Synthesis of Partially Observed Jump-Diffusion Systems via Control Barrier Functions

Niloofar Jahanshahi†, Pushpak Jagtap†, and Majid Zamani

***Abstract*—In this paper, we study formal synthesis of control policies for partially observed jump-diffusion systems against complex logic specifications. Given a state estimator, we utilize a discretization-free approach for formal synthesis of control policies by using a notation of control barrier functions without requiring any knowledge of the estimation accuracy. Our goal is to synthesize an offline control policy providing (potentially maximizing) a lower bound on the probability that the trajectories of the partially observed jump-diffusion system satisfy some complex specifications expressed by deterministic finite automata. Finally, we illustrate the effectiveness of the proposed results by synthesizing a policy for a jet engine example.**

***Index Terms*—Stochastic control systems, Control barrier functions, Controller synthesis, Output feedback control.**

I. Introduction

RECENT years have witnessed a growing interest in formal synthesis of controllers for complex systems against complex logic specifications [1]. These specifications are usually expressed using temporal logic formulae or as (in)finite strings over finite automata. Several approaches based on finite abstraction have been widely used to solve such synthesis problems. Existing techniques include policy synthesis enforcing linear temporal logic specifications for non-stochastic systems [2], [3] and for stochastic ones [4], [5], [6]. When dealing with large systems, these approaches suffer severely from the curse of dimensionality (*i.e.*, computational complexity grows exponentially with the dimension of the state set). In order to overcome the large computational burden, a discretization-free approach, based on control barrier functions has shown potential to solve the formal synthesis problems (See [7], [8], [9], [10] and references therein). The aforementioned works assume the availability of complete state information. However, in many real applications we do not have access to complete state information. Motivated by this limitation, the recent result in [11] provides the synthesis of controllers enforcing invariance properties for stochastic control systems with incomplete information by assuming a prior knowledge of the control barrier functions. In our recent result [12], we consider the problem of synthesizing controllers for partially observed stochastic control systems. In particular, we search for a control barrier function that provides a controller along with a lower bound on the probability that the system satisfies invariance specifications over a finite-time horizon. Similar to [11], this work also assumes the existence of an estimator with a given probabilistic accuracy. Then we provide the overall probability threshold using the probability bound on the estimator accuracy and that of the trajectories of the estimator satisfying the invariance specifications, obtained via control barrier functions.

The contributions of this paper in comparison with those of [11], [12] are twofold. First, we provide an offline controller synthesis approach enforcing complex logic specifications expressed by (non)deterministic finite automata for partially observed jump-diffusion systems. As a special case, those properties include invariance ones. Second, we provide an approach for computing lower bound on the probability that the system satisfies given specifications over a finite-time horizon *without* requiring any knowledge of the estimator's accuracy. Finally, we demonstrate the effectiveness of the proposed results on a nonlinear jet engine example.

II. Preliminaries and Problem Definition

Notations: We denote the set of natural, real, and non-negative real numbers by $\mathbb{N}$, $\mathbb{R}$, and $\mathbb{R}_0^+$, respectively. We use $\mathbb{R}^n$ to denote the n-dimensional Euclidean space and $\mathbb{R}^{n\times r}$ to denote the space of real matrices with n rows and r columns. We denote by $e_i \in \mathbb{R}^n$ the vector whose all elements are zero, except the i^{th} element, which is one. Given a matrix $A \in \mathbb{R}^{n\times n}$, $\text{Tr}(A)$ represents trace of A which is the sum of all diagonal elements of A. The zero matrix in $\mathbb{R}^{n\times m}$ is denoted by $0_{n\times m}$. Given sets X and Y, we donate $f : X \to Y$ an ordinary map from X to Y and the notation $|X|$ denotes the cardinality of set X.

A. Partially Observed Jump-Diffusion Systems

Let the triplet $(\Omega, \mathcal{F}, \mathbb{P})$ denote a probability space with a sample space Ω, filtration $\mathcal{F}$, and the probability measure $\mathbb{P}$. The filtration $\mathbb{F} = (\mathcal{F}_s)_{s\geq 0}$ satisfies the usual conditions of right continuity and completeness [13]. Let $(W_{ks})_{s\geq 0}$ be $\bar{r}_k$-dimensional $\mathbb{F}$-Brownian motions, $k = 1, 2$. Let $(P_{ks})_{s\geq 0}$ be a $\bar{q}_k$-dimensional $\mathbb{F}$-Poisson processes, with $k = 1, 2$. We assume that the Poisson processes and Brownian motions are independent of each other. The Poisson process $P_{ks} := [P_{ks}^1; \cdots ; P_{ks}^{\bar{q}_k}]$ models $\bar{q}_k$ kinds of events, $k = 1, 2$, whose occurrences are assumed to be independent of each other. We

†The authors contributed equally to this work.

This work was supported in part by the H2020 ERC Starting Grant AutoCPS (grant agreement No. 804639), the German Research Foundation (DFG) through the grants ZA 873/1-1 and the Research Training Group 2428, and the TUM International Graduate School of Science and Engineering (IGSSE).

N. Jahanshahi is with the Computer Science Department, Ludwig Maximilian University of Munich, Germany. P. Jagtap is with the Department of Electrical and Computer Engineering, Technical University of Munich, Germany. M. Zamani is with the Computer Science Department, University of Colorado Boulder, USA. M. Zamani is with the Computer Science Department, Ludwig Maximilian University of Munich, Germany. Emails: niloofar.jahanshahi@lmu.de, pushpak.jagtap@tum.de, majid.zamani@colorado.edu.

consider the partially observed jump-diffusion system (po-JDS), denoted by $\mathcal{S}$, which is described by the following stochastic differential equations (SDE)

$$\mathcal{S}: \begin{cases} \mathrm{d}\,\xi &= f(\xi,\upsilon)\,\mathrm{d}\,t + g_1(\xi)\,\mathrm{d}\,W_{1t} + r_1(\xi)\,\mathrm{d}\,P_{1t}, \\ \mathrm{d}\,y &= h(\xi)\,\mathrm{d}\,t + g_2(\xi)\,\mathrm{d}\,W_{2t} + r_2(\xi)\,\mathrm{d}\,P_{2t}, \end{cases} \tag{II.1}$$

where $\xi(t) \in X \subseteq \mathbb{R}^n$ is the value of solution process ξ of $\mathcal{S}$, $\upsilon(t) \in U \subseteq \mathbb{R}^m$ is the input vector, and $y(t) \in \mathbb{R}^p$ is the output vector representing the noisy partial observation at time $t \in \mathbb{R}_0^+$ $\mathbb{P}$-almost surely ($\mathbb{P}$-a.s.). Functions $f : X \times U \to \mathbb{R}^n$, $g_1 : X \to \mathbb{R}^{n\times \bar{r}_1}$, $g_2 : X \to \mathbb{R}^{p\times \bar{r}_2}$, $r_1 : X \to \mathbb{R}^{n\times \bar{q}_1}$, $r_2 : X \to \mathbb{R}^{p\times \bar{q}_2}$, and $h : X \to \mathbb{R}^p$ are assumed to be Lipschitz continuous to ensure existence and uniqueness of the solution of $\mathcal{S}$ [13]. Throughout the paper, we use the notation $\xi_{a\upsilon}(t)$ to denote the value of the solution process of $\mathcal{S}$ at time $t \in \mathbb{R}_0^+$ under the input signal υ starting from the initial state $\xi_{a\upsilon}(0) = a$ $\mathbb{P}$-a.s., in which a is a random variable that is measurable in $\mathcal{F}_0$. Here, we assume that the Poisson processes P^i_{ks} for any $i \in \{1,\dots,\bar{q}_k\}$, $k = 1, 2$, have the rates of λ_{ki}. In order to provide the results in this paper, we raise the following assumption on the existence of the estimator that estimates the state of the po-JDS (II.1).

Assumption 2.1: The states of the po-JDS $\mathcal{S}$ in (II.1) can be estimated by a proper estimator $\hat{\mathcal{S}}$ represented in the form of an SDE as:

$$\hat{\mathcal{S}}: \mathrm{d}\,\hat{\xi} = f(\hat{\xi},\upsilon)\,\mathrm{d}\,t + K\big(\mathrm{d}\,y - h(\hat{\xi})\,\mathrm{d}\,t\big), \tag{II.2}$$

where $K \in \mathbb{R}^{n\times p}$ is the estimator gain.

There are plenty of results in the literature on the computation of estimator gain K for various classes of stochastic systems; see the results in [14], [11], [15], and [16]. We define the augmented process $[\xi,\hat{\xi}]^T$, where ξ and $\hat{\xi}$ are the solution processes of $\mathcal{S}$ and $\hat{\mathcal{S}}$, respectively. The corresponding augmented jump-diffusion system $\tilde{\mathcal{S}}$ can be defined as:

$$\begin{aligned} \begin{bmatrix}\mathrm{d}\,\xi\\ \mathrm{d}\,\hat{\xi}\end{bmatrix} &= \Big(\begin{bmatrix} f(\xi,\upsilon)\\ f(\hat{\xi},\upsilon)\end{bmatrix} + \begin{bmatrix} 0_{n\times p} & 0_{n\times p}\\ K & -K\end{bmatrix}\begin{bmatrix} h(\xi)\\ h(\hat{\xi})\end{bmatrix}\Big)\,\mathrm{d}\,t \\ &+ \begin{bmatrix} g_1(\xi) & 0_{n\times \bar{r}_2}\\ 0_{n\times \bar{r}_1} & Kg_2(\xi)\end{bmatrix}\begin{bmatrix}\mathrm{d}\,W_{1t}\\ \mathrm{d}\,W_{2t}\end{bmatrix} + \begin{bmatrix} r_1(\xi)\\ 0_{n\times \bar{q}_1}\end{bmatrix}\mathrm{d}\,P_{1t} + \begin{bmatrix} 0_{n\times \bar{q}_2}\\ Kr_2(\xi)\end{bmatrix}\mathrm{d}\,P_{2t}. \end{aligned} \tag{II.3}$$

For later use, we provide the definition of the infinitesimal generator (denoted by operator $\mathcal{D}$) for $\tilde{\mathcal{S}}$ using Ito's differentiation [13]. Let $B : X \times X \to \mathbb{R}$ be a twice differentiable function. The infinitesimal generator of B associated with the system $\tilde{\mathcal{S}}$ for all $(x,\hat{x}) \in X \times X$ and for all $u \in U$ is given by

$$\begin{aligned} \mathcal{D}B(x,\hat{x},u) &= \begin{bmatrix}\partial_x B & \partial_{\hat{x}} B\end{bmatrix}\Big(\begin{bmatrix} f(x,u)\\ f(\hat{x},u)\end{bmatrix} + \begin{bmatrix} 0_{n\times p} & 0_{n\times p}\\ K & -K\end{bmatrix}\begin{bmatrix} h(x)\\ h(\hat{x})\end{bmatrix}\Big) \\ &+ \frac{1}{2}\mathrm{Tr}\Big(\begin{bmatrix} g_1(x) & 0_{n\times \bar{r}_2}\\ 0_{n\times \bar{r}_1} & Kg_2(x)\end{bmatrix}\begin{bmatrix} g_1(x) & 0_{n\times \bar{r}_2}\\ 0_{n\times \bar{r}_1} & Kg_2(x)\end{bmatrix}^T\begin{bmatrix}\partial_{xx}B & \partial_{x\hat{x}}B\\ \partial_{\hat{x}x}B & \partial_{\hat{x}\hat{x}}B\end{bmatrix}\Big) \\ &+ \sum_{i=1}^{\bar{q}_1}\lambda_{1i}\big(B(x + r_1(x)\mathrm{e}_i,\hat{x}) - B(x,\hat{x})\big) \\ &+ \sum_{i=1}^{\bar{q}_2}\lambda_{2i}\big(B(x + Kr_2(x)\mathrm{e}_i,\hat{x}) - B(x,\hat{x})\big). \end{aligned} \tag{II.4}$$

The symbols ∂_x and $\partial_{x,\hat{x}}$ in (II.4) represent first and second-order partial derivatives with respect to x (1st argument) and $\hat{x}$ (2nd argument), respectively. Note that we dropped the arguments of $\partial_x B$, $\partial_{\hat{x}} B$, $\partial_{x,x} B$, $\partial_{x,\hat{x}} B$, $\partial_{\hat{x},x} B$, and $\partial_{\hat{x},\hat{x}} B$ in (II.4) for the sake of simplicity.

Given a po-JDS $\mathcal{S}$ in (II.1), we aim at synthesizing a control policy that guarantees a potentially tight lower bound on the probability that system $\mathcal{S}$ satisfies a complex specification over a finite time horizon. The class of specifications considered in this paper are provided in the next subsection.

Remark 2.2: The use of the augmented system $\tilde{\mathcal{S}}$ will allow us to provide the main result of the paper without any correctness requirement on the observer. In particualr, our augmented system formulation provides the user the flexibility to design any observer by means of any technique. The probabilistic distance between the values of state and their estimator is natively considered in our formulation and one does not need to quantify this distance a-priori which is needed in the results proposed in [12], [11].

B. Specifications

In this subsection, we consider the class of specifications expressed by nondeterministic finite automata (NFA) as defined below.

Definition 2.3: [17] A nondeterministic finite automaton (NFA) is a tuple $\mathcal{A} = (Q, Q_0, \Sigma, \delta, F)$, where Q is a finite set of states, $Q_0 \subseteq Q$ is a set of initial states, Σ is a finite set (a.k.a. alphabet), $\delta : Q \times \Sigma \to P(Q)$ is a transition function, where $P(Q)$ denotes the power set of Q, and $F \subseteq Q$ is a set of accepting (or final) states.

NFA $\mathcal{A}$ is called *deterministic* if the transition function is defined as $\delta : Q \times \Sigma \to Q$, and we refer to it as deterministic finite automata (DFA). Since every NFA can be converted to its equivalent DFA using the powerset construction [18], in the rest of the paper, we only deal with DFA. Moreover, it is well known that the complement of a DFA $\mathcal{A}$, denoted by $\mathcal{A}^c$, is again a DFA [19]. We use the notation $q \xrightarrow{\sigma} q'$ to denote transition relation $(q,\sigma,q') \in \delta$. A finite word $\sigma = (\sigma_0,\sigma_1,\dots,\sigma_{k-1}) \in \Sigma^k$ is accepted by DFA $\mathcal{A}$ if there exists a finite state run $\mathbf{q} = (q_0,q_1,\dots,q_k) \in Q^{k+1}$ such that $q_0 \in Q_0$, $q_i \xrightarrow{\sigma_i} q_{i+1}$ for all $0 \le i < k$ and $q_k \in F$. The accepted language of $\mathcal{A}$, denoted by $\mathcal{L}(\mathcal{A})$, is the set of all words accepted by $\mathcal{A}$.

In this work, we consider those specifications given by the accepting languages of DFA $\mathcal{A}$ defined over a set of atomic propositions Π, i.e., the alphabet $\Sigma = \Pi$. We should highlight that all linear temporal logic specifications defined over finite traces, referred to as LTL_F, are recognized by DFA [20].

C. Satisfaction of Specification by po-JDS

A given po-JDS $\mathcal{S}$ in (II.1) is connected to the specification given by the accepting language of a DFA $\mathcal{A}$ defined over a set of atomic propositions Π, with the help of a measurable labeling function $L : X \to \Pi$ as described in the next definition which is similar to [21, Definition 2].

Definition 2.4: For a po-JDS $\mathcal{S}$ as in (II.1) and the labeling function $L : X \to \Pi$, a finite sequence $\sigma(\xi_{a\upsilon}) = (\sigma_0,\sigma_1,\dots,\sigma_{k-1}) \in \Pi^k$, $k \in \mathbb{N}$, is a finite trace of the

solution process ξ_{av} over a finite time horizon $[0,T) \subset \mathbb{R}_0^+$ if there exists an associated time sequence $t_0, t_1, \ldots, t_{k-1}$ such that $t_0 = 0$, $t_k = T$, and for all $j \in \{0,1,\ldots,k-1\}$, $t_j \in \mathbb{R}_0^+$ following conditions hold

- $t_j < t_{j+1}$;
- $\xi_{av}(t_j) \in L^{-1}(\sigma_j)$;
- If $\sigma_j \neq \sigma_{j+1}$, then for some $t_j' \in [t_j, t_{j+1}]$, $\xi_{av}(t) \in L^{-1}(\sigma_j)$ for all $t \in (t_j, t_j')$; $\xi_{av}(t) \in L^{-1}(\sigma_{j+1})$ for all $t \in (t_j', t_{j+1})$; and either $\xi_{av}(t_j') \in L^{-1}(\sigma_j)$ or $\xi_{av}(t_j') \in L^{-1}(\sigma_{j+1})$.

Next, we define the probability that the solution process ξ_{av} of the po-JDS $\mathcal{S}$ starting from some initial state $\xi_{av}(0) = a \in X_0$ under control policy υ satisfies the specification given by DFA $\mathcal{A}$.

Definition 2.5: The finite trace corresponding to the solution process of a po-JDS $\mathcal{S}$ starting from $a \in X$ and under the control policy υ over a finite-time horizon $[0,T) \subset \mathbb{R}_0^+$, i.e. $\sigma(\xi_{av}) = (\sigma_0, \sigma_1, \ldots, \sigma_j, \ldots, \sigma_{k-1}) \in \Pi^k$ as in Definition 2.4, satisfies a specification given by the language of a DFA $\mathcal{A}$, denoted by $\sigma(\xi_{av}) \models \mathcal{A}$, if there exists $j \in \{0,\ldots,k-1\}$ such that $(\sigma_0, \sigma_1, \ldots, \sigma_j) \in \mathcal{L}(\mathcal{A})$. The probability of satisfaction of the specification given by $\mathcal{A}$ is denoted by $\mathbb{P}\{\sigma(\xi_{av}) \models \mathcal{A}\}$.

Remark 2.6: The set of atomic propositions $\Pi = \{p_0, p_1, \ldots, p_M\}$ and the labeling function $L : X \to \Pi$ provide a measurable partition of the state set $X = \cup_{i=1}^N X_i$ as $X_i := L^{-1}(p_i)$. Without loss of generality, we assume that $X_i \neq \emptyset$ for any i.

D. Problem Definition

Now, we formally define the main synthesis problem considered in this work.

Problem 2.7: Given a po-JDS $\mathcal{S}$ as in (II.1), a specification given by the accepting language of DFA $\mathcal{A} = (Q, Q_0, \Pi, \delta, F)$ over a set of atomic propositions $\Pi = \{p_0, p_1, \ldots, p_M\}$, a labeling function $L : X \to \Pi$, and a real value $\vartheta \in (0,1)$, compute an offline control policy υ (if existing) such that $\mathbb{P}\{\sigma(\xi_{av}) \models \mathcal{A}\} \geq \vartheta$, for all $a \in L^{-1}(p_i)$ and some $i \in \{0,1,\ldots,M\}$.

Finding a solution to Problem 2.7 (if existing) is difficult in general. We should highlight that the proposed approach here is sound in solving the considered synthesis problem. This means that if the proposed method provides a solution to a synthesis problem, then we can formally conclude that the proposed controller renders the given specification with the corresponding lower bound on the probability of satisfaction. However, if the method fails to provide any solution, then there may or may not exist a solution to the original synthesis problem). Our approach is to compute a policy υ together with a lower bound $\underline{\vartheta}$. Our aim is to find the potentially largest lower bound, which can be compared with ϑ and gives policy, i.e., a solution for Problem 2.7 if $\underline{\vartheta} \geq \vartheta$. Instead of computing a control policy that guarantees the lower bound $\underline{\vartheta}$, we compute a policy that guarantees $\mathbb{P}\{\sigma(\xi_{av}) \models \mathcal{A}^c\} \leq \bar{\vartheta}$, for any $a \in L^{-1}(p_i)$ and some $i \in \{0,1,\ldots,M\}$. Then for the same control policy the lower bound can be easily obtained as $\underline{\vartheta} = 1 - \bar{\vartheta}$. This is done by constructing a DFA $\mathcal{A}^c$ whose language is the complement of the language of DFA $\mathcal{A}$. To synthesize a controller, we utilize the notion of control barrier functions defined for augmented jump-diffusion system $\tilde{\mathcal{S}}$ introduced in the next section.

III. Control Barrier Functions

In this section, we provide sufficient conditions using so-called control barrier functions under which we can provide the upper bound on the probability that the trajectories of system $\mathcal{S}$ starting from any initial state in $X_0 \subseteq X$ reach $X_1 \subseteq X$. To provide a result giving an upper bound on the reachability probability for the trajectory of $\mathcal{S}$, we provide conditions on barrier functions constructed over the augmented system $\tilde{\mathcal{S}}$.

Theorem 3.1: Consider a po-JDS $\mathcal{S}$ as in (II.1), its estimator $\hat{\mathcal{S}}$ as in (II.2), the resulting augmented system $\tilde{\mathcal{S}}$ as in (II.3) and sets $X_0, X_1 \subseteq X$. Suppose there exists a twice differentiable function $B : X \times X \to \mathbb{R}_0^+$, constants $c \geq 0$ and $\gamma \in [0,1)$ such that

$$\forall (x,\hat{x}) \in X_0 \times X_0, \qquad B(x,\hat{x}) \leq \gamma, \tag{III.1}$$

$$\forall (x,\hat{x}) \in X_1 \times X, \qquad B(x,\hat{x}) \geq 1, \tag{III.2}$$

$$\forall \hat{x} \in X, \exists u \in U, \forall x \in X, \quad \mathcal{D}B(x,\hat{x},u) \leq c. \tag{III.3}$$

Then the probability that the solution process ξ_{av} of the system $\mathcal{S}$ starts from any initial state $a \in X_0$ and reaches region X_1 under the control policy υ within time horizon $[0,T) \subset \mathbb{R}_0^+$ is upper bounded by $\gamma + cT$.

Proof: By using (III.1) and the fact that $X_1 \times X \subseteq \{(x,\hat{x}) \in X \times X \mid B(x,\hat{x}) \geq 1\}$, we have $\mathbb{P}\{\xi_{av}(t) \in X_1 \wedge \hat{\xi}_{\hat{a}v}(t) \in X \ \exists t \in [0,T) \mid a, \hat{a}\} \leq \mathbb{P}\{\sup_{0\leq t\leq T} B(\xi_{av}(t), \hat{\xi}_{\hat{a}v}(t)) \geq 1 \mid a, \hat{a}\} \leq B(a,\hat{a}) + cT \leq \gamma + cT$. The second inequality is obtained by utilizing the result of [22, Theorem 1]. This implies that the probability of the augmented trajectory of $\tilde{\mathcal{S}}$ staring from any $(a,\hat{a}) \in X_0 \times X_0$ and reaching $X_1 \times X$ is upper bounded by $\gamma + cT$.
Now we get $\mathbb{P}\{\xi_{av}(t) \in X_1 \wedge \hat{\xi}_{\hat{a}v}(t) \in X \ \exists t \in [0,T) \mid a, \hat{a}\} \leq \mathbb{P}\{\xi_{av}(t) \in X_1 \ \exists t \in [0,T) \mid a\} + \mathbb{P}\{\hat{\xi}_{\hat{a}v}(t) \in X \ \exists t \in [0,T) \mid \hat{a}\} - \mathbb{P}\{\xi_{av}(t) \in X_1 \vee \hat{\xi}_{\hat{a}v}(t) \in X \ \exists t \in [0,T) \mid a, \hat{a}\}$. Since, the second and last terms trivially hold with probability 1, one has $\mathbb{P}\{\xi_{av}(t) \in X_1 \wedge \hat{\xi}_{\hat{a}v}(t) \in X \ \exists t \in [0,T) \mid a, \hat{a}\} \leq \mathbb{P}\{\xi_{av}(t) \in X_1 \ \exists t \in [0,T) \mid a\}$. Now, since the right term of the `and` (i.e. $\wedge$) is held for all time, the inequality above becomes an equality and one gets $\mathbb{P}\{\xi_{av}(t) \in X_1 \ \exists t \in [0,T) \mid a\} \leq \gamma + Tc$ which concludes the proof. ■

The function B in Theorem 3.1 satisfying (III.1)-(III.3) is usually referred to as the control barrier function.

Remark 3.2: Condition (III.3) implicitly associates a stationary controller $\mathsf{u} : X \to U$ according to the existential quantifier on u for any $\hat{x} \in X$ and is independent of choice of $x \in X$. The stationary control policy υ driving the system is readily given by $\upsilon(t) = \mathsf{u}(\hat{\xi}_{av}(t))$, where $\hat{\xi}_{av}$ is the solution process of the estimator.

IV. Formal Synthesis of Controllers

To synthesize control policies using control barrier functions enforcing specifications expressed by DFA $\mathcal{A}$, we first provide the decomposition of specifications into sequential reachability tasks which will later be solved using control barrier functions.

A. Decomposition into Sequential Reachability

Consider a DFA $\mathcal{A}$ expressing the properties of interest for the system $\mathcal{S}$. Consider DFA $\mathcal{A}^c = (Q, Q_0, \Pi, \delta, F)$ whose language is the complement of the language of DFA $\mathcal{A}$. The sequence $\mathbf{q} = (q_0, q_1, \ldots, q_k) \in Q^{k+1}$, $k \in \mathbb{N}$ is called an accepting state run if $q_0 \in Q_0$, $q_k \in F$, and there exists a finite word $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_{k-1}) \in \Pi^k$ such that $q_i \xrightarrow{\sigma_i} q_{i+1}$ for all $i \in \{0, 1, \ldots, k-1\}$. We denote the finite word corresponding to accepting state run $\mathbf{q}$ by $\sigma(\mathbf{q})$. We also indicate the length of $\mathbf{q} \in Q^{k+1}$ by $|\mathbf{q}|$, which is $k+1$. Let $\mathcal{R}$ be the set of all finite accepting state runs starting from $q_0 \in Q_0$ excluding self-loops, where

$$\mathcal{R} := \{\mathbf{q} = (q_0, q_1, \ldots, q_k) \in Q^{k+1} \mid q_k \in F, q_i \neq q_{i+1}, \forall i < k\}.$$

Computation of $\mathcal{R}$ can be done algorithmically by viewing $\mathcal{A}^c$ as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = Q$ and edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ such that $(q, q') \in \mathcal{E}$ if and only if $q' \neq q$ and there exist $p \in \Pi$ such that $q \xrightarrow{p} q'$. For any $(q, q') \in \mathcal{E}$, we donate the atomic proposition associated with the edge (q, q') by $\sigma(q, q')$. From the construction of the graph, it is obvious that the finite path in the graph starting from vertices $q_0 \in Q_0$ and ending at $q_F \in F$ is an accepting state run $\mathbf{q}$ of $\mathcal{A}^c$ without any self-loop and therefore belongs to $\mathcal{R}$. One can easily compute $\mathcal{R}$ using depth first search algorithm [23]. For each $p \in \Pi$, we define a set $\mathcal{R}^p$ as

$$\mathcal{R}^p := \{\mathbf{q} = (q_0, q_1, \ldots, q_k) \in \mathcal{R} \mid \sigma(q_0, q_1) = p\}. \quad \text{(IV.1)}$$

Decomposition into sequential reachability is performed as follows. For any $\mathbf{q} = (q_0, q_1, \ldots, q_k) \in \mathcal{R}^p$ $\forall p \in \Pi$, we define $\mathcal{P}^p(\mathbf{q})$ as a set of all state runs of length 3,

$$\mathcal{P}^p(\mathbf{q}) := \{(q_i, q_{i+1}, q_{i+2}) \mid 0 \leq i \leq k-2\}. \quad \text{(IV.2)}$$

Now, we define $\mathcal{P}(\mathcal{A}^c) := \bigcup_{p \in \Pi} \bigcup_{\mathbf{q} \in \mathcal{R}^p} \mathcal{P}^p(\mathbf{q})$.

Remark 4.1: Note that $\mathcal{P}^p(\mathbf{q}) = \emptyset$ for $|\mathbf{q}| = 2$. In fact, any accepting state run of length 2 specifies a subset of the state set such that the system satisfies $\mathcal{A}^c$ whenever it starts from that subset. This gives trivial zero probability for satisfying the specification, thus neglected in the sequel.

For the illustration of the above sets, we kindly refer the interested reader to Example 1 in [8]. Having $\mathcal{P}^p(\mathbf{q})$ in (IV.2) as the set of state runs of length 3, in this subsection, we provide a systematic approach to compute a policy together with a (potentially tight) lower bound on the probability that the solution process of $\mathcal{S}$ satisfies the specifications given by DFA $\mathcal{A}$. Given a DFA $\mathcal{A}^c$, our approach relies on performing a reachability computation over each element of $\mathcal{P}(\mathcal{A}^c)$ (*i.e.*, $\bigcup_{p \in \Pi} \bigcup_{\mathbf{q} \in \mathcal{R}^p} \mathcal{P}^p(\mathbf{q})$), where reachability probability is upper bounded using control barrier functions along with appropriate choices of control inputs as mentioned in Theorem 3.1. However, computation of control barrier functions and the policies for each element $\nu \in \mathcal{P}(\mathcal{A}^c)$, can cause ambiguity while utilizing controllers in closed-loop whenever there are more than one outgoing edges from a state of the automaton. To resolve this ambiguity, we simply merge such reachability problems into one reachability problem by replacing the reachable set $X_1 \times X$ in Theorem 3.1 with the union of regions corresponding to the alphabets of all outgoing edges. Thus we get a common control barrier function and a corresponding controller. This enables us to partition $\mathcal{P}(\mathcal{A}^c)$ and put the elements sharing a common control barrier function and a corresponding controller in the same partition set. These sets can be formally defined as

$$\mu_{(q,q',\Delta(q'))} := \{(q, q', q'') \in \mathcal{P}(\mathcal{A}^c) \mid q, q', q'' \in Q \text{ and } q'' \in \Delta(q')\}.$$

The control barrier function and the controller (as discussed in Remark 3.2) corresponding to the partition set $\mu_{(q,q',\Delta(q'))}$ are denoted by $B_{\mu_{(q,q',\Delta(q'))}}(x, \hat{x})$ and $\mathsf{u}_{\mu_{(q,q',\Delta(q'))}}(\hat{x})$, respectively. Thus, for all $\nu \in \mathcal{P}(\mathcal{A}^c)$, we have

$$B_\nu(x, \hat{x}) = B_{\mu_{(q,q',\Delta(q'))}}(x, \hat{x}) \text{ and } \mathsf{u}_\nu(\hat{x}) = \mathsf{u}_{\mu_{(q,q',\Delta(q'))}}(\hat{x}), \text{ if } \nu \in \mu_{(q,q',\Delta(q'))}. \quad \text{(IV.3)}$$

B. Control Policy

From the above discussion, one can readily observe that we have different control policies at different locations of the automaton which can be interpreted as a switching control policy. Next, we define the automaton representing the switching mechanism for control policies. Consider the DFA $\mathcal{A}^c = (Q, Q_0, \Pi, \delta, F)$ corresponding to the complement of DFA $\mathcal{A}$ as discussed in Section IV-A, where $\Delta(q)$ denotes the set of all successor states of $q \in Q$. Now, the switching mechanism is given by a DFA $\mathcal{A}_\mathfrak{m} = (Q_\mathfrak{m}, Q_{\mathfrak{m}0}, \Pi_\mathfrak{m}, \delta_\mathfrak{m}, F_\mathfrak{m})$, where $Q_\mathfrak{m} := Q_{\mathfrak{m}0} \cup \{(q, q', \Delta(q')) \mid q, q' \in Q \setminus F\} \cup F_\mathfrak{m}$ is the set of states, $Q_{\mathfrak{m}0} := \{(q_0, \Delta(q_0)) \mid q_0 \in Q_0\}$ is the set of initial states, $\Pi_\mathfrak{m} = \Pi$, $F_\mathfrak{m} = F$, and the transition relation $(q_\mathfrak{m}, \sigma, q'_\mathfrak{m}) \in \delta_\mathfrak{m}$ is defined as

- for all $q_\mathfrak{m} = (q_0, \Delta(q_0)) \in Q_{\mathfrak{m}0}$, $(q_0, \Delta(q_0)) \xrightarrow{\sigma(q_0,q'')} (q_0, q'', \Delta(q''))$, where $q_0 \xrightarrow{\sigma(q_0,q'')} q''$;
- for all $q_\mathfrak{m} = (q, q', \Delta(q')) \in Q_\mathfrak{m} \setminus (Q_{\mathfrak{m}0} \cup F_\mathfrak{m})$,
  - $(q, q', \Delta(q')) \xrightarrow{\sigma(q',q'')} (q', q'', \Delta(q''))$, such that $q, q', q'' \in Q$, $q' \xrightarrow{\sigma(q',q'')} q''$, and $q'' \notin F$; and
  - $(q, q', \Delta(q')) \xrightarrow{\sigma(q',q'')} q''$, such that $q, q', q'' \in Q$, $q' \xrightarrow{\sigma(q',q'')} q''$, and $q'' \in F$.

The hybrid controller defined over augmented state-space $X \times Q_\mathfrak{m}$ that is a candidate for solving Problem 2.7 is given by

$$\tilde{\mathsf{u}}(\hat{x}, q_\mathfrak{m}) = \mathsf{u}_{\mu_{(q'_\mathfrak{m})}}(\hat{x}), \quad \forall (q_\mathfrak{m}, L(\hat{x}), q'_\mathfrak{m}) \in \delta_\mathfrak{m}. \quad \text{(IV.4)}$$

The corresponding hybrid control policy υ is given by $\upsilon(t) = \tilde{\mathsf{u}}(\hat{\xi}(t), q_\mathfrak{m})$. For the illustration of the switching mechanism, see Example 1 in [8, Section 5]. In the next subsection, we discuss the computation of bound on the probability of satisfying the specification under such a policy, which then can be used for checking if this policy is indeed a solution for Problem 2.7.

C. Computation of Probability

The next theorem provides an upper bound on the probability that the solution process satisfies the specifications given by $\mathcal{A}$.

Theorem 4.2: For a specification given by the accepting language of DFA $\mathcal{A}$, let $\mathcal{A}^c$ be the DFA corresponding to the

complement of $\mathcal{A}$, $\mathcal{R}^p$ be the set defined in (IV.1), and $\mathcal{P}^p$ be the set of runs of length 3 defined in (IV.2). Then the probability that the solution process of the system $\mathcal{S}$ starting from any initial state $a \in L^{-1}(p)$ under the hybrid control policy υ associated with the hybrid controller (IV.4) satisfies $\mathcal{A}^c$ within time horizon $[0, T)$ is upper bounded by

$$\mathbb{P}\{\sigma(\xi_{a\upsilon}) \models \mathcal{A}^c\} \leq \sum_{\mathbf{q} \in \mathcal{R}^{\mathbf{p}}} \prod \{(\gamma_\nu + c_\nu T) \,|\, \nu = (q, q', q'') \in \mathcal{P}^p(\mathbf{q})\}, \tag{IV.5}$$

where $\gamma_\nu + c_\nu T$ is the upper bound on the probability that the solution process of $\mathcal{S}$ starts from $X_0 := L^{-1}(\sigma(q, q'))$ and reaches $X_1 := L^{-1}(\sigma(q', q''))$ under control policy υ within time horizon $[0, T)$ which is computed via Theorem 3.1.

Proof: The proof is similar to that of [8, Theorem 5.2] and is omitted here due to the lack of space. ∎

Theorem 4.2 enables us to decompose the specification into a collection of sequential reachabilities, compute bounds on the reachability probabilities using Theorem 3.1, and then combine the bounds in a sum-product expression.

Remark 4.3: In case we are unable to find control barrier functions for some of the elements $\nu \in \mathcal{P}^p(\mathbf{q})$ in (IV.5), we replace the related term $(\gamma_\nu + c_\nu T)$ by the pessimistic bound 1 and apply random control input. In order to get a non-trivial bound in (IV.5), at least one control barrier function must be found for each $\mathbf{q} \in \mathcal{R}^p$.

Corollary 4.4: Given the result of Theorem 4.2, the probability that the solution process of $\mathcal{S}$ starts from any $a \in L^{-1}(p)$ under control policy υ and satisfies specifications given by DFA $\mathcal{A}$ over time horizon $[0, T) \subset \mathbb{R}_0^+$ is lower-bounded by

$$\mathbb{P}\{\sigma(\xi_{a\upsilon}) \models \mathcal{A}\} \geq 1 - \mathbb{P}\{\sigma(\xi_{a\upsilon}) \models \mathcal{A}^c\}.$$

D. Computation of Control Barrier Functions

Proving the existence of a control barrier function and finding one are in general hard problems. However, if functions f, h, g_1, g_2, r_1, and r_2 are polynomial with respect to their arguments and partition sets $X_i = L^{-1}(p_i), i \in \{0, 1, 2, \ldots, M\}$, are bounded semi-algebraic sets (i.e., they can be represented by polynomial (in)equalities), one can formulate conditions in Theorem 3.1 as a sum-of-squares (SOS) optimization problem. See [8, Section 5.3.1.] for a detailed discussion on a similar approach. Having an SOS optimization problem, one can efficiently search for a polynomial control barrier function $B_\nu(x, \hat{x})$ and controller $\mathsf{u}_\nu(\hat{x})$, for any $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$ as in (IV.3) using SOSTOOLS [24] in conjunction with a semidefinite programming solver such as SeDuMi [25] while minimizing constants γ_ν and c_ν. Having values of γ_ν and c_ν for all $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, one can simply utilize results of Theorem 4.2 and Corollary 4.4 to compute a lower bound on the probability of satisfying the given specification. Note that it may not be possible in advance to obtain a probability bound that is meaningful, in such cases the order of a control barrier function needs to increase to achieve the desired probability bound.

Remark 4.5: Under the assumption that sets X, X_0, and X_1 in Theorem 3.1 are compact and input set U is finite, one can utilize counterexample guided inductive synthesis (CEGIS) approach to search for barrier control functions for more general nonlinear functions f, h, g_1, g_2, r_1, and r_2 in (II.1). For more detailed discussion on CEGIS approach, we kindly refer interested readers to the algorithm in [8, Section 5.3.2.].
Computational Complexity: The number of triplets and hence the number of control barrier functions needed to be computed are bounded by $|Q|^3$, where $|Q|$ is the number of states in DFA $\mathcal{A}$. However, this is the worst-case bound and in practice, the number of control barrier functions is much smaller. In the case of sum-of-squares optimization approach, the computational complexity of finding polynomial control barrier functions depends on both the degree of polynomials and the number of state variables. One can easily see that for fixed polynomial degrees, the required computations grow polynomially with respect to the dimension of the augmented system. For the CEGIS approach, due to its iterative nature and lack of guarantee on termination, it is difficult to provide any analysis on the computational complexity.

V. Case Study

We consider a nonlinear Moore-Greitzer jet engine model in no-stall mode [26] as a partially observed jump-diffusion systems by adding noise and jump terms which is given by:

$$\begin{aligned}
\mathrm{d}\,\xi_1 &= (-\xi_2 - \frac{3}{2}\xi_1^2 - \frac{1}{2}\xi_1^3)\,\mathrm{d}\,t + 0.2\,\mathrm{d}\,W_{11t} + 0.9\,\mathrm{d}\,P_t,\\
\mathrm{d}\,\xi_2 &= (\xi_1 - \upsilon)\,\mathrm{d}\,t + 0.06\,\mathrm{d}\,W_{12t},\\
\mathrm{d}\,y &= \xi_2\,\mathrm{d}\,t + 0.06\,\mathrm{d}\,W_{2t},
\end{aligned}$$

where $\xi = [\xi_1, \xi_2]^T$, $\xi_1 = \Phi - 1$, $\xi_2 = \Psi - \psi - 2$, Φ is the mass flow, Ψ is the pressure rise, and ψ is a constant. Terms W_{11t}, W_{12t}, and W_{2t} denote the standard Brownian motions and P_t denotes the Poisson process with rate $\lambda = 5$. We consider a compact state set $X = [-1, 3] \times [-4, 4]$ and regions of interest $X_0 = [0, 1] \times [-1, 1]$, $X_1 = [-1, -0.2] \times [-4, -2.5]$, $X_2 = [1, 3] \times [2, 4]$, and $X_3 = X \setminus (X_0 \cup X_1 \cup X_2)$. The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3\}$ with labeling function $L(x_j) = p_j$ for all $x_j \in X_j$, $j \in \{0, 1, 2, 3\}$. The objective here is to compute a control policy that provides a lower bound on the probability that the trajectories of the system satisfy the specification given by the accepting language of the DFA $\mathcal{A}$ given in Figure 1 over finite time-horizon $[0, T = 10)$. Language of $\mathcal{A}$ entails that if we start in X_0 then the system will always stay away from X_1 or X_2. The corresponding DFA $\mathcal{A}^c$ accepting complement of $\mathcal{L}(\mathcal{A})$ is shown in Figure 1. Following Subsection IV-A, we only need to compute a control barrier function corresponding to triplet (q_0, q_1, q_2).

Now with an estimator gain in (II.2) as $K = [6.1394, 7.8927]^T$, we use SOSTOOLS and SeDuMi to compute a sum-of-squares polynomial control barrier function $B(x, \hat{x})$ of order 4, sum-of-square polynomials $\psi_0(x, \hat{x})$, $\psi_1(x, \hat{x})$, $\psi(x, \hat{x})$ of order 4, with total 1125 coefficients resulting in a computation time of about 15 minutes. The corresponding controller of order 2 is obtained as follows:

$$\mathsf{u}(\hat{x}) = 0.7321\hat{x}_1 - 1.8612\hat{x}_1\hat{x}_2 - 1.4356\hat{x}_2. \tag{V.1}$$

The values of $\gamma = 0.099$ and $c = 1 \times 10^{-5}$ are obtained using bisection method resulting in $\mathbb{P}\{\sigma(\xi_{a\upsilon}) \models \mathcal{A}\} \geq 0.89$

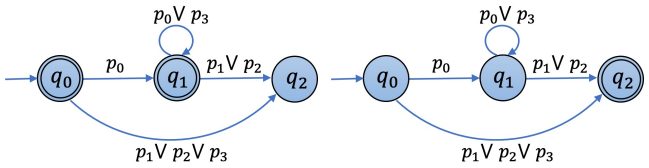

Fig. 1. The DFA $\mathcal{A}$ representing specification (left) and the DFA $\mathcal{A}^c$ representing complement of $\mathcal{A}$ (right).

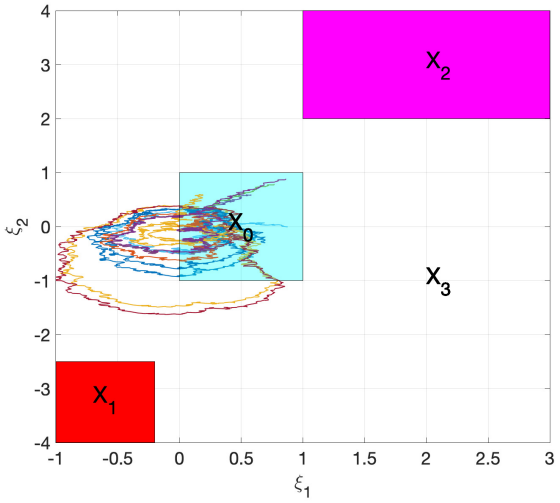

Fig. 2. A few closed loop trajectories starting from different initial conditions in X_0 under controller (V.1).

for all $x_0 \in L^{-1}(p_0)$, as discussed in Subsection IV-D. One can see that only one controller is enough for enforcing the specification, thus we do not need any switching mechanism. Figure 2 shows a few trajectories starting from different initial conditions under the control policy (V.1).

VI. Conclusions

In this paper, we proposed a discretization-free approach for the formal controller synthesis of partially observed jump-diffusion systems. The proposed method computes a hybrid control policy together with a lower bound on the probability of satisfying complex temporal logic specifications given by the accepting language of DFA $\mathcal{A}$ over a finite-time horizon. This is achieved by constructing control barrier functions over an augmented system consisting of both the system and the estimator. As a result, the probability bound is computed without requiring any prior information of estimation accuracy.

References

[1] C. Belta, B. Yordanov, and E. A. Gol, *Formal methods for discrete-time dynamical systems*. Springer, 2017, vol. 89.

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

[3] C. Belta, B. Yordanov, and E. A. Gol, "Discrete-time dynamical systems," in *Formal Methods for Discrete-Time Dynamical Systems*. Springer, 2017, pp. 111–118.

[4] M. Zamani, P. M. Esfahani, R. Majumdar, A. Abate, and J. Lygeros, "Symbolic control of stochastic systems via approximately bisimilar finite abstractions," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3135–3150, 2014.

[5] M. Zamani, I. Tkachev, and A. Abate, "Towards scalable synthesis of stochastic control systems," *Discrete Event Dynamic Systems*, vol. 27, no. 2, pp. 341–369, 2017.

[6] A. Lavaei, S. Soudjani, and M. Zamani, "Compositional (in) finite abstractions for large-scale interconnected stochastic systems," *IEEE Transactions on Automatic Control*, 2020.

[7] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[8] P. Jagtap, S. Soudjani, and M. Zamani, "Formal synthesis of stochastic systems via control barrier certificates," *arXiv preprint arXiv:1905.04585*, 2019.

[9] P. Jagtap, A. Swikir, and M. Zamani, "Compositional construction of control barrier functions for interconnected control systems," in *Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control*, 2020, pp. 1–11.

[10] C. Huang, X. Chen, W. Lin, Z. Yang, and X. Li, "Probabilistic safety verification of stochastic hybrid systems using barrier certificates," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 16, no. 5s, p. 186, 2017.

[11] A. Clark, "Control barrier functions for complete and incomplete information stochastic systems," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 2928–2935.

[12] N. Jahanshahi, P. Jagtap, and M. Zamani, "Synthesis of stochastic systems with partial information via control barrier functions," *21st IFAC World Congress*, 2020.

[13] B. Øksendal and A. Sulem, *Applied stochastic control of jump diffusions*. Springer Science & Business Media, 2007.

[14] X. Kai, C. Wei, and L. Liu, "Robust extended kalman filtering for nonlinear systems with stochastic uncertainties," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 2, pp. 399–405, 2009.

[15] B.-S. Chen, W.-H. Chen, and H.-L. Wu, "Robust h_2 / h_∞ global linearization filter design for nonlinear stochastic systems," *IEEE transactions on circuits and systems I: Regular Papers*, vol. 56, no. 7, pp. 1441–1454, 2008.

[16] C.-S. Tseng, "Robust fuzzy filter design for a class of nonlinear stochastic systems," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 2, pp. 261–274, 2007.

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

[18] F. Bonchi and D. Pous, "Checking nfa equivalence with bisimulations up to congruence," *ACM SIGPLAN Notices*, vol. 48, no. 1, pp. 457–468, 2013.

[19] J. E. Hopcroft, R. Motwani, and J. D. Ullman, "Introduction to automata theory, languages, and computation," *Acm Sigact News*, vol. 32, no. 1, pp. 60–65, 2001.

[20] G. De Giacomo and M. Vardi, "Synthesis for ltl and ldl on finite traces," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[21] T. Wongpiromsarn, U. Topcu, and A. Lamperski, "Automata theory meets barrier certificates: Temporal logic verification of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3344–3355, 2015.

[22] H. Kushner, "Stochastic stability and control, ser," *Mathematics in Science and Engineering. New York: Academic Press*, 1967.

[23] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Pearson Education, 2003.

[24] S. Prajna, A. Papachristodoulou, and P. A. Parrilo, "Introducing sostools: A general purpose sum of squares programming solver," in *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, vol. 1. IEEE, 2002, pp. 741–746.

[25] J. F. Sturm, "Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones," *Optimization methods and software*, vol. 11, no. 1-4, pp. 625–653, 1999.

[26] M. Krstic and P. V. Kokotovic, "Lean backstepping design for a jet engine compressor model," in *Proceedings of International Conference on Control Applications*. IEEE, 1995, pp. 1047–1052.