2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

Active Fault Diagnosis using Moving Horizon Input Design*

Davide M. Raimondo[1], Richard D. Braatz[2], and Joseph K. Scott[2]

Abstract— **This article considers the design of an input signal for improving the diagnosability of faults from process measurements. Previous work has focused on open-loop input design. In particular, deterministic methods are available for computing an input that guarantees fault diagnosis within a specified time horizon, whenever such an input exists. Here, two closed-loop approaches are considered that use feedback in order to reduce the length and/or cost of the required input, while maintaining this guarantee. The first method uses an existing open-loop input design method within a receding horizon framework. The second method approximates the first by an explicit feedback law in order to reduce online computations.**

I. Introduction

In chemical processes, aerospace systems, and many other applications, the trend toward increasing complexity and automation has made component malfunctions and other abnormal events (i.e., faults) increasingly frequent and difficult to diagnose [1], [2]. Without corrective action, such faults lead to performance degradation and potentially critical situations. Thus, maintaining safe and profitable operation requires fast and accurate determination of whether a fault has occurred (fault detection) and which component has failed (fault diagnosis). Owing to the confounding effects of disturbances and measurement noise, these tasks are often very difficult and are increasingly handled automatically as part of the supervisory control system.

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. The passive approach is quite mature and is reviewed in [1], [3], [4]. In contrast, the active approach involves injecting a signal into the system to improve the diagnosability of potential faults. Research on active fault diagnosis has been comparatively limited [5], [6], [7], [8], [9].

This article considers a deterministic input design problem for active fault diagnosis. The process of interest, under nominal and various faulty conditions, is described by a set of linear discrete-time models subject to bounded disturbances and measurement noise. In this setting, several methods have been proposed for designing open-loop inputs that guarantee fault diagnosis within a specified time horizon, or concluding that no such input exists [7], [8], [9] (extensions for continuous-time and nonlinear systems can be found in [8], [10], and asymptotically optimal implementations in [11], [12]). For such an input to be useful in practice, it should provide a diagnosis quickly and cause minimal harm with respect to other control objectives. Various optimization formulations have been proposed to address these considerations [8], [9].

The unifying feature of these approaches is that, once a suitable input has been determined, it is applied without modification. In contrast, this article considers closed-loop approaches to active input design. In particular, online measurements are used to refine an open-loop input initially computed off-line. The proposed approaches maintain the guarantee of fault diagnosis within a given time horizon. However, while open-loop active inputs must guarantee diagnosis robustly, closed-loop approaches have the potential to provide much less conservative inputs on average (e.g., reduced length, norm, etc.).

A few previous articles have considered closed-loop strategies for input design in the context of fault diagnosis. In [6], a moving horizon strategy is applied in a stochastic setting to minimize a penalty associated with incorrect diagnoses. In the deterministic setting of this article, incorrect diagnoses are impossible by design, so these problems are distinct. The articles [13] and [14] consider systems under linear feedback control and discuss the design of an exogenous input that guarantees fault diagnosis. However, the feedback law is given *a priori*, not designed for the purposes of fault diagnosis.

II. Problem Statement

Consider a system that obeys one of n_m discrete-time linear time-invariant (LTI) models. One of these models is considered to be nominal, and the rest are faulty. If model $i \in \mathscr{M} \equiv \{1,\dots,n_m\}$ is obeyed, then the states and measurements of the system, denoted $\mathbf{x}_k \in \mathbb{R}^{n_x}$ and $\mathbf{y}_k \in \mathbb{R}^{n_y}$, respectively, evolve according to

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

$$\mathbf{y}_k = \mathbf{C}^{[i]}\mathbf{x}_k + \mathbf{s}^{[i]} + \mathbf{D}_v^{[i]}\mathbf{v}_k. \tag{2}$$

Above, $\mathbf{w}_k \in W \subset \mathbb{R}^{n_w}$ is a disturbance, $\mathbf{v}_k \in V \subset \mathbb{R}^{n_v}$ is a measurement noise, and $\mathbf{u}_k \in U \subset \mathbb{R}^{n_u}$ is the input. It is assumed that U is a convex polytope and W and V are zero-centered zonotopes (see §III-C). The initial state is also assumed to lie in a zonotope $X_0^{[i]} \subset \mathbb{R}^{n_x}$. This zonotope is allowed to be different for each model i, which will occur if past measurements are used to characterize $\mathbf{x}_0$, as in §V. The vectors $\mathbf{r}^{[i]}$ and $\mathbf{s}^{[i]}$ are constant and are used to model additive faults (e.g., actuator or sensor bias).

*BP is acknowledged for financial support of this project.

[1]D. Raimondo is with the Identification and Control of Dynamic Systems Laboratory, University of Pavia davide.raimondo@unipv.it

[2]R. Braatz and J. Scott are with the Massachusetts Institute of Technology {braatz,jkscott}@mit.edu

We are interested in computing an integer N and a control sequence $(\mathbf{u}_0,\dots,\mathbf{u}_{N-1})$ that will generate a sequence of outputs $(\mathbf{y}_1,\dots,\mathbf{y}_N)$ that is consistent with exactly one $i\in\mathscr{M}$. Since each i describes a different fault scenario, this provides a complete fault diagnosis after N steps. Ideally, N should be as small as possible and the control sequence should have further desirable properties, such as a small norm. This input design problem has previously been considered using an open-loop approach (see §IV). In contrast, this article proposes a closed-loop approach in which recursive open-loop calculations are applied within a moving horizon framework (see §V).

III. Preliminaries

A. Notation

A tilde always designates a sequence associated with (1)–(2). For example, $\tilde{\mathbf{u}}=(\mathbf{u}_0,\dots,\mathbf{u}_{N-1})\in\mathbb{R}^{Nn_u}$. For $0\le\ell\le k<N$, we further denote $\tilde{\mathbf{u}}_{\ell:k}=(\mathbf{u}_\ell,\dots,\mathbf{u}_k)$.

For each $i\in\mathscr{M}$ and $k\ge 0$, define the solution map $\phi_k^{[i]}:\mathbb{R}^{kn_u}\times\mathbb{R}^{n_x}\times\mathbb{R}^{kn_w}\to\mathbb{R}^{n_x}$ so that $\phi_k^{[i]}(\tilde{\mathbf{u}},\mathbf{x}_0,\tilde{\mathbf{w}})$ is the state of (1) at k given initial state $\mathbf{x}_0$, input $\tilde{\mathbf{u}}$, and disturbance $\tilde{\mathbf{w}}$. Moreover, define $\tilde{\phi}_{\ell:k}^{[i]}:\mathbb{R}^{kn_u}\times\mathbb{R}^{n_x}\times\mathbb{R}^{kn_w}\to\mathbb{R}^{(k-\ell+1)n_x}$ by $\tilde{\phi}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},\mathbf{x}_0,\tilde{\mathbf{w}})=(\phi_\ell^{[i]},\dots,\phi_k^{[i]})$, where we have abbreviated $\phi_j^{[i]}=\phi_j^{[i]}(\tilde{\mathbf{u}}_{0:j-1},\mathbf{x}_0,\tilde{\mathbf{w}}_{0:j-1})$, $\ell\le j\le k$. Similarly, define the output map $\psi_k^{[i]}:\mathbb{R}^{kn_u}\times\mathbb{R}^{n_x}\times\mathbb{R}^{kn_w}\times\mathbb{R}^{n_v}\to\mathbb{R}^{n_y}$ by $\psi_k^{[i]}(\tilde{\mathbf{u}},\mathbf{x}_0,\tilde{\mathbf{w}},\mathbf{v})=\mathbf{C}^{[i]}\phi_k^{[i]}(\tilde{\mathbf{u}},\mathbf{x}_0,\tilde{\mathbf{w}})+\mathbf{s}^{[i]}+\mathbf{D}_v^{[i]}\mathbf{v}$, and define $\tilde{\psi}_{\ell:k}^{[i]}:\mathbb{R}^{kn_u}\times\mathbb{R}^{n_x}\times\mathbb{R}^{kn_w}\times\mathbb{R}^{(k-\ell+1)n_v}\to\mathbb{R}^{(k-\ell+1)n_y}$ by $\tilde{\psi}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},\tilde{\mathbf{w}},\mathbf{x}_0,\tilde{\mathbf{v}})=(\psi_\ell^{[i]},\dots,\psi_k^{[i]})$, where we have abbreviated $\psi_j^{[i]}=\psi_j^{[i]}(\tilde{\mathbf{u}}_{0:j-1},\mathbf{x}_0,\tilde{\mathbf{w}}_{0:j-1},\mathbf{v}_j)$, $\ell\le j\le k$. From (1)–(2), one can easily derive matrices $\tilde{\mathbf{A}}^{[i]}$, $\tilde{\mathbf{B}}^{[i]}$, etc., which depend on ℓ and k, such that

$$\tilde{\phi}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},\mathbf{x}_0,\tilde{\mathbf{w}})=\tilde{\mathbf{A}}^{[i]}\mathbf{x}_0+\tilde{\mathbf{B}}^{[i]}\tilde{\mathbf{u}}+\tilde{\mathbf{r}}^{[i]}+\tilde{\mathbf{B}}_w^{[i]}\tilde{\mathbf{w}},\tag{3}$$

$$\tilde{\psi}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},\mathbf{x}_0,\tilde{\mathbf{w}},\tilde{\mathbf{v}})=\tilde{\mathbf{C}}^{[i]}\tilde{\phi}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},\mathbf{x}_0,\tilde{\mathbf{w}})+\tilde{\mathbf{s}}^{[i]}+\tilde{\mathbf{D}}_v^{[i]}\tilde{\mathbf{v}}.\tag{4}$$

B. Reachable Sets

For each $i\in\mathscr{M}$, define the *reachable state and output sets*

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

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

where $\tilde{W}=W\times\ldots\times W$ and $\tilde{V}=V\times\ldots\times V$. Explicit dependence on $\tilde{W}$ and $\tilde{V}$ is omitted for brevity. The reachable state and output sets *at* k are defined as $X_k^{[i]}\equiv\tilde{X}_{k:k}^{[i]}$ and $Y_k^{[i]}\equiv\tilde{Y}_{k:k}^{[i]}$.

Explicit formulas for these sets are derived as follows. For any $Q,P\subset\mathbb{R}^n$ and $\mathbf{R}\in\mathbb{R}^{m\times n}$, define the operations

$$\mathbf{R}Q\equiv\{\mathbf{R}\mathbf{q}:\mathbf{q}\in Q\},\tag{5}$$

$$-Q\equiv\{-\mathbf{q}:\mathbf{q}\in Q\},\tag{6}$$

$$Q+P\equiv\{\mathbf{q}+\mathbf{p}:\mathbf{q}\in Q,\ \mathbf{p}\in P\}.\tag{7}$$

It follows from (3)–(4) that

$$\tilde{X}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},X_0)=\tilde{\mathbf{A}}^{[i]}X_0+\tilde{\mathbf{B}}^{[i]}\tilde{\mathbf{u}}+\tilde{\mathbf{r}}^{[i]}+\tilde{\mathbf{B}}_w^{[i]}\tilde{W},\tag{8}$$

$$\tilde{Y}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},X_0)=\tilde{\mathbf{C}}^{[i]}\tilde{X}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},X_0)+\tilde{\mathbf{s}}^{[i]}+\tilde{\mathbf{D}}_v^{[i]}\tilde{V}.\tag{9}$$

To isolate the effect of the inputs on the outputs, define

$$\tilde{F}_{\ell:k}^{[i]}(X_0)\equiv\tilde{\mathbf{C}}^{[i]}\left(\tilde{\mathbf{A}}^{[i]}X_0+\tilde{\mathbf{r}}^{[i]}+\tilde{\mathbf{B}}_w^{[i]}\tilde{W}\right)+\tilde{\mathbf{s}}^{[i]}+\tilde{\mathbf{D}}_v^{[i]}\tilde{V}\tag{10}$$

and note that $\tilde{Y}_{\ell:k}^{[i]}(\tilde{\mathbf{u}},X_0)=\tilde{F}_{\ell:k}^{[i]}(X_0)+\tilde{\mathbf{C}}^{[i]}\tilde{\mathbf{B}}^{[i]}\tilde{\mathbf{u}}$.

C. Zonotopes

The methods in this article involve computations with *zonotopes*, which are centrally symmetric convex polytopes that can be described as Minkowsi sums of line segments [15]. In *generator representation*, a zonotope Z is prescribed by its *center* $\mathbf{c}\in\mathbb{R}^n$ and *generators* $\mathbf{g}_1,\dots,\mathbf{g}_{n_g}\in\mathbb{R}^n$ as $Z=\{\mathbf{G}\xi+\mathbf{c}:\|\xi\|_\infty\le 1\}$, where $\mathbf{G}\equiv[\mathbf{g}_1\ \dots\ \mathbf{g}_{n_g}]$. We use the notation $Z=\{\mathbf{G},\mathbf{c}\}$.

Zonotopes obey the following identities, where $Z=\{\mathbf{G}_z,\mathbf{c}_z\}$ and $Y=\{\mathbf{G}_y,\mathbf{c}_y\}$:

$$\mathbf{R}Z=\{\mathbf{R}\mathbf{G}_z,\mathbf{R}\mathbf{c}_z\},\tag{11}$$

$$-Z=\{\mathbf{G}_z,-\mathbf{c}_z\},\tag{12}$$

$$Z+Y=\{[\mathbf{G}_z\ \mathbf{G}_y],\mathbf{c}_z+\mathbf{c}_y\},\tag{13}$$

$$Z\times Y=\left\{\begin{bmatrix}\mathbf{G}_z & \mathbf{0}\\ \mathbf{0} & \mathbf{G}_y\end{bmatrix},\begin{bmatrix}\mathbf{c}_z\\ \mathbf{c}_y\end{bmatrix}\right\}.\tag{14}$$

In particular, zonotopes are closed under the operations (5)–(7), and the results can be computed very efficiently.

The *order* of a zonotope is defined as n_g/n. The results of (11)-(14) can be higher order than the arguments. To avoid increasing orders, techniques are available for enclosing a given zonotope within a zonotope of lower order. In §VI, Method C in [16] is used.

If X_0, V, and W are zonotopes, then it follows from (8)–(10) that $\tilde{X}_{\ell:k}(\tilde{\mathbf{u}},X_0)$, $\tilde{Y}_{\ell:k}(\tilde{\mathbf{u}},X_0)$, and $\tilde{F}_{\ell:k}^{[i]}(X_0)$ are zonotopes and can be computed efficiently using (11)–(14).

IV. Open-Loop Input Design

An efficient method for computing open-loop active inputs for fault diagnosis was presented in [9]. The main results are summarized in this section, with slight modifications to facilitate the moving horizon approach of the next section.

For each $i\in\mathscr{M}$, suppose that it is known that $\mathbf{x}_0\in X_0^{[i]}$ if model $i\in\mathscr{M}$ is active (e.g., $X_0^{[i]}$ has been computed using past measurements through (2)).

Definition 1: An input $\tilde{\mathbf{u}}\in\mathbb{R}^{Nn_u}$ *separates* $\{(i,X_0^{[i]})\}_{i\in\mathscr{M}}$ *in* N *steps* if

$$\tilde{Y}_{1:N}^{[i]}(\tilde{\mathbf{u}},X_0^{[i]})\cap\tilde{Y}_{1:N}^{[j]}(\tilde{\mathbf{u}},X_0^{[j]})=\emptyset,\quad\forall i,j\in\mathscr{M},\ i\ne j.\tag{15}$$

The set of all such inputs is denoted by $\mathscr{S}_N(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$.

Suppose that $\tilde{\mathbf{u}}\in\mathscr{S}_N(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$ is injected into the system and the output $\tilde{\mathbf{y}}=(\mathbf{y}_1,\dots,\mathbf{y}_N)$ is measured. By definition, the test $\tilde{\mathbf{y}}\in\tilde{Y}_{1:N}^{[i]}(\tilde{\mathbf{u}},X_0^{[i]})$ will fail for all $i\in\mathscr{M}$ except possibly one. Thus, $\tilde{\mathbf{u}}$ either provides the desired fault diagnosis, or concludes that no model in $\mathscr{M}$ is correct.

Theorem 1: $\tilde{\mathbf{u}}\in\mathscr{S}_N(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$ if and only if

$$\tilde{\mathbf{L}}^{[ij]}\tilde{\mathbf{u}}\notin\left(\tilde{F}_{1:N}^{[i]}(X_0^{[i]})+(-\tilde{F}_{1:N}^{[j]}(X_0^{[j]}))\right),\tag{16}$$

for all $i,j\in\mathscr{M}$ with $i\ne j$, where $\tilde{\mathbf{L}}^{[ij]}\equiv\tilde{\mathbf{C}}^{[j]}\tilde{\mathbf{B}}^{[j]}-\tilde{\mathbf{C}}^{[i]}\tilde{\mathbf{B}}^{[i]}$.

The proof uses (10) to show that (15) and (16) are equivalent [9]. The details are omitted.

The importance of Theorem 1 is that, for every $i, j \in \mathscr{M}$ with $i \neq j$, the set on the right-hand side of (16) is a zonotope provided that W, V, $X_0^{[i]}$, and $X_0^{[j]}$ are zonotopes, and can be computed efficiently using (11)–(14) (see §III-C). This provides a simple, efficient, and numerically robust method for characterizing $\mathscr{S}_N(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$.

A. Optimal Separating Inputs

In this article, separating inputs are ranked first according to their length and second according to an objective function. Let $\tilde{U}_Q = U \times \ldots \times U$ with Q products. The smallest integer $Q > 0$ such that $\tilde{U}_Q \cap \mathscr{S}_Q(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$ is nonempty is called the *minimum separation horizon for* $\{(i,X_0^{[i]})\}_{i\in\mathscr{M}}$. Assuming $Q < \infty$, an *optimal input separating* $\{(i,X_0^{[i]})\}_{i\in\mathscr{M}}$ *in* Q *steps* is defined to be any solution[1] of

$$\inf\{J_Q(\tilde{\mathbf{u}}) : \tilde{\mathbf{u}} \in \tilde{U}_Q \cap \mathscr{S}_Q(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})\}, \qquad (17)$$

where $J_Q(\tilde{\mathbf{u}}) = \sum_{\ell=1}^{N-1} \mathbf{u}_\ell^{\mathrm{T}} \mathbf{R} \mathbf{u}_\ell$ and $\mathbf{R} \in \mathbb{R}^{n_u \times n_u}$ is positive semidefinite. Broadly, the purpose of this optimization is to minimize any harmful effects of the separating input with respect to other control objectives.

Suppose Q is known. Because $\mathscr{S}_Q(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$ is defined as the complement of several convex polytopes, (17) is a nonconvex program. However, it was shown in [9] that this program can be reformulated as a mixed-integer quadratic program (MIQP), which can be solved efficiently using, for example, CPLEX [17]. The reformulation involves two essential steps. First, the condition $\tilde{\mathbf{u}} \in \mathscr{S}_Q(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$ is written as a series of inequality constraints on the optimal objective values of linear programs, which converts (17) to a bilevel program. Second, this bilevel program is reformulated as an MIQP. The details are omitted.

An advantage of this reformulation is that the number of binary variables is proportional to the total number of generators in the zonotopes defining $\mathscr{S}_Q(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$ in (16), which can be kept small by simple order reduction techniques [16]. This leads to a restriction of (17) and hence an optimal input with larger objective value than necessary for separation. Case studies in [9] show that large efficiency gains are achievable with minor added conservatism.

Given an integer $\overline{N}$, Q can either be computed or determined to be larger than $\overline{N}$ by simply repeating the above optimization procedure with N increasing from 1 until either a feasible program is generated or $\overline{N}$ is exceeded.

V. Moving Horizon Input Design

This section presents two closed-loop input design approaches for active fault diagnosis. The *direct method* applies the open-loop input design method described in the previous section within a moving horizon framework. This method is illustrative and clearly demonstrates the advantages of a closed-loop approach. However, the cost of the online computations may make this method impractical for many applications of interest. To address this, the *explicit method* uses an explicit control law, computed off-line, that approximates the action of the direct method.

[1]Theorem 1 shows that $\mathscr{S}_Q(\{(i,X_0^{[i]})\}_{i\in\mathscr{M}})$ need not be compact, so we should speak of ε-solutions with $J_Q(\tilde{\mathbf{u}})$ within $\varepsilon > 0$ of the infimum.

A. Set-Valued Observers

For the purposes of closed-loop input design, we are interested in the set of all states at time k that are consistent with a given model i, a given initial set X_0 (assumed to be at $k = 0$ w.l.o.g.), a known input sequence $\tilde{\mathbf{u}} = (\mathbf{u}_0, \ldots, \mathbf{u}_{N-1})$, and a measured output sequence $\tilde{\mathbf{y}} = (\mathbf{y}_0, \ldots, \mathbf{y}_N)$. Precisely, this set is

$$H_k^{[i]}(\tilde{\mathbf{u}}, \tilde{\mathbf{y}}, X_0) \equiv \{\phi_k^{[i]}(\tilde{\mathbf{u}}, \mathbf{x}_0, \tilde{\mathbf{w}}) : (\mathbf{x}_0, \tilde{\mathbf{w}}) \in X_0 \times \tilde{W}, \qquad (18)$$
$$\phi_\ell^{[i]}(\tilde{\mathbf{u}}_{0:\ell-1}, \mathbf{x}_0, \tilde{\mathbf{w}}_{0:\ell-1}) \in X_{\mathrm{m}}^{[i]}(\mathbf{y}_\ell),\ 0 \leq \ell \leq k\},$$

where $X_{\mathrm{m}}^{[i]}(\mathbf{y}_\ell) \equiv \{\mathbf{x}_\ell : \mathbf{y}_\ell - \mathbf{C}^{[i]}\mathbf{x}_\ell - \mathbf{s}^{[i]} \in \mathbf{D}_v^{[i]} V\}$. Since this set is difficult to compute exactly, it is enclosed using a set-valued observer [18].

Definition 2: A *set-valued observer* for model i is a family of set-valued mappings $\{\mathscr{O}_k^{[i]}\}$, $k \in \mathbb{N}$, satisfying $\mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}, \tilde{\mathbf{y}}, X_0) \supset H_k^{[i]}(\tilde{\mathbf{u}}, \tilde{\mathbf{y}}, X_0)$ for all $(\tilde{\mathbf{u}}, \tilde{\mathbf{y}}) \in \mathbb{R}^{kn_u} \times \mathbb{R}^{(k+1)n_y}$, $X_0 \subset \mathbb{R}^{n_x}$, and $k \in \mathbb{N}$. It is *exact* if these sets are equal.

This article uses an observer based on zonotope computations proposed in [19]. For $X_0 \subset \mathbb{R}^{n_x}$ and sequences $\tilde{\mathbf{u}} = (\mathbf{u}_0, \mathbf{u}_1, \ldots)$ and $\tilde{\mathbf{y}} = (\mathbf{y}_0, \mathbf{y}_1, \ldots)$, this observer satisfies the recursion relation

$$\mathscr{O}_0^{[i]}(\mathbf{y}_0, X_0) \supset \left[X_{\mathrm{m}}^{[i]}(\mathbf{y}_0) \cap X_0\right], \qquad (19)$$

$$\mathscr{O}_{k+1}^{[i]}(\tilde{\mathbf{u}}_{0:k}, \tilde{\mathbf{y}}_{0:k+1}, X_0) \qquad (20)$$
$$\supset \left[X_{\mathrm{m}}^{[i]}(\mathbf{y}_{k+1}) \cap X_1^{[i]}(\mathbf{u}_k, \mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}_{0:k-1}, \tilde{\mathbf{y}}_{0:k}, X_0))\right].$$

Note that $X_1^{[i]}(\mathbf{u}_k, \mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}_{0:k-1}, \tilde{\mathbf{y}}_{0:k}, X_0))$ is the set of states reachable from $\mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}_{0:k-1}, \tilde{\mathbf{y}}_{0:k}, X_0)$ in one step with input $\mathbf{u}_k$, and is a zonotope provided that $\mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}_{0:k-1}, \tilde{\mathbf{y}}_{0:k}, X_0)$ is a zonotope. However, the intersections above are not generally zonotopes and must be conservatively enclosed, so this observer is not exact.

B. The Direct Method

The basic idea of the direct method is as follows. First, an optimal input separating $\{(i,X_0^{[i]})\}_{i\in\mathscr{M}}$ in N steps is computed and the first element of this sequence is injected. At each sampling time k, enclosures of the current states consistent with each model are provided by set-valued observers, and these enclosures are used to compute a new optimal separating input. The first element of this new input is injected and the process is repeated.

A basic question is whether or not an N-step separating input can be refined at $M < N$, using a new $(N-M)$-step separating input, while maintaining the guarantee of diagnosability at time N. The following theorem answers this in the affirmative.

Theorem 2: Choose $0 < M < N$, $\tilde{\mathbf{u}} = (\mathbf{u}_0, \ldots, \mathbf{u}_{N-1})$, and $\tilde{\mathbf{y}} = (\mathbf{y}_0, \ldots, \mathbf{y}_M)$. Let $\{\mathscr{O}_k^{[i]}\}$ be a set-valued observer for model i and let $\overline{X}_M^{[i]} \equiv \mathscr{O}_M^{[i]}(\tilde{\mathbf{u}}_{0:M-1}, \tilde{\mathbf{y}}_{0:M}, X_0^{[i]})$, $\forall i \in \mathscr{M}$. Then

$$\tilde{\mathbf{u}}_{M:N-1} \in \mathscr{S}_{N-M}(\{(i, \overline{X}_M^{[i]})\}_{i\in\mathscr{M}}) \implies \tilde{Y}_{0:N}^{[i]}(\tilde{\mathbf{u}}, X_0^{[i]}) \cap \tilde{Y}_{0:N}^{[j]}(\tilde{\mathbf{u}}, X_0^{[j]}) = \emptyset,\ \forall i, j \in \mathscr{M},\ i \neq j. \tag{21}$$

The converse of (21) holds if $\{\mathscr{O}_k^{[i]}\}$ is exact.

Proof: See Appendix VII-A. ∎

Although, the basic moving horizon procedure outlined above does guarantee diagnosis at time N, it does not satisfy recursive feasibility unless $\{\mathscr{O}_k^{[i]}\}$ is exact:

Corollary 1: Choose $0 \leq k < k+1 < N$, $\tilde{\mathbf{u}} = (\mathbf{u}_0, \ldots, \mathbf{u}_{N-1})$, and $\tilde{\mathbf{y}} = (\mathbf{y}_0, \ldots, \mathbf{y}_{k+1})$. For all $i \in \mathscr{M}$, let $\{\mathscr{O}_k^{[i]}\}$ be a set-valued observer and define $\overline{X}_k^{[i]} \equiv \mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}_{0:k-1}, \tilde{\mathbf{y}}_{0:k}, X_0^{[i]})$ and $\overline{X}_{k+1}^{[i]} \equiv \mathscr{O}_{k+1}^{[i]}(\tilde{\mathbf{u}}_{0:k}, \tilde{\mathbf{y}}_{0:k+1}, X_0^{[i]})$. If $\tilde{\mathbf{u}}_{k:N-1} \in \mathscr{S}_{N-k}(\{(i, \overline{X}_k^{[i]})\}_{i\in\mathscr{M}})$ and $\{\mathscr{O}_k^{[i]}\}$ is exact, then $\tilde{\mathbf{u}}_{k+1:N-1} \in \mathscr{S}_{N-k-1}(\{(i, \overline{X}_{k+1}^{[i]})\}_{i\in\mathscr{M}})$.

Proof: See Appendix VII-A. ∎

As mentioned in §V-A, we use an inexact recursive observer for computational efficiency. Thus, we propose a modified receding horizon method described in detail below. The difficulty raised by Corollary 1 is circumvented by Step 7.

In the algorithm below, $\tilde{\mathbf{y}} = (\mathbf{y}_0, \mathbf{y}_1, \ldots)$ denotes the output measurements. Absolute times are denoted by k, T and L, whereas horizons are denoted by N. T_k is the time at which diagnosis will be possible given the input computed at time k, while N_k is the remaining time needed for diagnosis, i.e., $T_k = k + N_k$. $L_k \leq k$ denotes the last time at which the input sequence was *updated* (see Step 7). $\tilde{\mathbf{u}}^*_{(k:T_k-1|k)}$ and $\tilde{\mathbf{u}}^{OL}_{(k:T_k-1|k)}$ denote open-loop inputs computed at time k, and $\tilde{\mathbf{u}}^{CL} = (\mathbf{u}_0^{CL}, \mathbf{u}_1^{CL}, \ldots)$ denotes the closed-loop input.

Algorithm 1 (Moving Horizon Method)

1) Input: $\mathscr{M}$, W, V, $X_0^{[i]}$.
2) (Initialization) $k = 0$, $T_{-1} = \infty$, $L_{-1} = 0$, $N_{-1} = \infty$, $\mathscr{M}_{-1} = \mathscr{M}$, $\tilde{\mathbf{u}}^{CL}_{0:-1} = \emptyset$.
3) (Measurement) Measure the output $\mathbf{y}_k$.
4) (Termination) If $k = T_{k-1}$, assign $T_k := k$, $L_k := L_{k-1}$, and $\mathscr{M}_k := \mathscr{M}_{k-1}$ and go to Step 10.
5) (Observer Update) Compute $\overline{X}_k^{[i]}$ and $\mathscr{M}_k$ as follows:
   a) $\overline{X}_k^{[i]} := \mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}^{CL}_{0:k-1}, \tilde{\mathbf{y}}_{0:k}, X_0^{[i]})$, $\forall i \in \mathscr{M}_{k-1}$.
   b) Assign $\mathscr{M}_k := \mathscr{M}_{k-1}$. Choose $q \in \{0, \ldots, k\}$ and remove i from $\mathscr{M}_k$ if $\exists j \in \{q, \ldots, k\}$ with
   $$X_{k-j}^{[i]}(\tilde{\mathbf{u}}^{CL}_{j:k-1}, \overline{X}_j^{[i]}) \cap X_{\mathrm{m}}^{[i]}(\mathbf{y}_k) = \emptyset. \tag{22}$$
6) (Input Computation) Compute the minimum separation horizon N_k^* for $\{(i, \overline{X}_k^{[i]})\}_{i\in\mathscr{M}_k}$ or determine that $N_k^* > N_{k-1} - 1$. If $N_k^* \leq N_{k-1} - 1$, compute an optimal input $\tilde{\mathbf{u}}^*_{(k:k+N_k^*-1|k)} \in \mathscr{S}_{N_k^*}(\{(i, \overline{X}_k^{[i]})\}_{i\in\mathscr{M}_k})$.
7) (Input Update) Assign $N_k := \min(N_k^*, N_k - 1)$ and $T_k := k + N_k$. If $k = 0$, or $T_k < T_{k-1}$, or [$T_k = T_{k-1}$ and $J_{N_k}(\tilde{\mathbf{u}}^*_{(k:T_k-1|k)}) < J_{N_k}(\tilde{\mathbf{u}}^{OL}_{(k:T_k-1|k-1)})$], do (23). Otherwise, do (24).
   $$\tilde{\mathbf{u}}^{OL}_{(k:T_k-1|k)} := \tilde{\mathbf{u}}^*_{(k:T_k-1|k)}, \quad L_k := k, \tag{23}$$
   $$\tilde{\mathbf{u}}^{OL}_{(k:T_k-1|k)} := \tilde{\mathbf{u}}^{OL}_{(k:T_k-1|k-1)}, \quad L_k := L_{k-1}. \tag{24}$$
8) (Input Injection) Inject the input $\mathbf{u}_k^{CL} := \mathbf{u}^{OL}_{(k|k)}$.
9) (Increment Time) Assign $k := k+1$ and go to Step 3.
10) (Diagnosis) $\mathscr{M}_{\text{out}} := \mathscr{M}_k$. Remove i from $\mathscr{M}_{\text{out}}$ if
    $$\tilde{\mathbf{y}}_{L_k+1:T_k} \notin \tilde{Y}_{1:T_k-L_k}^{[i]}(\tilde{\mathbf{u}}^{CL}_{L_k:T_k-1}, \overline{X}_{L_k}^{[i]}). \tag{25}$$
11) Output: $\mathscr{M}_{\text{out}}$.

Theorem 3: If $N_0^* < \infty$, then Algorithm 1 terminates in $\beta \leq N_0^*$ steps with $|\mathscr{M}_{\text{out}}| \leq 1$ and $i \in \mathscr{M}_{\text{out}}$ if and only if

$$\tilde{\mathbf{y}}_{0:\beta} \in \tilde{Y}_{0:\beta}^{[i]}(\tilde{\mathbf{u}}^{CL}_{0:\beta-1}, X_0^{[i]}) \tag{26}$$

Proof: See the Appendix. ∎

Remark 1: Step 5b is used to rule out models so that better inputs can potentially be obtained in Step 6. However, Step 5b is not guaranteed to eliminate all inconsistent models. Determining exactly which models can be eliminated generally requires checking $H_k^{[i]}(\tilde{\mathbf{u}}^{CL}_{0:k-1}, \tilde{\mathbf{y}}_{0:k}, X_0^{[i]}) = \emptyset$, which is very difficult. At termination, (25) suffices to provide a diagnosis by design (Theorem 3).

C. The Explicit Method

The online computational cost of the direct method is dominated by the computation of the new separating input in Step 6. In this section, it is observed that this computation can be moved off-line by using a weaker set-valued observer, leading to an explicit control law that approximates the action of the direct method. The following assumption is required.

Assumption 1: A set $\overline{Y} \subset \mathbb{R}^{n_y}$ is known that contains the outputs of all models $i \in \mathscr{M}$, for all $k \in \mathbb{N}$. Moreover, $\mathbf{C}^{[i]}$ is invertible for every $i \in \mathscr{M}$.

Let $\{P_\sigma\}_{\sigma\in S}$ be a partition of $\overline{Y}$ into intervals $P_\sigma = \hat{\mathbf{y}}_\sigma + Y_{\text{grid}}$, where Y_{grid} is a zero-centered interval. Define

$$X_\sigma^{[i]} \equiv \{\mathbf{x} : \hat{\mathbf{y}}_\sigma - \mathbf{C}^{[i]}\mathbf{x} - \mathbf{s}^{[i]} \in \mathbf{D}_v^{[i]} V + Y_{\text{grid}}\}. \tag{27}$$

Note that $X_{\mathrm{m}}^{[i]}(\mathbf{y}) \subset X_\sigma^{[i]}$, $\forall \mathbf{y} \in P_\sigma$. Moreover, invertibility of $\mathbf{C}^{[i]}$ implies that each $X_\sigma^{[i]}$ is a zonotope. This motivates the definition of the simple zonotopic observer

$$\mathscr{O}_k^{[i]}(\tilde{\mathbf{u}}, \tilde{\mathbf{y}}, X_0) = X_\sigma^{[i]} \text{ if } \mathbf{y}_k \in P_\sigma. \tag{28}$$

Consider applying Algorithm 1 with $X_0^{[i]} = \mathbb{R}^{n_x}$, $\forall i \in \mathscr{M}$, and the set-valued observer defined by (28). With this observer, each of the sets $\overline{X}_k^{[i]}$ generated by Algorithm 1 is one of the sets $X_\sigma^{[i]}$ with $\sigma \in S$. Thus, Step 6 can be reduced to a simple look-up operation online by pre-computing an optimal input separating $\{(i, X_\sigma^{[i]})\}_{i\in\mathscr{M}}$ for every $\sigma \in S$. Moreover, all of the reachable sets required in Step 5b can be computed off-line and stored. To follow Algorithm 1 exactly, these off-line computations must be completed for every possible subset of models $\mathscr{M}_k \subset \mathscr{M}$. However, it is also possible to consider only $\mathscr{M}$ and omit the elimination of models in Step 5b.

In any case, 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 as compared to the direct method. On the other hand, it uses a much weaker observer. Thus, updated separating inputs will be accepted less frequently in Step 7, thereby diminishing the advantages of a closed-loop input design procedure.

VI. Numerical Example

Consider the second-order linear nominal system

$$\mathbf{A}^{[1]} = \begin{bmatrix} 0.84 & 0.28 \\ -0.28 & 0.98 \end{bmatrix}, \quad \mathbf{B}^{[1]} = \begin{bmatrix} -0.386 & 0.199 \\ -0.199 & 0.386 \end{bmatrix}, \quad \mathbf{C}^{[1]} = \begin{bmatrix} 0.7 & 0 \\ 0 & 0.3 \end{bmatrix},$$
$$\mathbf{B}_w^{[1]} = \begin{bmatrix} 0.1215 & 0.0598 \\ 0.0598 & 0.1215 \end{bmatrix}, \quad \mathbf{D}_v^{[1]} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{r}^{[1]} = \mathbf{s}^{[1]} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

with four fault models, $i = 2, \ldots, 5$, defined by the modifications to the nominal model:

$$\mathbf{B}^{[2]} = \begin{bmatrix} -0.3861 & 0 \\ -0.1994 & 0 \end{bmatrix}, \quad \mathbf{B}^{[3]} = \begin{bmatrix} 0 & 0.1994 \\ 0 & 0.3861 \end{bmatrix},$$
$$\mathbf{A}^{[4]} = \begin{bmatrix} 0.84 & 0 \\ -0.28 & 0.98 \end{bmatrix}, \quad \mathbf{A}^{[5]} = \begin{bmatrix} 0.84 & 0.28 \\ 0 & 0.98 \end{bmatrix}.$$

In generator notation, $W = \{0.7\mathbf{I}, \mathbf{0}\}$ (e.g., $W = \{\mathbf{w} : \|\mathbf{w}\|_\infty \leq 0.7\}$), and $V = \{0.28\mathbf{I}, \mathbf{0}\}$. For the optimization (17), $\mathbf{R} = \mathbf{I}$ and $U = \{\mathbf{u} : \|\mathbf{u}\|_\infty \leq 2.5\}$. The number of generators in the zonotopes defining $\mathscr{S}_N$ was limited to 2 using the order reduction in method in [16].

The top row of Figure 1 shows the performance of the direct method relative to the open-loop input computed in the first iteration of Algorithm 1. In Step 5b of Algorithm 1, we always choose $q = 0$ and consider only $j = 0$. Although measurements are not used to update the input in the open-loop method, they are used to discard models by checking $X_{\mathrm{m}}^{[i]}(\mathbf{y}_k) \cap X_k^{[i]}(\tilde{\mathbf{u}}_{0:k-1}^{CL}, \overline{X}_0^{[i]}) = \emptyset$ as in Step 5b. In Figure 1, T_{DM}^{CL} and T_{DM}^{OL} denote the time required for diagnosis for the direct method and the corresponding open-loop method, respectively, and $\tilde{\mathbf{u}}_{DM}^{CL}$ and $\tilde{\mathbf{u}}_{DM}^{OL}$ denote the corresponding inputs. Ten-thousand random realizations of $\mathbf{y}_0 \in \{4\mathbf{I}, \mathbf{0}\}$ and $(\mathbf{w}_k, \mathbf{v}_k) \in W \times V$ were considered. As expected, the direct method provides a substantial reduction in the average input length and norm compared to the corresponding open-loop input.

The middle row of Figure 1 shows the performance of the explicit method with $Y_{\mathrm{grid}} = \{0.1\mathbf{I}, \mathbf{0}\}$ relative to the open-loop input computed in the first iteration of Algorithm 1. Note that this open-loop input differs from that used in the comparison for the direct method above because the sets $\overline{X}_0^{[i]}$ depend on the observer used at $k = 0$. Again, measurements are used in the open-loop method only to discard models by checking $X_{\mathrm{m}}^{[i]}(\mathbf{y}_k) \cap X_k^{[i]}(\tilde{\mathbf{u}}_{0:k-1}^{CL}, \overline{X}_0^{[i]}) = \emptyset$. In Figure 1, T_{EM}^{CL} and T_{EM}^{OL} denote the time required for diagnosis for the explicit method and the corresponding open-loop method, respectively, and $\tilde{\mathbf{u}}_{EM}^{CL}$ and $\tilde{\mathbf{u}}_{EM}^{OL}$ denote the corresponding inputs. Figure 1 clearly shows the importance of using a closed-loop input, even when using the weaker explicit method. Although the observer (28) is quite basic, the input sequence is still refined periodically in Step 7. Moreover, by eliminating models in Step 5b, an explicit controller computed for a subset of $\mathscr{M}$ can be applied. Figure 2 shows the input length and norm required by the explicit feedback laws for $\mathscr{M} = \{1,2,3,4,5\}$ and $\mathscr{M}' = \{3,4\}$. From this figure it can be seen that eliminating models can lead to a significant reduction in input length and norm.

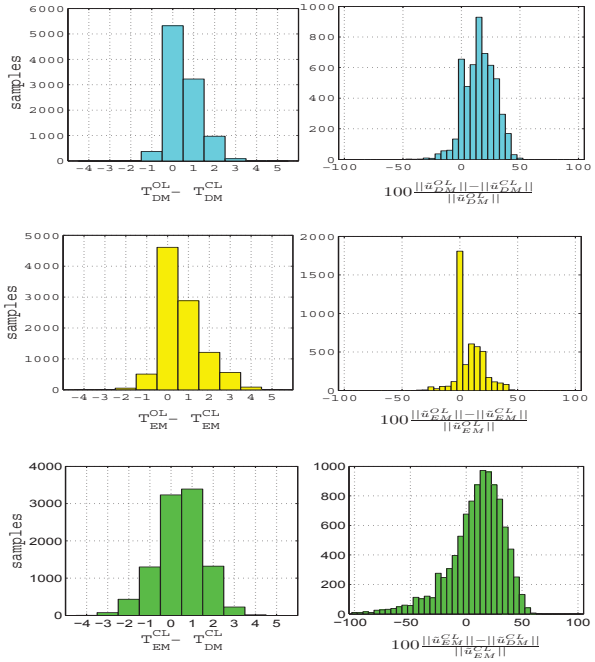

Fig. 1. Average reduction in input length (left) and norm (right) of the direct method compared to the corresponding open-loop input (top), the explicit method compared to the corresponding open-loop input (middle), and the direct method compared to the explicit method (bottom).

The bottom row of Figure 1 shows a comparison of the direct and explicit methods. The direct method provides inputs of significantly smaller length and norm, as expected. On the other hand, the time required to solve (17) online may be prohibitive for some applications. With $X_0^{[i]} = \{0.28\mathbf{I}, [2\ 1]^{\mathrm{T}}\}$, $\forall i \in \mathscr{M}$, the proposed solution method required $0.52s$ [2]. In contrast, the explicit method provides an approximate solution in ns [20].

VII. Conclusions

A moving horizon method was proposed for computing an active input that is guaranteed to lead to a fault diagnosis in a specified period of time. This method was shown to achieve a significant reduction in the length and norm of the required input compared to an open-loop method with the same guarantee. Furthermore, it was shown that an approximate explicit solution to the feedback problem can be synthesized off-line to reduce the online computational burden. Although this results in a suboptimal input sequence, the method is still superior to the open-loop method.

[2]Desktop PC (Intel i7, 3.4GHz, 8 GB RAM) running Ubuntu 11.10 and using a single core; Optimization using CPLEX 12.2 [17].

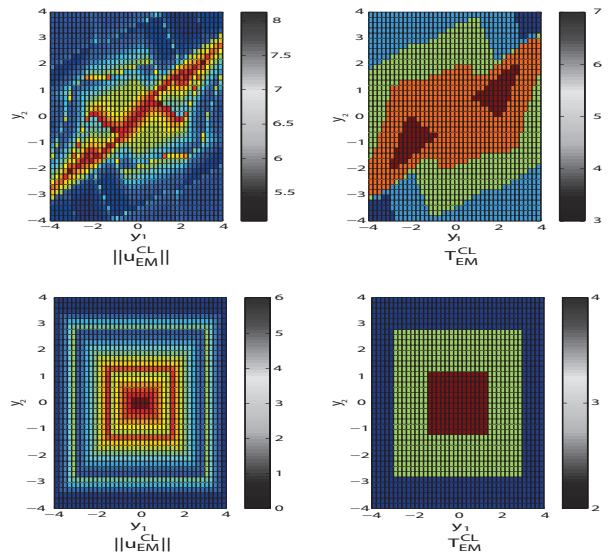

Fig. 2. Norm (left) and length (right) of the open-loop separating inputs prescribed by the explicit method as a function of the measurement when $\mathscr{M} = \{1,\ldots,5\}$ (top) and $\mathscr{M} = \{3,4\}$ (bottom).

APPENDIX

A. Proof of Theorem 2 and Corollary 1

Lemma 1: Choose $0 < M < N$, $\tilde{\mathbf{u}} = (\mathbf{u}_0,\ldots,\mathbf{u}_{N-1})$, and $\tilde{\mathbf{y}} = (\mathbf{y}_0,\ldots,\mathbf{y}_N)$. Let $\{\mathscr{O}_k^{[i]}\}$ be a set-valued observer for model i and let $\overline{X}_M^{[i]} \equiv \mathscr{O}_M^{[i]}(\tilde{\mathbf{u}}_{0:M-1}, \tilde{\mathbf{y}}_{0:M}, X_0^{[i]})$. Then

$$\tilde{\mathbf{y}} \in \tilde{Y}_{0:N}^{[i]}(\tilde{\mathbf{u}}, X_0^{[i]}) \implies \tilde{\mathbf{y}}_{M+1:N} \in \tilde{Y}_{1:N-M}^{[i]}(\tilde{\mathbf{u}}_{M:N-1}, \overline{X}_M^{[i]}). \quad (29)$$

The converse of (29) holds if $\{\mathscr{O}_k^{[i]}\}$ is exact.

Proof: Since $\tilde{\mathbf{y}} \in \tilde{Y}_{0:N}^{[i]}(\tilde{\mathbf{u}}, X_0^{[i]})$, there exists $\mathbf{x}_0 \in X_0^{[i]}$, $\tilde{\mathbf{w}} \in \tilde{W}$, and $\tilde{\mathbf{v}} \in \tilde{V}$ such that $\tilde{\mathbf{y}} = \tilde{\psi}_{0:N}^{[i]}(\tilde{\mathbf{u}}, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}})$. Defining $\mathbf{x}_M^* \equiv \phi_M^{[i]}(\tilde{\mathbf{u}}_{0:M-1}, \mathbf{x}_0, \tilde{\mathbf{w}}_{0:M-1})$, this implies $\mathbf{x}_M^* \in H_M^{[i]}(\tilde{\mathbf{u}}_{0:M-1}, \tilde{\mathbf{y}}_{0:M}, X_0^{[i]}) \subset \overline{X}_M^{[i]}$. By definition, this implies that $\tilde{\mathbf{y}}_{M+1:N}^* \equiv \tilde{\psi}_{1:N-M}^{[i]}(\tilde{\mathbf{u}}_{M:N-1}, \mathbf{x}_M^*, \tilde{\mathbf{w}}_{M:N-1}, \tilde{\mathbf{v}}_{M+1:N})$ is in $\tilde{Y}_{1:N-M}^{[i]}(\tilde{\mathbf{u}}_{M:N-1}, \overline{X}_M^{[i]})$. But the definition of $\mathbf{x}_M^*$ ensures that $\tilde{\mathbf{y}}_{M+1:N}^* = \tilde{\psi}_{M+1:N}^{[i]}(\tilde{\mathbf{u}}, \mathbf{x}_0, \tilde{\mathbf{w}}, \tilde{\mathbf{v}}_{M+1:N}) = \tilde{\mathbf{y}}_{M+1:N}$, so (29) follows. The converse follows by similar arguments. ■

To prove Theorem 2, choose any $\tilde{\mathbf{y}}_{M+1:N}$ and define $\tilde{\mathbf{y}} = (\tilde{\mathbf{y}}_{0:M}, \tilde{\mathbf{y}}_{M+1:N})$. The hypothesis in (21) implies that the right-hand side of (29) fails for all but possibly one $i \in \mathscr{M}$. By the contrapositive of Lemma 1, the same holds for the left-hand side of (29), which proves (21). The converse follows similarly from the converse of Lemma 1.

Under the hypothesis of Corollary 1, Theorem 2 with $M = k$ implies that the right-hand side of (21) holds. The converse of Theorem 2 with $M = k+1$ proves the Corollary.

B. Proof of Theorem 3

By Step 7, $T_0 = N_0^* < \infty$. Let α be the largest integer less than T_0 for which the assignment (23) is made with $\alpha = k$, and let $\beta \equiv T_\alpha$. It follows $\beta \le T_0$, $T_k = \beta$, and $L_k = \alpha$ for all $\alpha \le k \le T_0$. Moreover, $\tilde{\mathbf{u}}_{\alpha:\beta-1}^{CL} = \tilde{\mathbf{u}}_{(\alpha:\beta-1|\alpha)}^*$. When $k = \beta$, Step 4 dictates that Steps 10-11 are visited, terminating the algorithm. Upon termination, (25) reads $\tilde{\mathbf{y}}_{\alpha+1:\beta} \notin \tilde{Y}_{1:\beta-\alpha}^{[i]}(\tilde{\mathbf{u}}_{\alpha:\beta-1}^{CL}, \overline{X}_\alpha^{[i]})$. Since $\tilde{\mathbf{u}}_{\alpha:\beta-1}^{CL}$ separates $\{(i, \overline{X}_\alpha^{[i]})\}_{i\in\mathscr{M}_\alpha}$ in $(\beta-\alpha)$ steps, this holds for all but one $i \in \mathscr{M}_\alpha$. By Step 5b, $\mathscr{M}_{\text{out}} \subset \mathscr{M}_\beta \subset \mathscr{M}_\alpha$, so that $|\mathscr{M}_{\text{out}}| \le 1$.

If $i \in \mathscr{M}_{\text{out}}$, then Lemma 1 and the failure of (25) imply (26). It remains to show that (26) fails for $i \notin \mathscr{M}_{\text{out}}$. By Theorem 2 and the definition of $\tilde{\mathbf{u}}_{\alpha:\beta-1}^{CL}$, (26) fails for all but one $i^* \in \mathscr{M}_\alpha$, and this i^* must satisfy (25), so that $i^* \in \mathscr{M}_{\text{out}}$. For any $i \notin \mathscr{M}_\alpha$, Step 5b implies that there exist $j \le k < \alpha$ such that (22) holds. Thus, $\tilde{\mathbf{y}}_{j+1:k} \notin \tilde{Y}_{1:k-j}^{[i]}(\tilde{\mathbf{u}}_{j:k-1}^{CL}, \overline{X}_j^{[i]})$. The contrapositive of Lemma 1 shows that (26) fails.

REFERENCES

[1] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri, "A review of process fault detection and diagnosis: Part I: Quantitative model-based methods," *Comp. & Chem. Eng.*, vol. 27, no. 3, pp. 293–311, 2003.

[2] A. Zolghadri, "Advanced model-based FDIR techniques for aerospace systems: Today challenges and opportunities," *Progress in Aerospace Sciences*, vol. 53, no. 3, pp. 18–29, 2010.

[3] L. H. Chiang, E. L. Russell, and R. D. Braatz, *Fault Detection and Diagnosis in Industrial Systems*. Springer-Verlag, London, 2001.

[4] I. Hwang, S. Kim, Y. Kim, and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE Trans. Contr. Syst. Technol.*, vol. 18, no. 3, pp. 636–653, 2010.

[5] X. J. Zhang, *Auxiliary Signal Design in Fault Detection and Diagnosis*. Springer-Verlag, Berlin, 1989.

[6] M. Simandl, I. Puncochar, and J. Kralovec, "Rolling horizon for active fault detection," in *IEEE CDC Proc.*, 2005, pp. 3789–3794.

[7] R. Nikoukhah, "Guaranteed active failure detection and isolation for linear dynamical systems," *Automatica*, vol. 34, no. 11, pp. 1345–1358, 1998.

[8] S. L. Campbell and R. Nikoukhah, *Auxiliary Signal Design for Failure Detection*. Princeton University Press, 2004.

[9] J. K. Scott, R. Findeisen, R. D. Braatz, and D. M. Raimondo, "Design of active inputs for set-based fault diagnosis," in *Proc. of the American Control Conference*, 2013.

[10] I. Andjelkovic, K. Sweetingham, and S. Campbell, "Active fault detection in nonlinear systems using auxiliary signals," in *Proc. of the American Control Conference*, 2008, pp. 2142–2147.

[11] R. Nikoukhah, S. L. Campbell, and F. Delebecque, "Detection signal design for failure detection: A robust approach," *Int. J. Adaptive Control and Signal Processing*, vol. 14, no. 7, pp. 701–724, 2000.

[12] R. Nikoukhah and S. L. Campbell, "Auxiliary signal design for active failure detection in uncertain linear systems with a priori information," *Automatica*, vol. 42, no. 2, pp. 219–228, 2006.

[13] A. E. Ashari, R. Nikoukhah, and S. L. Campbell, "Asymptotic behavior and solution approximation of active robust fault detection for closed-loop systems," in *IEEE CDC Proc.*, 2009, pp. 1026–1031.

[14] A. Ashari, R. Nikoukhah, and S. L. Campbell, "Effects of feedback on active fault detection," *Automatica*, vol. 48, pp. 866–872, 2012.

[15] L. J. Guibas, A. Nguyen, and L. Zhang, "Zonotopes as bounding volumes," in *14th ACM-SIAM Symp. on Discrete Algorithms*, 2003.

[16] M. Althoff, O. Stursberg, and M. Buss, "Computing reachable sets of hybrid systems using a combination of zonotopes and polytopes," *Nonlinear Analysis-Hybrid Systems*, vol. 4, no. 2, pp. 233–249, 2010.

[17] *IBM ILOG CPLEX V12.2 User's Manual for CPLEX*, 2012.

[18] J. S. Shamma, "Set-valued observers and optimal disturbance rejection," *IEEE Trans. Automat. Contr.*, vol. 44, pp. 253–264, 1999.

[19] J. Bravo, T. Alamo, and E. Camacho, "Bounded error identification of systems with time-varying parameters," *IEEE Trans. Automat. Contr.*, vol. 51, pp. 1144–1150, 2006.

[20] S. Summers, C. Jones, J. Lygeros, and M. Morari, "A multiresolution approximation method for fast explicit model predictive control," *IEEE Trans. Automat. Contr.*, vol. 56, no. 11, pp. 2530–2541, 2011.