This requires the following lemma, which shows that (14) can only hold if the future output sequence $\tilde{\mathbf{y}}_{K+1:N}$ lies in the set of outputs reachable from $\hat{X}_{K+1|K}(i)$ in $N - K - 1$ steps. For any $k \ge -1$, let $\hat{\Phi}_{k+1}(\tilde{\mathbf{u}}_{0:k}, \tilde{\mathbf{y}}_{0:k}, i)$ denote the state $\hat{X}_{k+1|k}(i)$ of (22)–(23) given $\tilde{\mathbf{u}}_{0:k}$ and $\tilde{\mathbf{y}}_{0:k}$.

Lemma 3. *Choose any $i \in \mathbb{I}$, $\tilde{\mathbf{u}} \in \mathbb{R}^{n_u N}$, $\tilde{\mathbf{y}} \in \mathbb{R}^{n_y(N+1)}$, and $0 \le K < N$, and define $(\tilde{\mathbf{u}}_-, \tilde{\mathbf{u}}_+) \equiv (\tilde{\mathbf{u}}_{0:K}, \tilde{\mathbf{u}}_{K+1:N-1})$ and $(\tilde{\mathbf{y}}_-, \tilde{\mathbf{y}}_+) \equiv (\tilde{\mathbf{y}}_{0:K}, \tilde{\mathbf{y}}_{K+1:N})$. Then*

$$\tilde{\mathbf{y}} \in \tilde{\Psi}_{0:N}(\tilde{\mathbf{u}}, i, X_0(i)) \Longrightarrow \tilde{\mathbf{y}}_+ \in \tilde{\Psi}_{0:N-K-1}(\tilde{\mathbf{u}}_+, i, \hat{X}_{K+1|K}(i)),$$

where $\hat{X}_{K+1|K}(i) \equiv \hat{\Phi}_{K+1}(\tilde{\mathbf{u}}_-, \tilde{\mathbf{y}}_-, i)$. The converse holds for the exact observer.

Theorem 4. *Using the notation of Lemma 3,*

$$\tilde{\mathbf{u}}_+ \in \mathscr{S}_{N-K-1}(\mathbb{I}, \hat{X}^{\mathbb{I}}_{K+1|K}) \Longrightarrow \tag{24}$$

$$\tilde{\mathbf{y}} \in \tilde{\Psi}_{0:N}(\tilde{\mathbf{u}}, i, X_0(i)) \text{ for at most one } i \in \mathbb{I}.$$

The converse holds for the exact observer.

4.2. The direct method

In light of Theorem 4, a closed-loop input could be constructed by computing updated open-loop inputs within a moving horizon framework as follows. Prior to time $k = 0$, the set-valued observers (22)–(23) are initialized via $\hat{X}_{0|-1}(i) \equiv X_0(i)$, $\forall i \in \mathbb{I}$. Then, a minimum separation horizon N and an optimal input $\tilde{\mathbf{u}}_{0:N-1}$ separating $(\mathbb{I}, X_0^{\mathbb{I}})$ in N steps are computed as in Section 3. Now, consider a generic time $k \ge 0$, with previously computed data $\mathbf{u}_k$ and $\hat{X}_{k|k-1}(i)$ available. At time k, $\mathbf{u}_k$ is injected, $\mathbf{y}_k$ is observed, $\hat{X}_{k+1|k}(i)$ is computed via (22)–(23) for each $i \in \mathbb{I}$, and a new optimal input $\tilde{\mathbf{u}}_{k+1:N-1}$ is computed that separates $(\mathbb{I}, \hat{X}^{\mathbb{I}}_{k+1|k})$ in at most $N - k - 1$ steps. At $k + 1$, $\mathbf{u}_{k+1}$ is injected and the process is repeated.

A drawback of this scheme is that it does not consider that the updated input may be worse than the previous input. This cannot occur if exact observers are used, but does happen with the inexact observers used here. To see that a previous input can be better, let $\tilde{\mathbf{u}}_{k:N-1|k-1}$ denote the optimal open-loop input computed at time $k - 1$ (as indicated by the additional subscript $|k-1$). At time k, $\mathbf{u}_{k|k-1}$ is injected, leaving the *shifted* sequence $\tilde{\mathbf{u}}_{k+1:N-1|k-1}$. If implemented, the shifted sequence would guarantee diagnosis at N by design. Even so, the shifted sequence does not necessarily separate $(\mathbb{I}, \hat{X}^{\mathbb{I}}_{k+1|k})$ when an inexact observer is used because the observer states can be more conservative at k than they were are $k - 1$, when the shifted sequence was designed (as the converse of Theorem 4 does not hold). Thus, the attempt to compute a better input sequence at k by finding the optimal input separating $(\mathbb{I}, \hat{X}^{\mathbb{I}}_{k+1|k})$ may fail because either no such input exists, or the best among these inputs is not better than the shifted sequence. The below algorithm fixes this problem by retaining the shifted sequence in such cases (Step 9).

Algorithm 1 describes the complete moving horizon input design method, where $\tilde{\mathbf{u}}^{CL}$ denotes the synthesized closed-loop input sequence, L denotes the last time at which the input sequence was *updated* (see Step 9), and k and L are absolute times, whereas N is a horizon; i.e., the time required for diagnosis since the last input update.

Algorithm 1 (*Input:* $\mathbb{I}, W, V, X_0^{\mathbb{I}}$).

(1) Compute the minimum horizon N and an optimal $\tilde{\mathbf{u}}_{0:N-1|-1}$ separating $(\mathbb{I}, X_0^{\mathbb{I}})$ in N steps.
(2) Set $k = 0, L = -1$.
(3) Inject $\mathbf{u}_k^{CL} := \mathbf{u}_{k|L}$.
(4) Measure $\mathbf{y}_k$.
(5) Remove i from $\mathbb{I}$ if

$$\tilde{\mathbf{y}}_{L+1:k} \notin \tilde{\Psi}_{L+1:k}(\tilde{\mathbf{u}}^{CL}_{L+1:k-1}, i, \hat{X}_{L+1|L}(i)). \tag{25}$$

(6) If $k = L + 1 + N$, terminate with output $\mathbb{I}_{\text{out}} := \mathbb{I}$.
(7) For each $i \in \mathbb{I}$, compute $\hat{X}_{k+1|k}(i)$ via (22)–(23).
(8) Compute the minimum horizon N^* and an optimal input $\tilde{\mathbf{u}}_{k+1:k+N^*|k}$ separating $(\mathbb{I}, \hat{X}^{\mathbb{I}}_{k+1|k})$ in N^* steps.
(9) If $\tilde{\mathbf{u}}_{k+1:k+N^*|k}$ is shorter than the shifted sequence $\tilde{\mathbf{u}}_{k+1:L+N|L}$, or if it is the same length and has a smaller objective value, assign $N := N^*$ and $L := k$.
(10) Assign $k := k + 1$ and go to Step 3.

The next theorem is proven in the Appendix.

Theorem 5. *Let $\overline{N}$ denote the value of N computed in Step 1 of Algorithm 1. If $\overline{N} < \infty$, then Algorithm 1 terminates in $\beta \le \overline{N}$ steps with $|\mathbb{I}_{\text{out}}| = 1$, and $i \notin \mathbb{I}_{\text{out}}$ implies*

$$\tilde{\mathbf{y}}_{0:\beta} \notin \tilde{\Psi}_{0:\beta}(\tilde{\mathbf{u}}^{CL}_{0:\beta-1}, i, X_0(i)). \tag{26}$$

Remark 1. In Algorithm 1, $\mathbf{u}^{CL}_{k+1}$ is computed before $\mathbf{y}_{k+1}$ becomes available (i.e., during the interval $[k, k+1]$), which is different from (Raimondo, Braatz et al., 2013), where $\mathbf{u}^{CL}_{k+1}$ depends on $\mathbf{y}_{k+1}$. The latter is sensible only when the online computation time is much faster than the sampling time, which is unlikely here but may be possible for the explicit method described in Section 5.

5. Explicit method

The online cost of the direct method is dominated by the computation of the new separating input in Step 8. This section moves this computation off-line at the expense of a weaker set-valued observer, leading to an explicit control law that approximates the action of the direct method. In Raimondo, Braatz et al. (2013), an explicit method was presented for the case where $\mathbf{C}(i)$ is invertible for all $i \in \mathbb{I}$. The solution here requires only observability and enables the use of more accurate observers.

Assumption 1. For each $i \in \mathbb{I}$, (1)–(2) is observable and $\Omega + 1$ is the maximum observability index over all models. Moreover, sets $\overline{U} \subset \mathbb{R}^{n_u}$ and $\overline{Y} \subset \mathbb{R}^{n_y}$ are known that contain all permissible inputs and all possible outputs, respectively, for all $i \in \mathbb{I}$ and $k \in \mathbb{N}$.

Consider the computation of the updated sequence at time K. The difficulty in moving this computation offline is that the sets $\hat{X}_{K+1|K}(i)$ are not known *a priori*. These sets depend on the sets $X_0(i)$, which are known *a priori*, but also on the sequences $\tilde{\mathbf{u}}_- \equiv \tilde{\mathbf{u}}_{0:K}$ and $\tilde{\mathbf{y}}_- \equiv \tilde{\mathbf{y}}_{0:K}$. To circumvent this, consider a partition of $\overline{U} \times \overline{Y}$ into hyperrectangles $P_\sigma \equiv U_\sigma \times Y_\sigma$ indexed by $\sigma \in S \equiv \{1, \dots, n_\sigma\}$. Each permissible pair $(\tilde{\mathbf{u}}_-, \tilde{\mathbf{y}}_-)$ can now be assigned to a set of the form $P_{\sigma_{0:K}} \equiv P_{\sigma_0} \times \cdots \times P_{\sigma_K}$. The basic idea behind the explicit method is compute offline a set of sequences $\tilde{\mathbf{u}}$ indexed by $\sigma_{0:K}$, each of which is appropriate for any pair $(\tilde{\mathbf{u}}_-, \tilde{\mathbf{y}}_-)$ in $P_{\sigma_{0:K}}$. Then, the online computation is reduced to simply looking up which $\sigma_{0:K}$ the measurements belong to.

Two problems are evident. First, the number of input sequences to be computed offline for use at K is $|S|^{K+1}$, which is unmanageable for large $|S|$ and K. Second, because we consider a finite diagnosis horizon, the inputs prepared for use at K are different from those needed at $K+1$. We address both problems here by designing observers whose states at $K+1$ depend only on $(\mathbf{u}_{K-j}, \mathbf{y}_{K-j})$ with $0 \le j \le J$, and do not depend on $X_0(i)$ at all.

5.1. Finite-memory set-valued observers

Let Ω be as in Assumption 1 and choose any integers $K \ge \Omega$ and $J \in [\Omega, K]$. Our aim is to construct an observer $\hat{X}_{K+1|K}(i)$ based only on the measurements $\tilde{\mathbf{y}}_{K-J:K}$ and knowledge of the corresponding inputs $\tilde{\mathbf{u}}_{K-J:K}$. Using the notation of Section 2.2 and omitting the i-dependence for convenience, we have

$$\mathbf{x}_{K+1} = \phi_{J+1}(\tilde{\mathbf{u}}_{K-J:K}, \mathbf{x}_{K-J}, \tilde{\mathbf{w}}_{K-J:K}, \tilde{\mathbf{v}}_{K-J:K+1}), \tag{27}$$

$$\tilde{\mathbf{y}}_{K-J:K} = \tilde{\psi}_{0:J}(\tilde{\mathbf{u}}_{K-J:K-1}, \mathbf{x}_{K-J}, \tilde{\mathbf{w}}_{K-J:K-1}, \tilde{\mathbf{v}}_{K-J:K}). \tag{28}$$

These equations can be readily written in matrix form,

$$\mathbf{P}\begin{bmatrix}\mathbf{x}_{K+1}\\ \mathbf{x}_{K-J}\end{bmatrix} = \mathbf{Q}\begin{bmatrix}\tilde{\mathbf{u}}_{K-J:K}\\ \tilde{\mathbf{y}}_{K-J:K}\end{bmatrix} + \mathbf{R}\begin{bmatrix}\tilde{\mathbf{w}}_{K-J:K}\\ \tilde{\mathbf{v}}_{K-J:K}\end{bmatrix} + \mathbf{d} \tag{29}$$

where, e.g., $\mathbf{P} = \begin{bmatrix}\mathbf{I} & -\mathbf{A}^{J+1}\\ \mathbf{0} & -\tilde{\mathbf{C}}\tilde{\mathbf{A}}\end{bmatrix}$ with $\tilde{\mathbf{C}}$ and $\tilde{\mathbf{A}}$ as in Section 2.2. Since $J \ge \Omega$, Assumption 1 implies that $\tilde{\mathbf{C}}\tilde{\mathbf{A}}$ has full column rank, and hence so does $\mathbf{P}$. Thus, we may apply elementary row operations to (29) by forming the augmented matrix $[\mathbf{P}|\mathbf{Q}\ \mathbf{R}\ \mathbf{d}]$ and taking $\mathbf{P}$ to its reduced row echelon form $\begin{bmatrix}\mathbf{I}\\ \mathbf{0}\end{bmatrix}$ by Gauss elimination, where the zero matrix is $[(J+1)n_y - n_x] \times 2n_x$. This puts (29) in the form

$$\begin{bmatrix}\mathbf{x}_{K+1}\\ \mathbf{x}_{K-J}\\ \mathbf{0}\end{bmatrix} = \begin{bmatrix}\hat{\mathbf{Q}}_1\\ \hat{\mathbf{Q}}_2\\ \hat{\mathbf{Q}}_3\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{u}}_{K-J:K}\\ \tilde{\mathbf{y}}_{K-J:K}\end{bmatrix} + \begin{bmatrix}\hat{\mathbf{R}}_1\\ \hat{\mathbf{R}}_2\\ \hat{\mathbf{R}}_3\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{w}}_{K-J:K}\\ \tilde{\mathbf{v}}_{K-J:K}\end{bmatrix} + \begin{bmatrix}\hat{\mathbf{d}}_1\\ \hat{\mathbf{d}}_2\\ \hat{\mathbf{d}}_3\end{bmatrix}. \tag{30}$$

Since these operations eliminate the dependence of $\mathbf{x}_{K+1}$ on $\mathbf{x}_{K-J}$, the second block row in (30) is no longer needed. For fixed values of $(\tilde{\mathbf{u}}_{K-J:K}, \tilde{\mathbf{y}}_{K-J:K})$, the first block row can now be used to compute an enclosure of $\mathbf{x}_{K+1}$, noting that $(\tilde{\mathbf{w}}_{K-J:K}, \tilde{\mathbf{v}}_{K-J:K}) \in \tilde{W} \times \tilde{V}$. Moreover, the third block row imposes further constraints on the possible values of $(\tilde{\mathbf{w}}_{K-J:K}, \tilde{\mathbf{v}}_{K-J:K})$ that can be used to sharpen this enclosure using the CG-rep. More generally, consider the case where $(\tilde{\mathbf{u}}_{K-J:K}, \tilde{\mathbf{y}}_{K-J:K})$ is not known exactly, but is known to lie in some partition element $U_\sigma \times Y_\sigma$ with $\sigma \equiv \sigma_{0:J}$ of length $J+1$. Then,

$$\begin{bmatrix}\mathbf{x}_{K+1}\\ \mathbf{0}\end{bmatrix} \in \begin{bmatrix}\hat{\mathbf{Q}}_1\\ \hat{\mathbf{Q}}_3\end{bmatrix}(U_\sigma \times Y_\sigma) \oplus \begin{bmatrix}\hat{\mathbf{R}}_1\\ \hat{\mathbf{R}}_3\end{bmatrix}\left(\tilde{W} \times \tilde{V}\right) \oplus \begin{bmatrix}\hat{\mathbf{d}}_1\\ \hat{\mathbf{d}}_3\end{bmatrix}. \tag{31}$$

Since the hyperrectangles U_σ and Y_σ can be written in G-rep, the CG-rep of the right-hand side of (31) is easily computed using (7)–(8). For brevity, denote this set simply by $\{\mathbf{G}^\sigma, \mathbf{c}^\sigma, \mathbf{A}^\sigma, \mathbf{b}^\sigma\}$. Then, $(\mathbf{x}_{K+1}, \mathbf{0}) \in \{\mathbf{G}^\sigma, \mathbf{c}^\sigma, \mathbf{A}^\sigma, \mathbf{b}^\sigma\}$, and partitioning $\mathbf{G}^\sigma$ and $\mathbf{c}^\sigma$ into block rows accordingly, Proposition 3 in Scott et al. (2016) shows that this relation is equivalent to

$$\mathbf{x}_{K+1} \in \hat{X}^\sigma_{K+1:K}(i) \equiv \left\{\mathbf{G}^\sigma_1, \mathbf{c}^\sigma_1, \begin{bmatrix}\mathbf{A}^\sigma\\ \mathbf{G}^\sigma_2\end{bmatrix}, \begin{bmatrix}\mathbf{b}^\sigma\\ -\mathbf{c}^\sigma_2\end{bmatrix}\right\}. \tag{32}$$

This step provides the desired observer state $\hat{X}^\sigma_{K+1|K}$ in CG-rep. If a zonotopic observer is desired, the constraints can simply be dropped, which is equivalent to dropping the third block row in (30) and is done in all numerical experiments with zonotopic observers herein. An alternative option is eliminate the constraints from (32) using the constrained zonotope reduction techniques in Scott et al. (2016). The constrained zonotopic observers used herein compute $\hat{X}^\sigma_{K+1|K}$ as in (32) and subsequently reduce the number of constraints and generators to target values using these techniques.

We call $J+1$ the *memory* of the observer, which must be at least $\Omega + 1$. The observer becomes more accurate as J increases. On the other hand, the first observer state only becomes available at $k = J$. In the meantime, no active input can be computed, and we simply apply $\mathbf{u}_k = \mathbf{0}$ for $k \le J$.

Remark 2. The development of the direct method in Section 4.2 assumes that the polytopes $X_0(i)$ are known. The preceding developments can be used to eliminate this assumption as follows. The input is set to $\mathbf{0}$ for $0 \le k \le \Omega$. At $k = \Omega$, enough information is available to compute $\hat{X}^\sigma_{\Omega+1|\Omega}(i)$, $\forall i \in \mathbb{I}$, as discussed above. In this computation, the right-hand side of (31) uses the fixed values $(\tilde{\mathbf{u}}_{K-J:K}, \tilde{\mathbf{y}}_{K-J:K})$ in place of the set $U_\sigma \times Y_\sigma$. The observer (22)–(23) is then initialized using $\hat{X}^\sigma_{\Omega+1|\Omega}(i)$ and Algorithm 1 proceeds as written.

5.2. The explicit method

Consider applying the finite-memory observer with memory $J+1$ for each $i \in \mathbb{I}$, and denote the states $\hat{X}^\sigma_{K+1|K}(i)$ for all $i \in \mathbb{I}$ and $\boldsymbol{\sigma} \in S^{J+1} \equiv S \times \cdots \times S$. An essential feature of these observers is that the states $\hat{X}^\sigma_{K+1|K}(i)$ can be computed entirely offline, and provide valid enclosures of the states $\mathbf{x}_{K+1}$ consistent with each $i \in \mathbb{I}$ and any $(\tilde{\mathbf{u}}_{K-J:K}, \tilde{\mathbf{y}}_{K-J:K})$ in the partition element $U_\sigma \times Y_\sigma$. Therefore, an optimal input separating $(\mathbb{I}, \hat{X}^{\sigma,\mathbb{I}}_{K+1|K})$ can be computed offline for each partition element $\boldsymbol{\sigma} \in S^{J+1}$.

Algorithm 1 is modified as follows. The input $\mathbf{u}^{CL}_k$ is set to $\mathbf{0}$ for $0 \le k \le J$. Now, consider any generic time $k \ge J$ with a pre-computed input $\mathbf{u}^{CL}_k$ available. At time k, Steps 3–5 are executed as written. Steps 7–8 are accomplished by simply looking up the partition element $\boldsymbol{\sigma} \in S^{J+1}$ such that $(\tilde{\mathbf{u}}_{K-J:K}, \tilde{\mathbf{y}}_{K-J:K})$ belongs to $U_\sigma \times Y_\sigma$, and retrieving the pre-computed observer states $\hat{X}_{K+1|K}(i) \equiv \hat{X}^\sigma_{K+1|K}(i)$ and input $\tilde{\mathbf{u}}^{OL}_{k+1:k+N^*|k} \equiv \tilde{\mathbf{u}}(\sigma)$. The remainder

of the algorithm is unaltered. This procedure defines an explicit control law, and we refer to Algorithm 1 with these modifications as the *explicit method*.

The explicit method has a clear computational advantage over the direct method, but it uses a weaker observer. Thus, updated inputs will be accepted less frequently in Step 9, thereby diminishing the advantages of a closed-loop input design procedure. Moreover, J measurements must be accumulated before the active input sequence can begin. Finally, note that Step 5 eliminates some $i \in \mathbb{I}$ prior to termination, which leads to the ability to compute improved input sequences in subsequent steps. To take full advantage of this ability in the explicit method, it is necessary to compute additional input sequences for every partition element that separate, respectively, every possible subset of models in $\mathbb{I}$.

6. Numerical examples

6.1. Example 1

The first example compares the open-loop input design method in Scott et al. (2014), which only permits $X_0(i)$, W, and V to be zonotopes, with its extension in Section 3, which permits arbitrary convex polytopes using constrained zonotope computations. Our goal is to show that significant performance gains can be achieved with only a minor increase in complexity. Consider the three models defined by

$$\mathbf{A}(1) = \begin{bmatrix} 0.6 & 0.2 \\ -0.2 & 0.7 \end{bmatrix}, \ \mathbf{A}(2) = \begin{bmatrix} 0.6 & 0 \\ -0.2 & 0.7 \end{bmatrix}, \ \mathbf{A}(3) = \begin{bmatrix} 0.6 & 0.2 \\ 0 & 0.7 \end{bmatrix},$$

and, for all $i \in \mathbb{I} \equiv \{1, 2, 3\}$,

$$\mathbf{B}(i) = \begin{bmatrix} -0.3861 & 0.1994 \\ -0.1994 & 0.3861 \end{bmatrix}, \ \mathbf{B}_w(i) = \begin{bmatrix} 0.1215 & 0.0598 \\ 0.0598 & 0.1215 \end{bmatrix},$$
$$\mathbf{C}(i) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ \mathbf{D}_v(i) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ \mathbf{r}(i) = \mathbf{s}(i) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Model $i = 1$ is nominal, while 2 and 3 have system faults. The measurement noise is a zero-centered interval, given in G-rep as $V = \{0.2\mathbf{I}_{2\times 2}, \mathbf{0}\}$. In contrast, $X_0(i)$ and W are chosen to be constrained zonotopes for all $i \in \mathbb{I}$:

$$W = \left\{ \begin{bmatrix} 0.5 & 0 & 0.5 \\ 0 & 0.5 & 0.5 \end{bmatrix}, \ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ [\,1\ 0.5\ 0.5\,], \ -1 \right\}, \tag{33}$$

$$X_0(i) = \left\{ \begin{bmatrix} 0.1 & 0 & 0.1 \\ 0 & 0.1 & 0.1 \end{bmatrix}, \ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \ [\,1\ 0.5\ 0.5\,], \ 1 \right\}. \tag{34}$$

The sets $X_0(i)$ are generally the results of observers, e.g., as in Section 5.1, and may not be zonotopes. Similarly, W is often a polytope when it is used to bound unmodeled nonlinear dynamics (Dang, Le Guernic, & Maler, 2011). Finally, we require $\|\mathbf{u}_k\|_\infty \leq 3$.

To apply the zonotopic method in Scott et al. (2014), W and $X_0(i)$ are overapproximated by zonotopes by dropping the constraints in (33)–(34). As described in Section 3, this method conservatively reduces the number of generators in each $Z(i, j)$ to a fixed value prior to optimization. We chose $(N + 1)n_y$ here. Similarly, in the method of Section 3, $\mathscr{Z}(i, j)$ is reduced to 2 constraints and $(N + 1)n_y + 2$ generators. Fig. 1 clearly shows that the use of constrained zonotopes leads to better performance in both the length and norm of the input required to guarantee diagnosis. Additionally, the constrained zonotope-based approach also requires less time (0.01 s versus 0.13 s[2]). This counter-intuitive result is due to the fact that the constrained zonotope-based approach provides a diagnosis with a much shorter horizon N (if the horizons were the same, the computation time of the zonotope-based approach would be lower).

[2] Laptop PC (Intel i7, 2.8 GHz, 8 GB RAM) running Windows 7 and using a single core; optimization using CPLEX 12.4.

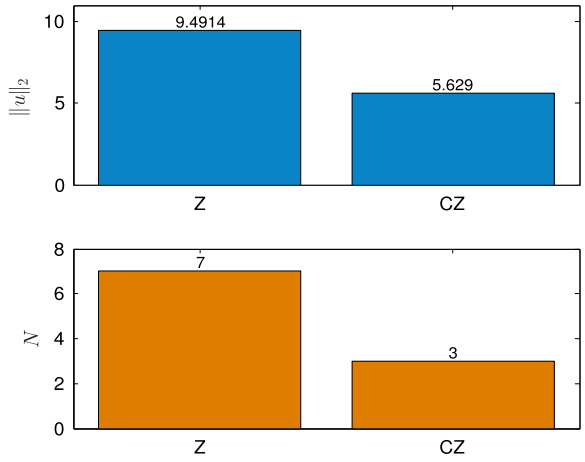


Fig. 1. Length and norm of open-loop inputs required for diagnosis using zonotopes and constrained zonotopes in Example 1.

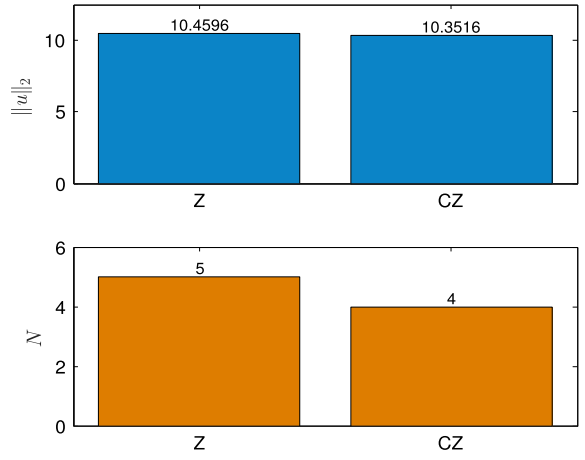


Fig. 2. Length and norm of open-loop inputs required for diagnosis using zonotopes and constrained zonotopes in Example 2.

6.2. Example 2

To show the scaling of the proposed methods, we repeat the first comparison for a 10-state model with $\mathbf{B}(i) = 0.5\mathbf{I}_{10\times 10}$, $\mathbf{B}_w(i) = 0.05\mathbf{I}_{10\times 10}$, and $\mathbf{C}(i) = \mathbf{D}_v(i) = \mathbf{I}_{10\times 10}$, $\forall i \in \mathbb{I} \equiv \{1, 2, 3\}$. Moreover,

$$\mathbf{A}(1) = \begin{bmatrix} 0.5 & 0.5 & -0.5 & -0.5 & 0 & 0 & 1 & 1 & -0.1 & 0.1 \\ 0.5 & 0.5 & -0.5 & -0.5 & 1 & 1 & 0 & 0 & -0.1 & 0.1 \\ -0.5 & 0.5 & -0.5 & -0.5 & 0 & 0 & 1 & 0 & -0.1 & 0.1 \\ 0.5 & -0.5 & -0.5 & 0.5 & 1 & 0 & 1 & 0 & 0.1 & -0.1 \\ -0.5 & -0.5 & -0.5 & -0.5 & 0 & 0 & 1 & 1 & 0.1 & 0.1 \\ 0.5 & 0.5 & -0.5 & 0.5 & 1 & 0 & 0 & 1 & 0.1 & 0.1 \\ 0.5 & -0.5 & 0.5 & -0.5 & 1 & 0 & 0 & 1 & -0.1 & -0.1 \\ -0.5 & 0.5 & 0.5 & -0.5 & 1 & 0 & 0 & 0 & 0.1 & 0.1 \\ 0.5 & 0.5 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0.1 & -0.1 \\ -0.5 & -0.5 & -0.5 & 0.5 & 1 & 1 & 1 & 1 & 0.1 & -0.1 \end{bmatrix},$$

and $\mathbf{A}(2) = \mathbf{A}(3) = \mathbf{A}(1)$ with the modifications $a_{j,1}(2) = 0$ for all $j \in \{1, \ldots, 5\}$, and $a_{jk}(3) = 0$ for all $j \in \{8, 9, 10\}$ and $k \in \{9, 10\}$. $A(2)$ and $A(3)$ are chosen to differ from $A(1)$ in only a few entries in order to make the separation problem challenging. Finally, $V = \{0.1\mathbf{I}_{10\times 10}, \mathbf{0}\}$ and, letting $\mathbf{1}$ be a column vector of 10 ones, $X_0(i) = \{0.4\,[\mathbf{I}_{10\times 10}\ \mathbf{1}], \mathbf{0}\}$, $\forall i \in \mathbb{I}$, and $W = \left\{0.6\,[\mathbf{I}_{10\times 10}\ \mathbf{1}_{10}], \mathbf{0}, 0.5\mathbf{1}^{\mathrm{T}}, -1\right\}$. As in Example 1, each $Z(i, j)$ is reduced to $(N+1)n_y$ generators and each $\mathscr{Z}(i, j)$ to two constraints and $(N + 1)n_y + 2$ generators.

Fig. 2 shows that the new constrained zonotope method outperforms the zonotope method again. Remarkably, it is also

Table 1
Fault model parameters.

Model	R_a (Ω)	$L \times 10^{-3}$ (H)	$K_e \times 10^{-2}$ (V rad/s)	$J_1 \times 10^{-4}$ (N m s^2/rad)	$f_r \times 10^{-4}$ (N m s/rad)
1	1.2030	5.5840	8.5740	1.3528	2.3396
2	1.7725	5.5837	8.0203	1.3320	2.3769
3	1.4365	8.7548	7.7020	1.4185	4.1279

substantially more efficient, taking only 5.92 s compared to 129 s. This is because the horizon required for diagnosis is reduced from 5 to 4, and due to the complexity of the considered models, this leads to a dramatic reduction in computational effort.

6.3. Example 3

Next, the closed-loop input design method of (Raimondo, Braatz et al., 2013) for zonotopic $(X_0(i), W, V)$ is compared with its extension in Section 4, which enables more accurate observers using constrained zonotopes. consider the following model of a low-frequency permanent-magnet DC motor, where the input u is the armature voltage, the states are the current i and motor speed n, and the parameters R_a, L, K_e, K_t, J_1, and f_r are, respectively, the resistance, inductance, torque constant, back EMF constant, motor inertia, and friction coefficient:

$$\begin{bmatrix} \frac{di(t)}{dt} \\ \frac{dn(t)}{dt} \end{bmatrix} = \begin{bmatrix} -R_a/L & -K_e/L \\ K_t/J_1 & -f_r/J_1 \end{bmatrix} \begin{bmatrix} i(t) \\ n(t) \end{bmatrix} + \begin{bmatrix} 1/L \\ 0 \end{bmatrix} u(t),$$
$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}.$$

To keep the motor speed near 70.3 rad/s, the nominal input was set to $u_c = 6$ V, which was modeled by adding $\mathbf{r}(i) = \mathbf{B}(i)u_c$ to the state equations. The active input u_s was added to u_c, subject to the constraint $|u_s| \leq 6$ V. Table 1 gives the parameter values for the nominal and two faulty models, corresponding respectively to an increase of armature resistance and a disconnection of a coil from the commutator bar (Liu, Zhang, Liu, & Yang, 2000) (K_t is always determined by $K_t = 1.0005K_e$).

All models were discretized by forward Euler with a sampling interval of 5 ms to obtain models of the form:

$$\mathbf{x}_k = \mathbf{A}(i)\mathbf{x}_{k-1} + \mathbf{B}(i)\mathbf{u}_{k-1} + \mathbf{B}_w(i)\mathbf{w}_{k-1}, \tag{35}$$
$$\mathbf{y}_k = \mathbf{C}(i)\mathbf{x}_k + \mathbf{D}_v(i)\mathbf{v}_k, \tag{36}$$

where $i = 1, 2, 3$ distinguishes the nominal and faulty models. Specifically, $\mathbf{A}(i)$, $\mathbf{B}(i)$, and $\mathbf{C}(i)$ were obtained from discretization, and the measurement and process noise terms were added with $\mathbf{D}_v(i) = \mathbf{I}$, $i = 1, 2$. The matrices $\mathbf{B}_w(i)$ were obtained assuming 5% uncertainty in R_a, K_e, J_1, and f_r, and computing the worst-case additive error when the current and the motor speed are bounded in, respectively, $[-2, 2]$ A and $[-150, 150]$ rad/s:

$$\mathbf{B}_w(1) = \begin{bmatrix} -0.0254 & -0.0778 \\ -0.3996 & 0.3026 \end{bmatrix}, \quad \mathbf{B}_w(2) = \begin{bmatrix} -0.0231 & -0.0471 \\ -0.3470 & 0.2798 \end{bmatrix},$$
$$\mathbf{B}_w(3) = \begin{bmatrix} -0.0282 & -0.0589 \\ -0.3926 & 0.1684 \end{bmatrix}.$$

We assume $X_0(i) \equiv \left\{ \begin{bmatrix} 0.3 & 0 \\ 0 & 3 \end{bmatrix}, \begin{bmatrix} 0.6 \\ 70 \end{bmatrix} \right\}$, $V \equiv \left\{ \begin{bmatrix} 0.3 & 0 \\ 0 & 3 \end{bmatrix}, \mathbf{0} \right\}$, and $W \equiv \left\{ \begin{bmatrix} 0.8 & 0 & 0.8 & 0.4 \\ 0 & 0.8 & 0.8 & 0.4 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & -1 & 0.3 & 1 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix} \right\}$. The sets $X_0(i)$ are assumed to be unknown for the purposes of input design, and are only used for simulating sample trajectories. The observers used for closed-loop input design are initialized online using the measurements $(\mathbf{y}_0, \mathbf{y}_1)$ with $\mathbf{u}_0 = \mathbf{u}_1 = \mathbf{0}$ (see Remark 2). As in the previous examples, each $Z(i, j)$ is reduced to $(N + 1)n_y$ generators and each $\mathscr{Z}(i, j)$ to two constraints and $(N + 1)n_y + 2$ generators.

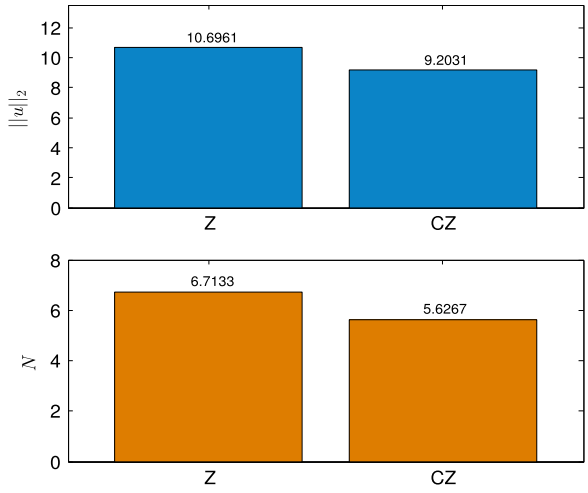


Fig. 3. Mean length and norm of closed-loop inputs required for diagnosis using zonotopic and constrained zonotopic observers in Example 3.

Fig. 3 shows the mean length and norm of the optimal input required for fault diagnosis for both closed-loop methods. Again, the use of constrained zonotopes lead to better performance in terms of both length and norm.

6.4. Example 4

The last example compares the explicit closed-loop approach of Section 5 with the direct closed-loop approach of Section 4 and the open-loop approach of Section 3. Consider the models defined by

$$\mathbf{A}(1) = \begin{bmatrix} 0.6 & 0.2 & 0 \\ -0.2 & 0.7 & 0 \\ 0 & 0.1 & 0.2 \end{bmatrix}, \tag{37}$$

$\mathbf{A}(2) = \mathbf{A}(3) = \mathbf{A}(1)$ with the modifications $a_{1,2}(2) = 0$ and $a_{2,1}(3) = 0$, and, for all $i \in \{1, 2, 3\}$,

$$\mathbf{B}(i) = \begin{bmatrix} -0.3861 & 0.1994 \\ -0.1994 & 0.3861 \\ 0 & 0.12 \end{bmatrix}, \ \mathbf{B}_w(i) = \begin{bmatrix} 0.1215 & 0.0598 \\ 0.0598 & 0.1215 \\ 0.01 & 0.002 \end{bmatrix}, \tag{38}$$
$$\mathbf{C}(i) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0.2 \end{bmatrix}, \ \mathbf{D}_v(i) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ \mathbf{r}(i) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ \mathbf{s}(i) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Note that these models do not have full state measurements. The extension of the explicit method to such systems is an important contribution of this article relative to the preliminary results in Raimondo, Braatz et al. (2013). Define $V = \{0.2\mathbf{I}_{2\times 2}, \mathbf{0}\}$,

$$W = \left\{ \begin{bmatrix} 0.1 & 0 & 0.1 \\ 0 & 0.1 & 0.1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix}, [\,1\ 0.5\ 0.5\,], -1 \right\}, \tag{39}$$
$$X_0(i) = \left\{ \begin{bmatrix} 0.1 & 0 & 0 & 0.1 \\ 0 & 0.1 & 0 & 0.1 \\ 0 & 0 & 0.1 & 0.1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, [\,1\ 0.5\ 0.5\ 0.5\,], 1 \right\}. \tag{40}$$

Finally, we impose the constraint $\|\mathbf{u}_k\|_\infty \leq 1$.

The initial sets $X_0(i)$ are assumed to be unknown for the purposes of input design. Rather, all methods inject $\mathbf{u}_0 = \mathbf{0}$, and the measurements $(\mathbf{y}_0, \mathbf{y}_1)$ are used to compute the initial observer

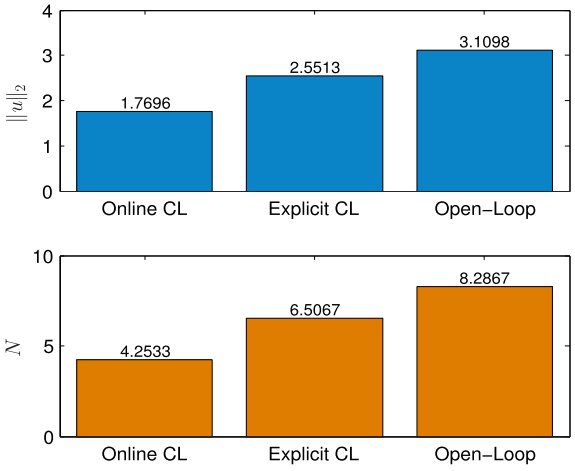


Fig. 4. Mean length and norm of inputs required for diagnosis in Example 4 using the open-loop, direct closed-loop, and explicit closed-loop methods developed in this article.

states $\hat{X}_{1|1}(i)$. To achieve a consistent comparison with the closed-loop methods, these initial states are used even for the open-loop method, which computes an input sequence beginning with $\mathbf{u}_1$. For the direct closed-loop approach, the complexity of the optimization that must be solved online requires that $\mathbf{u}_k$ is computed before $\mathbf{y}_k$ becomes available (i.e., during the interval $[k-1, k]$). Thus, this method must additionally inject $\mathbf{u}_1 = \mathbf{0}$, predict $\hat{X}_{2|1}(i)$, and produce its first nonzero control input in time to inject at $k = 2$. In contrast, for the explicit closed-loop approach, we assume that the online computation time is much faster than the sampling time, so that $\mathbf{u}_k$ can be computed with knowledge of $\mathbf{y}_k$. Thus, the first nonzero control input for this method is $\mathbf{u}_1$. Accordingly, we pre-computed separating inputs on a grid on the 6-dimensional space of the variables $(\mathbf{y}_0, \mathbf{y}_1, \mathbf{u}_0)$ only. The grid covers the bounded interval $[-1, 1]$ with four uniform subintervals in each input dimension, and covers the interval $[-1.5, 1.5]$ with 6 uniform subintervals in each output dimension. We assume that sets $\overline{U} \subset \mathbb{R}^{n_u}$ and $\overline{Y} \subset \mathbb{R}^{n_y}$ contain all permissible inputs and all possible outputs, respectively, for all $i \in \mathbb{I}$ and $k \in \mathbb{N}$ (see Assumption 1). The total number of cells in 6-dimensions is $6^4 \times 4^2 = 20{,}736$. Inputs were computed on this grid for the task of separating all three models, as well as for the tasks of separating each combination of two models. The latter were used to make use of Step 5 in Algorithm 1 prior to the terminal time. The total cost of pre-computing all inputs was 7.6 h on a single core (see specs in Example 1). The cost of computing the input for separating all three models on a single cell averaged 0.97 s. In all methods, each $\mathscr{Z}(i, j)$ is reduced to two constraints and $(N+1)n_y + 2$ generators.

Fig. 4 shows that the explicit method outperforms the open-loop method in both the mean length and norm of the inputs required for diagnosis. On the other hand, the closed-loop direct approach still provides the best performance, even though $\mathbf{u}_{k+1}$ is computed without knowledge of $\mathbf{y}_{k+1}$. As expected, the explicit method provides some of the advantages of the direct closed-loop approach when the cost of online optimization is infeasible.

7. Conclusions

A deterministic active FDI method based on constrained zonotopes (CZs) was proposed in this paper. The use of CZs in the optimization allows to consider general polytopic uncertainties while maintaining the efficiency of the zonotope-based schemes. The benefits have been demonstrated also in closed loop where CZs permit to naturally capture the information coming from online measurements, and lead to faster and less expensive diagnosis. Finally, the use of finite memory observers allows, also for the case of incomplete state measurements, to explicitly solve the input design problem, thus dramatically reducing the online cost of the proposed methods, at the cost of some additional conservatism.

Appendix. Proofs

Proof of Theorem 1. From (9) with $\mathbf{R} = \mathbf{I}$, the intersection on the left-hand side of (15) is a constrained zonotope with constraints

$$\begin{bmatrix} \mathbf{A}^{\psi}_{0:N}(i) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^{\psi}_{0:N}(j) \\ \mathbf{G}^{\psi}_{0:N}(i) & -\mathbf{G}^{\psi}_{0:N}(j) \end{bmatrix} \xi = \begin{bmatrix} \mathbf{b}^{\psi}_{0:N}(i) \\ \mathbf{b}^{\psi}_{0:N}(j) \\ \overline{\psi}_{0:N}(\tilde{\mathbf{u}}, j) - \overline{\psi}_{0:N}(\tilde{\mathbf{u}}, i) \end{bmatrix}. \tag{41}$$

Thus, (15) holds iff $\nexists \xi$ satisfying $\|\xi\|_\infty \leq 1$ and (41). Equivalently, (15) holds iff $\overline{\psi}_{0:N}(\tilde{\mathbf{u}}, j) - \overline{\psi}_{0:N}(\tilde{\mathbf{u}}, i)$ is not an element of the constrained zonotope

$$\left\{ \begin{bmatrix} \mathbf{G}^{\psi}_{0:N}(i) & -\mathbf{G}^{\psi}_{0:N}(j) \end{bmatrix}, \mathbf{0}, \begin{bmatrix} \mathbf{A}^{\psi}_{0:N}(i) & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^{\psi}_{0:N}(j) \end{bmatrix}, \begin{bmatrix} \mathbf{b}^{\psi}_{0:N}(i) \\ \mathbf{b}^{\psi}_{0:N}(j) \end{bmatrix} \right\}. \tag{42}$$

Using $\overline{\psi}_{0:N}(\tilde{\mathbf{u}}, i) = \tilde{\mathbf{C}}(i)\tilde{\mathbf{B}}(i)\tilde{\mathbf{u}} + \overline{\psi}_{0:N}(\mathbf{0}, i)$ gives (19). □

Proof of Corollary 2. (21) $\iff$ (19) by Proposition 3 in Scott et al. (2016). □

The below lemma is used in the proof of Lemma 3.

Lemma 6. *Let $i \in \mathbb{I}$, $K \geq 0$, $\tilde{\mathbf{u}}_{0:K} \in \mathbb{R}^{n_u(K+1)}$, $\tilde{\mathbf{y}}_{0:K} \in \mathbb{R}^{n_y(K+1)}$, and $\mathbf{x}_{K+1} \in \mathbb{R}^{n_x}$. If $\exists(\mathbf{x}_0, \tilde{\mathbf{w}}_{0:K}, \tilde{\mathbf{v}}_{0:K}) \in X_0 \times \tilde{W} \times \tilde{V}$ such that*

$$\tilde{\mathbf{y}}_{0:K} = \tilde{\psi}_{0:K}(\tilde{\mathbf{u}}_{0:K-1}, i, \mathbf{x}_0, \tilde{\mathbf{w}}_{0:K-1}, \tilde{\mathbf{v}}_{0:K}), \tag{43}$$

$$\mathbf{x}_{K+1} = \phi_{K+1}(\tilde{\mathbf{u}}_{0:K}, i, \mathbf{x}_0, \tilde{\mathbf{w}}_{0:K}, \tilde{\mathbf{v}}_{0:K}), \tag{44}$$

then $\mathbf{x}_{K+1} \in \hat{\Phi}_{K+1}(\tilde{\mathbf{u}}_{0:K}, \tilde{\mathbf{y}}_{0:K}, i)$. The converse holds for the exact observer.

Proof. Denote $\hat{X}_{k+1|k} \equiv \hat{\Phi}_{k+1}(\tilde{\mathbf{u}}_{0:k}, \tilde{\mathbf{y}}_{0:k}, i)$, $-1 \leq k \leq K$. Assume $\exists(\mathbf{x}_0, \tilde{\mathbf{w}}_{0:K}, \tilde{\mathbf{v}}_{0:K})$ as in the hypothesis and define $\tilde{\mathbf{x}}_{0:K} \equiv \tilde{\phi}_{0:K}(\tilde{\mathbf{u}}_{0:K-1}, i, \mathbf{x}_0, \tilde{\mathbf{w}}_{0:K-1}, \tilde{\mathbf{v}}_{0:K})$. For any $0 \leq k \leq K$, if $\mathbf{x}_k \in \hat{X}_{k|k-1}$, then (43) ensures that $\mathbf{x}_k \in \hat{X}_{k|k}$, and hence $\mathbf{x}_{k+1} \in \hat{X}_{k+1|k}$ by the definition of $\mathbf{x}_{k+1}$. Since $\mathbf{x}_0 \in \hat{X}_{0|-1} = X_0(i)$, finite induction gives $\mathbf{x}_{K+1} \in \hat{X}_{K+1|K}$.

Conversely, assume that $\mathbf{x}_{K+1} \in \hat{X}_{K+1|K}$ and the observer is exact. To set up an inductive proof from K to 0, choose any $0 \leq k \leq K$ and any $\mathbf{x}_{k+1} \in \hat{X}_{k+1|k}$. Since (22)–(23) hold with equality, there must exist $(\mathbf{w}_k, \mathbf{v}_k) \in W \times V$ and $\mathbf{x}_k \in \hat{X}_{k|k-1}$ such that (1)–(2) hold. Thus, finite induction provides $\mathbf{x}_0 \in X_0$ and $(\mathbf{w}_k, \mathbf{v}_k) \in W \times V$, $\forall k \in \{0, \ldots, K\}$, such that (1)–(2) hold, which verifies (43)–(44). □

Proof of Lemma 3. If $\tilde{\mathbf{y}} \in \tilde{\Psi}_{0:N}(\tilde{\mathbf{u}}, i, X_0)$, then $\exists(\mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}) \in X_0 \times \tilde{W} \times \tilde{V}$ such that $\tilde{\mathbf{y}} = \tilde{\psi}_{0:N}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}})$. Define $\mathbf{x}_{K+1} \equiv \phi_{K+1}(\tilde{\mathbf{u}}_-, i, \mathbf{x}_0, \tilde{\mathbf{w}}_-, \tilde{\mathbf{v}}_-)$. By the definition of $\tilde{\psi}$, it follows that $\tilde{\mathbf{y}}_+ = \tilde{\psi}_{0:N-k-1}(\tilde{\mathbf{u}}_+, i, \mathbf{x}_{K+1}, \tilde{\mathbf{w}}_+, \tilde{\mathbf{v}}_+)$. But, by Lemma 6, $\mathbf{x}_{K+1} \in \hat{X}_{K+1|K}$, and hence $\tilde{\mathbf{y}}_+ \in \tilde{\Psi}_{0:N-k-1}(\tilde{\mathbf{u}}_+, i, \hat{X}_{K+1|K})$.

Conversely, if $\tilde{\mathbf{y}}_+ \in \tilde{\Psi}_{0:N-k-1}(\tilde{\mathbf{u}}_+, i, \hat{X}_{K+1|K})$, then there exists $(\mathbf{x}_{K+1}, \tilde{\mathbf{w}}_+, \tilde{\mathbf{v}}_+) \in \hat{X}_{K+1|K} \times \tilde{W}_+ \times \tilde{V}_+$ with $\tilde{\mathbf{y}}_+ = \tilde{\psi}_{0:N-k-1}(\tilde{\mathbf{u}}_+, i, \mathbf{x}_{K+1}, \tilde{\mathbf{w}}_+, \tilde{\mathbf{v}}_+)$. But, by Lemma 6, $\mathbf{x}_{K+1} \in \hat{X}_{K+1|K}$ implies that $\exists(\mathbf{x}_0, \tilde{\mathbf{w}}_-, \tilde{\mathbf{v}}_-) \in X_0 \times \tilde{W}_- \times \tilde{V}_-$ such that

$$\tilde{\mathbf{y}}_- = \tilde{\psi}_{0:K}(\tilde{\mathbf{u}}_-, i, \mathbf{x}_0, \tilde{\mathbf{w}}_-, \tilde{\mathbf{v}}_-), \tag{45}$$

$$\mathbf{x}_{K+1} = \phi_{K+1}(\tilde{\mathbf{u}}_-, i, \mathbf{x}_0, \tilde{\mathbf{w}}_-, \tilde{\mathbf{v}}_-). \tag{46}$$

Letting, $\tilde{\mathbf{w}} \equiv (\tilde{\mathbf{w}}_-, \tilde{\mathbf{w}}_+)$ and $\tilde{\mathbf{v}} \equiv (\tilde{\mathbf{v}}_-, \tilde{\mathbf{v}}_+)$, it follows that $\tilde{\mathbf{y}} = \tilde{\psi}_{0:N}(\tilde{\mathbf{u}}, i, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}})$, and hence $\tilde{\mathbf{y}} \in \tilde{\Psi}_{0:N}(\tilde{\mathbf{u}}, i, X_0)$. □

Proof of Theorem 4. If $\tilde{\mathbf{u}}_+$ separates $(\mathbb{I}, \hat{X}^{\mathbb{I}}_{K+1|K})$ in $N-K-1$ steps, then by definition $\tilde{\mathbf{y}}_+ \notin \tilde{\Psi}_{0:N-k-1}(\tilde{\mathbf{u}}_+, i, \hat{X}_{K+1|K}(i))$ for all $i \in \mathbb{I}$ except possibly one. By the contrapositive of Lemma 3, this implies that $\tilde{\mathbf{y}} \notin \tilde{\Psi}_{0:N}(\tilde{\mathbf{u}}, i, X_0(i))$ for all i save one. The converse follows from the converse of Lemma 3. □

Proof of Theorem 5. Let T_k denote the value of $L+1+N$ in Step 6. The algorithm terminates when $k = T_k$. Clearly, $T_0 = \overline{N}$, and $T_{k+1} = T_k$ unless N and L are updated in Step 9. An update requires that $k+N^* \leq L+N$, and hence $T_{k+1} = k+1+N^* \leq L+1+N = T_{k+1}$. Thus, T_k is decreasing, and termination must occur for $k = \beta \leq \overline{N}$.

When $k = \beta$, Step 5 is visited and (25) reads

$$\tilde{\mathbf{y}}_{L+1:\beta} \notin \tilde{\Psi}_{L+1:\beta}(\tilde{\mathbf{u}}^{CL}_{L+1:\beta-1}, i, \hat{X}_{L+1|L}(i)), \tag{47}$$

where L is the time of the last update via Step 9. Let $\mathbb{I}_L$ denote $\mathbb{I}$ at the time of that update and note that $\tilde{\mathbf{u}}^{CL}_{L+1:\beta-1} = \tilde{\mathbf{u}}_{L+1:\beta-1|L}$. By design (see Step 8), this implies that (47) holds for all but one $i \in \mathbb{I}_L$, and hence $|\mathbb{I}_{out}| = 1$.

By Lemma 3, each $i \in \mathbb{I}_L$ satisfying (47), and hence not in $\mathbb{I}_{out}$, must also satisfy (26). Then, it remains to show that (26) also holds for $i \notin \mathbb{I}_L$. If $i \notin \mathbb{I}_L$, then (25) must hold for some $k < \beta$, and applying Lemma 3 with $N := k$ and K equal to the value of L at k implies (26). □

References

Althoff, M., Stursberg, O., & Buss, M. (2010). Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes. *Nonlinear Analysis-Hybrid Systems*, *4*(2), 233–249.

Andjelkovic, I., & Campbell, S. L. (2011). Direct optimization determination of auxiliary test signals for linear problems with model uncertainty. In *Proc. 50th IEEE conference on decision and control* (ppp. 909–914).

Andjelkovic, I., Sweetingham, K., & Campbell, S. L. (2008). Active fault detection in nonlinear systems using auxiliary signals. In *Proc. American control conference* (pp. 2142–2147).

Ashari, A., Nikoukhah, R., & Campbell, S. L. (2012a). Active robust fault detection in closed-loop systems: Quadratic optimization approach. *IEEE Transactions on Automatic Control*, *57*(10), 2532–2544.

Ashari, A., Nikoukhah, R., & Campbell, S. L. (2012b). Effects of feedback on active fault detection. *Automatica*, *48*(5), 866–872.

Blackmore, L., Rajamanoharan, S., & Williams, B. C. (2008). Active estimation for jump Markov linear systems. *IEEE Transactions on Automatic Control*, *53*(10), 2223–2236.

Bravo, J. M., Alamo, T., & Camacho, E. F. (2006). Bounded error identification of systems with time-varying parameters. *IEEE Transactions on Automatic Control*, *51*(7), 1144–1150.

Broman, V., & Shensa, M. J. (1990). A compact algorithm for the intersection and approximation of n-dimensional polytopes. *Mathematics and Computers in Simulation*, *32*(5), 469–480.

Cheong, S., & Manchester, I. R. (2015). Input design for discrimination between classes of LTI models. *Automatica*, *53*(5), 103–110.

Chisci, L., Garulli, A., & Zappa, G. (1996). Recursive state bounding by parallelotopes. *Automatica*, *32*(7), 1049–1055.

Dang, T., Le Guernic, C., & Maler, O. (2011). Computing reachable states for nonlinear biological models. *Theoretical Computer Science*, *412*(21), 2095–2107.

Fukuda, K. (2004). From the zonotope construction to the Minkowski addition of convex polytopes. *Journal of Symbolic Computation*, *38*(4), 1261–1272.

Gao, Z., Cecati, C., & Ding, S. (2015). A survey of fault diagnosis and fault-tolerant techniques part II: Fault diagnosis with knowledge-based and hybrid/active approaches. *IEEE Transactions on Industrial Electronics*, *62*(6), 3768–3774.

Ingimundarson, A., Bravo, J. M., Puig, V., Alamo, T., & Guerra, P. (2009). Robust fault detection using zonotope-based set-membership consistency test. *International Journal of Adaptive Control and Signal Processing*, *23*(4), 311–330.

Keresteciogl u, F., & Cetin, I. (2004). Optimal input design for the detection of changes towards unknown hypotheses. *International Journal of Systems Science*, *35*(7), 435–444.

Liu, X. Q., Zhang, H. Y., Liu, J., & Yang, J. (2000). Fault detection and diagnosis of permanent-magnet DC motor based on parameter estimation and neural network. *IEEE Transactions on Industrial Electronics*, *47*(5), 1021–1030.

Marseglia, G. R., Scott, J. K., Magni, L., Braatz, R. D., & Raimondo, D. M. (2014). A hybrid stochastic-deterministic approach for active fault diagnosis using scenario optimization. In *Proc. IFAC world congress* (pp. 1102–1107).

Niemann, H. H. (2006). A setup for active fault diagnosis. *IEEE Transactions on Automatic Control*, *51*(9), 1572–1578.

Niemann, H., Stoustrup, J., & Poulsen, N. K. (2014). Controller modification applied for active fault detection. In *Proc. American control conference*.

Nikoukhah, R. (1998). Guaranteed active failure detection and isolation for linear dynamical systems. *Automatica*, *34*(11), 1345–1358.

Nikoukhah, R., & Campbell, S. L. (2006). Auxiliary signal design for active failure detection in uncertain linear systems with a priori information. *Automatica*, *42*(2), 219–228.

Paulson, J. A., Raimondo, D. M., Braatz, R. D., Findeisen, R., & Streif, S. (2014). Active fault diagnosis for nonlinear uncertain systems. In *Proc. European control conference* (pp. 926–931).

Puncochar, I., Siroky, J., & Simandl, M. (2015). Constrained active fault detection and control. *IEEE Transactions on Automatic Control*, *60*(1), 253–258.

Raimondo, D. M., Braatz, R. D., & Scott, J. K. (2013). Active fault diagnosis using moving horizon input design. In *Proc. European control conference* (pp. 3131–3136).

Raimondo, D. M., Marseglia, G. R., Braatz, R. D., & Scott, J. K. (2013). Fault-tolerant model predictive control with active fault isolation. In *Proc. int. conf. control and fault tolerant systems* (pp. 444–449).

Rosa, P., Silvestre, C., Shamma, J. S., & Athans, M. (2010). Fault detection and isolation of LTV systems using set-valued observers. In *Proc. 49th IEEE conference on decision and control* (pp. 768–773).

Scott, J. K., Findeisen, R., Braatz, R. D., & Raimondo, D. M. (2013). Design of active inputs for set-based fault diagnosis. In *Proc. American control conference* (pp. 3567–3572).

Scott, J. K., Findeisen, R., Braatz, R. D., & Raimondo, D. M. (2014). Input design for guaranteed fault diagnosis using zonotopes. *Automatica*, *50*(6), 1580–1589.

Scott, J. K., Marseglia, G. R., Magni, L., Braatz, R. D., & Raimondo, D. M. (2013). A hybrid stochastic-deterministic input design method for active fault diagnosis. In *Proc. 52nd IEEE conference on decision and control* (pp. 5656–5661).

Scott, J. K., Marseglia, G. R., Raimondo, D. M., & Braatz, R. D. (2016). Constrained zonotopes: A new tool for set-based estimation and fault detection. *Automatica*, *69*, 126–136.

Seron, M. M., Zhuo, X. W., De Dona, J. A., & Martinez, J. J. (2008). Multisensor switching control strategy with fault tolerance guarantees. *Automatica*, *44*(1), 88–97.

Simandl, M., & Puncochar, I. (2009). Active fault detection and control: Unified formulation and optimal design. *Automatica*, *45*(9), 2052–2059.

Stoican, Florin, Olaru, Sorin, Seron, María M., & De Doná, José A. (2012). Reference governor design for tracking problems with fault detection guarantees. *Journal of Process Control*, *22*(5), 829–836.

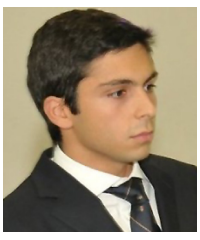
Streif, S., Petzke, F., Mesbah, A., Findeisen, R., & Braatz, R. D. (2014). Optimal experimental design for probabilistic model discrimination using polynomial chaos. In *Proc. IFAC world congress* (pp. 4103–4109).

Tabatabaeipour, S. M. (2015). Active fault detection and isolation of discrete-time linear time-varying systems: a set-membership approach. *International Journal of Systems Science*, *46*(11), 1917–1933.

Tchakoua, P., Wamkeue, R., Ouhrouche, M., Slaoui-Hasnaoui, F., Tameghe, T. A., & Ekemb, G. (2014). Wind turbine condition monitoring: State-of-the-art review, new trends, and future challenges. *Energies*, *7*(4), 2595–2630.

Yu, Y., Woradechjumroen, D., & Yu, D. (2014). A review of fault detection and diagnosis methodologies on air-handling units. *Energy and Buildings*, *82*, 550–562.

Davide M. Raimondo is an associate professor in the Department of Electrical, Computer and Biomedical Engineering at University of Pavia, Italy. He received the Ph.D. in Electronics, Computer Science and Electrical Engineering from the University of Pavia, Italy, in 2009. As a Ph.D. student he held a visiting position at the Department of Automation and Systems Engineering, University of Seville, Spain. From January 2009 to December 2010, he was a postdoctoral fellow in the Automatic Control Laboratory, ETH Zürich, Switzerland. In 2012 (March–June), 2013 (August–September) and 2014 (September–November) he was a visiting scholar in Prof. Braatz Group, Department of Chemical Engineering, MIT, USA. From December 2010 to May 2015, he was an assistant professor at the University of Pavia. He is the author or co-author of more than 60 papers published in refereed journals, edited books, and refereed conference proceedings. He is a subject editor of the Journal of Optimal Control Applications and Methods. His current research interests include model predictive control, active fault diagnosis, fault-tolerant control, distributed control, renewable energy, autonomous surveillance, and control of glycemia in diabetics.



Giuseppe Roberto Marseglia received the Ph.D. in Electronics, Computer Science and Electrical Engineering from the University of Pavia, Italy, in 2016. As a Ph.D. student he held a visiting position at the Braatz Group, Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA. He is the author or co-author of diverse papers published in refereed journals, edited books, and refereed conference proceedings. His current research interests include active fault diagnosis, fault-tolerant control and renewable energy.

Richard D. Braatz is the Edwin R. Gilliland Professor at the Massachusetts Institute of Technology (MIT) where he does research in applied mathematics and control theory and its application to manufacturing processes, biomedical systems, and nanotechnology. He received M.S. and Ph.D. degrees from the California Institute of Technology and was on the faculty at the University of Illinois at Urbana–Champaign and was a visiting scholar at Harvard University before moving to MIT. He has consulted or collaborated with more than 20 companies including United Technologies Corporation, IBM, BP, and Novartis. Honors include the AACC Donald P. Eckman Award, the Antonio Ruberti Young Researcher Prize, the IEEE Control Systems Society Transition to Practice Award, and best paper awards from IEEE- and IFAC-sponsored control journals. He is a Fellow of IEEE and IFAC.

Joseph K. Scott is an assistant professor in the Department of Chemical and Biomolecular Engineering at Clemson University. He received his B.S. (2006) in Chemical Engineering from Wayne State University, and his M.S. (2008) and Ph.D. (2012) in Chemical Engineering from MIT. His honors include the 2012 Best Paper Award for the Journal of Global Optimization and the 2016 Air Force Young Investigator Research Program Award. His research interests include dynamical systems, optimization theory, simulation and optimization of chemical processes, advanced process control, and fault diagnosis. Current applications focus optimization and control of renewable energy systems.