Discrete State Estimators for a Class of Nondeterministic Hybrid Systems on a Lattice

D. Del Vecchio and R. M. Murray

Abstract— **The problem of estimating the discrete variables in nondeterministic hybrid systems where the continuous variables are available for measurement is considered. Using partial order theory, we construct a discrete state estimator, the LU estimator, which updates the lower (L) and upper (U) bounds of the set of all possible discrete variables values compatible with the output sequence and with the systems' dynamics. If the system is weakly observable, we show that there always exist a lattice on which to construct the LU estimator. For computational issues, some partial orders are to be preferred to others. We thus show that nondeterminism may be added to a system to obtain a new system that satisfies the requirements for the construction of the LU estimator on a chosen lattice. These ideas are applied to a nondeterministic multi-robot system.**

I. INTRODUCTION

The problem of estimating discrete variables in nondeterministic hybrid systems where the continuous variables are available for measurement is considered. This scenario has already practical interest as in the case of decentralized multi-robot systems, such as are found in robot soccer, where the continuous variables represent physical quantities such as position and velocity, and discrete variables represent the state of the internal logical system or communication protocol used by the robots to coordinate their actions. The estimation problem involves estimating the internal discrete variables of the system given the evolution of the continuous physical variables. The systems considered in this paper are not probabilistic, as nondeterminism arises because a state can be updated to a known set of possible values instead of being updated to one possible value only. This is a way of taking modeling uncertainty into account.

For deterministic systems, the problem of estimating and tracking the values of non-measurable variables in hybrid systems has been investigated by several authors. Bemporad et al. [3] show that observability properties are hard to check for hybrid systems and an observer is proposed that requires large amounts of computation. In [14] an algebraic check is proposed to determine the observability property of a jump linear system, and the state change is detected *a posteriori* assuming a minimum time between switches. In this work, the time scales of the discrete and continuous dynamics are comparable, and the state of the system is tracked. For non-deterministic systems, [10] studies observability

R. M. Murray is in the Department of Control and Dynamical Systems at the California Institute of Technology, USA `murray@cds.caltech.edu`

D. Del Vecchio is in the Department of Control and Dynamical Systems at the California Institute of Technology, USA `ddomitilla@cds.caltech.edu`

conditions for exact reconstruction of the current state after each system event, and [5] consider the problem of finding optimal control strategies for partially observable Markov-decision processes. The systems treated in this paper are not probabilistic, and they are closer to the one proposed in [10], excepts that exact reconstruction of the state after each event is not required. In the deterministic case, [4] and [7] show that the complexity of the observer often arises from the need to compute maps on large sets of values, corresponding to the set of all possible internal states compatible with the observed output sequence. These same difficulties are encountered in [11], where the proposed observer fails to be applicable for large problem sizes. In the models that we consider, due to the heavy coupling of discrete and continuous variables evolution, discrete state estimation strategies where the analysis of the continuous signal suffices for determining the discrete state, such as the ones proposed by [1], are not applicable. Also, the time scales of the continuous and discrete dynamics are comparable, so that we have no guarantee that the system remains in each discrete mode for a sufficiently long time as assumed in [2].

In [12] some of the complexity issues, such as those encountered in [11] or [4], [7], were avoided by finding a good way of representing the sets of interest and of computing maps on them. In particular, a system defined on its space of variables, is extended to a larger space of variables equipped with lattice structure to obtain an extended system. Provided this extended system satisfies certain requirements, a discrete state estimator, the LU estimator, can be constructed, which updates the least and greatest element of the set of all values of discrete variables compatible with the output sequence and with the dynamics of the system. This work builds on these ideas to generalize to nondeterministic systems. If the system is weakly observable, a lattice always exists on which the LU estimator can be constructed. This shows that the lattice approach to estimation generally applies to observable systems. For complexity reasons, those partial orders that allow the use of algebraic properties for the computation of their elements are to be preferred. We show that nondeterminism can be added to the system to obtain a new system that satisfies the properties needed for the construction of the LU estimator on a chosen lattice. This way, the complexity of the estimator can be reduced by paying the price of having a possibly slower convergence rate. This is a compromise between performance and complexity.

The contents of this paper are as follows. In Section II,

basic definitions on transition systems, and basic definitions on partial order theory are reviewed. In Section III, the LU discrete state estimator is constructed, and in Section IV its existence is investigated. In Section V, a way is proposed to construct the estimator on a chosen lattice. These ideas are applied to a multi-robot system in Section VI.

II. BASIC DEFINITIONS

Let S be the set of states with $s \in S$. A *transition function* on S is a function $F : S \longrightarrow 2^S$ which updates the state s to a new set of states $s' \in F(s)$, with $2^S := \{A | A \subseteq S\}$. Given a transition function F, an *execution* of F is a sequence $\sigma = \{s(k)\}_{k \in \mathbb{N}}$ such that $s(k+1) = F(s(k))$ for all $k \in \mathbb{N}$. The set of all executions of F is denoted $\mathscr{E}(F)$.

Definition 2.1: Let F be a transition function on a set of states S, the set $\Omega \subset S$ is the *ω-limit set* of F, denoted $\omega(F)$, if it is the smallest set such that the following hold:

(i) if $s \in \Omega$ and $s' \in F(s)$, then $s' \in \Omega$;

(ii) for each $\sigma \in \mathscr{E}(F)$, there exists a time k_σ such that $\sigma(k_\sigma) \in \Omega$ for all $k \geq k_\sigma$.

We now recall the notion of observability for transition systems as it can be found in [11].

Definition 2.2: Given a transition function F and an output map $g : S \to \mathscr{Y}$, for some $\mathscr{Y}$, two executions $\sigma_1, \sigma_2 \in \mathscr{E}(F)$ are *weakly equivalent*, denoted $\sigma_1 \sim \sigma_2$, if there exists k^* such that $\sigma_1(k^*) \notin \omega(F)$ and $\sigma_1(k) = \sigma_2(k)$ for all $k \geq k^*$.

Definition 2.3: The transition function F is said to be *observable* with respect to the output function $g : S \to \mathscr{Y}$ if whenever σ_1 is not weakly equivalent to σ_2 then there is k such that $g(\sigma_1(k)) \neq g(\sigma_2(k))$.

We will consider transition systems where $s = (\alpha, z)$, where $\alpha \in \mathscr{U}$ is the discrete part of the state with $\mathscr{U}$ a finite discrete set, and $z \in \mathscr{Z}$ is the continuous portion of the state, for example $\mathscr{Z} = \mathbb{R}^N$. In such a case $F = (f, h)$, where $f : \mathscr{U} \times \mathscr{Z} \to 2^{\mathscr{U}}$ is the function that updates the discrete state, and $h : \mathscr{U} \times \mathscr{Z} \to 2^{\mathscr{Z}}$ is the function that updates the continuous part of the state. We will denote such nondeterministic transition systems by the tuple $\Sigma = (f, h, \mathscr{U}, \mathscr{Z})$. The executions of Σ, denoted $\mathscr{E}(\Sigma)$, are of the form $\sigma = \{\alpha(k), z(k)\}_{k \in \mathbb{N}}$, with $\alpha(k+1) \in f(\alpha(k), z(k))$ and $z(k+1) \in h(\alpha(k), z(k))$. Since we assume that the continuous variables z are measured, the output sequence is given by $g(\sigma) = \{y(k)\}_{k \in \mathbb{N}} := \{z(k)\}_{k \in \mathbb{N}}$.

We recall some basic notions from partial order theory (see [6] for more details). A set χ with a partial order relation "$\leq$" is said to be a lattice if any two elements have their supremum and their infimum in $(\chi, \leq)$. Note that the order is partial as some pairs of elements are not related by "$\leq$". For $x, w \in \chi$ their supremum is called join and is denoted by $x \curlyvee w := sup\{x, w\}$, and their infimum is called meet and is denoted by $x \curlywedge w := inf\{x, w\}$. If $S \subset \chi$, we define $\bigvee S := sup(S)$ and $\bigwedge S := inf(S)$. Let $(\chi, \leq)$ be a non-empty ordered set. If $x \curlywedge w$ and $x \curlyvee w$ exist for any $x, w \in \chi$, then $(\chi, \leq)$ is a *lattice*. If $\bigvee S$ and $\bigwedge S$ exist for all $S \subseteq \chi$, then $(\chi, \leq)$ is a *complete lattice*. Let $(\chi, \leq)$ be a lattice and let $S \subseteq \chi$ be a subset of χ. Then S is

a *sublattice* of $(\chi, \leq)$ if $a, b \in S$ implies that $a \curlyvee b \in S$ and $a \curlywedge b \in S$. The cardinality of S is denoted $|S|$. Given a complete lattice $(\chi, \leq)$, an *interval sublattice* is defined as follows. Any interval sublattice of $(\chi, \leq)$ is given by $[L, U] = \{w \in \chi : L \leq w \leq U\}$ for $L, U \in \chi$. That is, this special sublattice can be represented by only two elements. The *power lattice* of a set $\mathscr{U}$, denoted $(\mathscr{P}(\mathscr{U}), \subseteq)$, is given by the power set of $\mathscr{U}$, $\mathscr{P}(\mathscr{U})$ (the set of all subsets of $\mathscr{U}$), ordered according to the set inclusion $\subseteq$.

Definition 2.4: Let $x, w \in \chi$, with $(\chi, \leq)$ a lattice, and $f : \chi \to 2^\chi$ be non deterministic. We say that f is *generalized order preserving* if $x \leq w \implies f(x) \leq f(w)$, where we say that $f(x) \leq f(w)$ if $\bigvee f(x) \leq \bigvee f(w)$ and $\bigwedge f(x) \leq \bigwedge f(w)$. This definition is weaker than the one given for deterministic maps (see [6]), as there can be elements in $f(x)$ greater than elements in $f(w)$.

Definition 2.5: Let $f : \chi \to P$ be non deterministic. We say that f is *onto* if for any $w \in P$ there exists a $x \in \chi$ such that $w \in f(x)$.

The cardinality of P can be greater than the one of χ because one element can be mapped to many.

In the next section we show how to use these notions to construct a discrete state estimator for nondeterministic systems.

III. LU DISCRETE STATE ESTIMATOR: CONSTRUCTION

Consider the nondeterministic transition system $\Sigma = (f, h, \mathscr{U}, \mathscr{Z})$, All pairs (z_1, z_2) for which there exist $\alpha \in \mathscr{U}$ such that $z_2 \in h(\alpha, z_1)$ group the set $\mathscr{U}$ in classes of the form $\{\alpha \in \mathscr{U} : z_2 \in h(\alpha, z_1)\}$. Each class contains all of the α values that allow the transition from z_1 to z_2 through the transition function h. In many practical examples, such as the multi robot examples that we consider, such classes are equivalence classes, as they are all non-intersecting. The following theorem shows that an estimator like the one constructed for deterministic systems in [12] can be constructed for nondeterministic systems as well.

Theorem 3.1: Consider the system $\Sigma = (f, h, \mathscr{U}, \mathscr{Z})$. Assume that

1. Σ is weakly observable;
2. There exist a lattice $(\chi, \leq)$, with $\mathscr{U} \subset \chi$, and extensions $\tilde{f} : \chi \times \mathbb{R}^N \to 2^\chi$ and $\tilde{h} : \chi \times \mathscr{Z} \to 2^{\mathscr{Z}}$, with $f = \tilde{f}|_{\mathscr{U} \times \mathscr{Z}} \cap \mathscr{U}$ and $\tilde{h}|_{\mathscr{U} \times \mathscr{Z}} = h$, such that

   2a. $A_y(k) = \{x \in \chi : y(k+1) \in \tilde{h}(x, y(k))\} = [l_y(k), u_y(k)]$ for some $l_y(k), u_y(k) \in \chi$ for any k;

   2b. $\tilde{f} : ([L, U], y(k)) \longrightarrow [\bigwedge \tilde{f}(L, y(k)), \bigvee \tilde{f}(U, y(k))]$ is generalized order preserving for any $[L, U] \subseteq [l_y(k), u_y(k)]$;

   2c. $\tilde{f} : ([L, U], y(k)) \cap \mathscr{U} \longrightarrow [\bigwedge \tilde{f}(L, y(k)), \bigvee \tilde{f}(U, y(k))] \cap \mathscr{U}$ is onto for any $[L, U] \subseteq [l_y(k), u_y(k)]$;

then the deterministic system

$$
\begin{aligned}
L(k+1) &= \bigwedge \tilde{f}((L(k) \curlyvee l_y(k)), y(k)) \\
U(k+1) &= \bigvee \tilde{f}((U(k) \curlywedge u_y(k)), y(k))
\end{aligned}
\tag{1}
$$

with $L(0) = \bigwedge \chi$ and $U(0) = \bigvee \chi$, is such that

(i) $\alpha(k) \in [L(k), U(k)]$ for all k;

(ii) $|[L(k), U(k)] \cap \mathcal{U}| \longrightarrow 1$ as $k \to \infty$.

Proof: (Sketch). The proof is similar to the one in [12], except that now the function $\tilde{f}$ is nondeterministic, and thus one has to carry out the arguments using $\bigvee \tilde{f}$ and $\bigwedge \tilde{f}$ as opposed to $\tilde{f}$ itself. This is sketched in what follows. For simplifying notation, we omit the dependence of $\tilde{f}$ on y.

Item (i) can be proved by induction on k. By the initialization of the estimator $L(0) \le \alpha(0) \le U(0)$ (base case). Assume that $L(k) \le \alpha(k) \le U(k)$, and show this holds at step $k+1$. It suffices to notice that $l_y(k) \curlyvee L(k) \le \alpha(k) \le U(k) \curlywedge u_y(k)$, because $\alpha(k) \in A_y(k)$ by definition. By the order preserving property of $\tilde{f}$, we have

$$\bigwedge \tilde{f}(l_y(k) \curlyvee L(k)) \le \bigwedge \tilde{f}(\alpha(k)) \le \alpha(k+1)$$

and

$$\alpha(k+1) \le \bigvee \tilde{f}(\alpha(k)) \le \bigvee \tilde{f}(U(k) \curlywedge u_y(k)).$$

Item (ii) is proved by contradiction. Assume $\beta_1'', \beta_2'' \in [L(k+1), U(k+1)] \cap \mathcal{U}$. By equations (1) and by property 2c., there are $\beta_1', \beta_2' \in [l_y(k) \curlyvee L(k), U(k) \curlywedge u_y(k)] \cap \mathcal{U}$ such that $\beta_1'' \in f(\beta_1')$ and $\beta_2'' \in f(\beta_2')$, and $\beta_1', \beta_2' \in A_y(k)$. In analogous way, there are $\beta_1, \beta_2 \in [l_y(k-1) \curlyvee L(k-1), U(k-1) \curlywedge u_y(k-1)] \cap \mathcal{U}$ such that $\beta_1' \in f(\beta_1)$ and $\beta_2' \in f(\beta_2)$, and $\beta_1, \beta_2 \in A_y(k-1)$. This implies that there are two executions of Σ, $\sigma_1 = \{\beta_1(k), z(k)\}_{k \in \mathbb{N}}$ and $\sigma_2 = \{\beta_2(k), z(k)\}_{k \in \mathbb{N}}$, that share the same output. This contradicts the weak observability of Σ. ∎

Note that the main difference between this theorem and the one holding for the deterministic case ([12]) is that in this case, the system extension cannot be a bijection as it is nondeterministic. As a consequence, also observability needs to be relaxed to weak observability.

IV. LU DISCRETE STATE ESTIMATOR: EXISTENCE

The following theorem shows that it is always possible to find a lattice $(\chi, \le)$ such that there exists extensions $\tilde{f}$ and $\tilde{h}$ that satisfy properties 2. of Theorem 3.1. Therefore, if the system is weakly observable an LU estimator can be constructed on $(\chi, \le)$.

Theorem 4.1: Consider the system $\Sigma = (f, h, \mathcal{U}, \mathcal{Z})$. There exists a lattice $(\chi, \le)$, with $\mathcal{U} \subset \chi$, and extensions $\tilde{f} : \chi \times \mathcal{Z} \to 2^{\chi}$, $\tilde{h} : \chi \times \mathcal{Z} \to 2^{\mathcal{Z}}$, with $f = \tilde{f}|_{\mathcal{U} \times \mathcal{Z}} \cap \mathcal{U}$ and $\tilde{h}|_{\mathcal{U} \times \mathcal{Z}} = h$, such that

(i) $A_y(k) = \{x \in \chi : y(k+1) \in \tilde{h}(x, y(k))\} = [l_y(k), u_y(k)]$ for some $l_y(k), u_y(k) \in \chi$ for any k;

(ii) $\tilde{f} : ([L, U], y(k)) \longrightarrow [\bigwedge \tilde{f}(L, y(k)), \bigvee \tilde{f}(U, y(k))]$ is generalized order preserving for any $[L, U] \subseteq [l_y(k), u_y(k)]$;

(iii) $\tilde{f} : ([L, U], y(k)) \cap \mathcal{U} \longrightarrow [\bigwedge \tilde{f}(L, y(k)), \bigvee \tilde{f}(U, y(k))] \cap \mathcal{U}$ is onto for any $[L, U] \subseteq [l_y(k), u_y(k)]$;

Proof: The proof proceeds by construction. (0) A lattice $(\chi, \le)$ with $\mathcal{U} \subset \chi$ is constructed; (1) the map
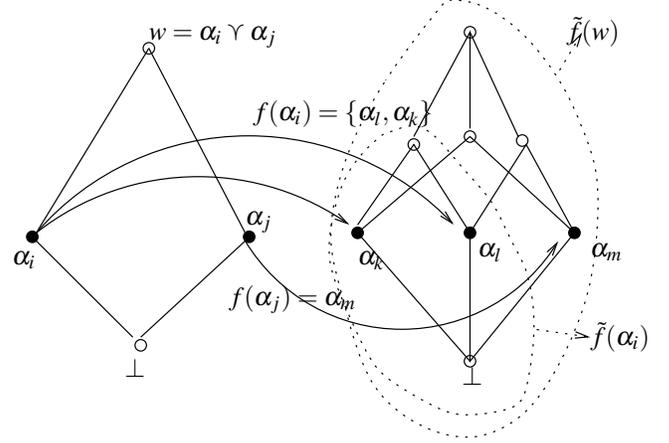

Fig. 1. Extension $\tilde{f}$ on lattice χ.

$h : \mathcal{U} \times \mathcal{Z} \to 2^{\mathcal{Z}}$ is extended to $(\chi, \le)$ such that (i) is verified; (2) the map $f : \mathcal{U} \times \mathcal{Z} \to 2^{\mathcal{U}}$ is extended to $(\chi, \le)$ such that $\tilde{f}|_{\mathcal{U} \times \mathcal{Z}} \cap \mathcal{U} = f$, and such that (ii)-(iii) are verified. Since the constructions (0) and (1) are identical to the deterministic case (see [13]), the proof concentrates on (iii).

(2) In order to prove (ii), $\tilde{f}$ is defined. For simplifying the notation, we omit the dependence on y. For every pair of atomic elements (α_i, α_j), define

$$\begin{aligned}
\tilde{f}(\alpha_i \curlyvee \alpha_j) &:= \tilde{f}(\alpha_i) \curlyvee \tilde{f}(\alpha_j), \\
\tilde{f}(\alpha_i) \curlyvee \tilde{f}(\alpha_j) &:= \mathscr{P}((f(\alpha_i) \cup f(\alpha_j)),
\end{aligned} \quad (2)$$

where $\curlyvee$ is the set union as established in (0). See Figure 1. In analogous way $\tilde{f}(w \curlyvee z)$ is defined for $w, z \in \chi$. Therefore, for any $w \in \chi$, it follows that $\tilde{f}(w) = [\bot, \bigvee \tilde{f}(w)]$. Also define $\tilde{f}(\bot) = \bot$. It follows by construction that $f(\alpha) = \tilde{f}(\alpha) \cap \mathcal{U}$ for any $\alpha \in \mathcal{U}$. To show that $\tilde{f}$ is order preserving (ii), we check Definition 2.4. Since for any $w \le z$ we have $\bigwedge \tilde{f}(w) = \bigwedge \tilde{f}(z) = \bot$, we need to check that $\bigvee \tilde{f}(w) \le \bigvee \tilde{f}(z)$. In fact if $w \le z$, then $w = \alpha_1 \curlyvee ... \curlyvee \alpha_m$ and $z = \alpha_1 \curlyvee ... \curlyvee \alpha_m \curlyvee \alpha_{m+1} \curlyvee ... \curlyvee \alpha_n$ for some $\alpha_i \in \mathcal{U}$ by part (0). By definition $\tilde{f}(w) = \mathscr{P}((f(\alpha_1) \cup ... \cup f(\alpha_m))$ and $\tilde{f}(z) = \mathscr{P}((f(\alpha_1) \cup ... \cup f(\alpha_m) ... \cup f(\alpha_n))$. Therefore $\bigvee \tilde{f}(w) = \cup_{j=1:m} f(\alpha_j) \subset \cup_{j=1:n} f(\alpha_j) = \bigvee \tilde{f}(z)$. (iii) To prove that $\tilde{f} : [\bot, U] \cap \mathcal{U} \longrightarrow [\bot, \bigvee \tilde{f}(U)] \cap \mathcal{U}$ is onto, we need to show that for any $\beta \in [\bot, \bigvee \tilde{f}(U)] \cap \mathcal{U}$ there is $\alpha \in [\bot, U] \cap \mathcal{U}$ such that $\beta \in \tilde{f}(\alpha)$. By construction (part (0)) we have that $U = \alpha_1 \curlyvee ... \curlyvee \alpha_n$, for some $\alpha_1, ..., \alpha_n$. Therefore $[\bot, \bigvee \tilde{f}(U)] \cap \mathcal{U} = \cup_{i=1:n} f(\alpha_i)$, which implies that $\beta \in f(\alpha_i)$ for some $i \in \{1, ..., n\}$. Since $U = \alpha_1 \curlyvee ... \curlyvee \alpha_n$ we have that $\alpha_i \in [\bot, U] \cap \mathcal{U}$ for all i. ∎

Theorem 4.1 states that there is always a lattice $(\chi, \le)$, $\mathcal{U} \subset \chi$, with extensions $\tilde{f}$ and $\tilde{h}$ of f and h on $(\chi, \le)$, such that the conditions that allow to apply the estimator given in Theorem 3.1 are satisfied. However, the resulting lattice $(\chi, \le)$ can be large, and the computation of joins and meets between elements in the lattice can be computational expensive, and may require storage of a number of joins

and meets for on-line implementation of the estimator. This problem can be avoided if an algebraic structure is naturally associated with $\mathcal{U}$, so that join and meet, and $\tilde{f}$ on elements in $(\chi, \leq)$ can be computed exploiting algebraic properties. But if an order structure $(\chi, \leq)$ is fixed, extensions $\tilde{f}$ and $\tilde{h}$ that satisfy the conditions in Theorem 3.1 on such a lattice are not guaranteed to exist. To overcome this problem, we propose to replace f with a function F such that $F(S)$ contains $f(S)$ for any set $S \in \mathcal{U}$ and F has the properties required in Theorem 3.1 on the chosen $(\chi, \leq)$. This is formally explained in the following section.

V. ADDING NONDETERMINISM FOR CONSTRUCTING THE LU ESTIMATOR

In the following result, which is a Corollary of Theorem 3.1 , we show that if the order structure $(\chi, \leq)$ is fixed even if there does not exist extensions of f and h on $(\chi, \leq)$ that have the properties required for the estimator construction, we can find a nondeterministic system (F, H) with the desired properties, whose executions contain the ones of (f, h). However, if (f, h) is deterministic, monotonicity of the error as prescribed by [12] can be lost, and if (f, h) is not deterministic a slower convergence rate can result depending on the choice of (F, H). This is a compromise between complexity and performance.

Corollary 5.1: Consider the system $\Sigma = (f, h, \mathcal{U}, \mathcal{Z})$. Let Σ be weakly observable and fix a lattice $(\chi, \leq)$, with $\mathcal{U} \subset \chi$. If there are nondeterministic maps $F : \mathcal{U} \times \mathcal{Z} \times \mathcal{Z} \to 2^{\mathcal{U}}$ and $H : \mathcal{U} \times \mathcal{Z} \to 2^{\mathcal{Z}}$ such that the system $\Sigma_e = (F, H, \mathcal{U}, \mathcal{Z})$ is such that

(a) $\mathcal{E}(\Sigma) \subset \mathcal{E}(\Sigma_e)$;
(b) each $\sigma_e \in \mathcal{E}(\Sigma_e)$ such that $g(\sigma_e) = g(\sigma)$ for some $\sigma \in \mathcal{E}(\Sigma)$ is weakly equivalent to σ;
(c) Σ_e is such that items 2a, 2b, 2c of Theorem 3.1 are satisfied;

then

$$L(k+1) = \bigwedge \tilde{F}\left((L(k) \curlyvee l_y(k)), y(k), y(k+1)\right)$$
$$U(k+1) = \bigvee \tilde{F}\left((U(k) \curlywedge u_y(k)), y(k), y)k+1)\right) \quad (3)$$

with $L(0) = \bigwedge \chi$, $U(0) = \bigvee \chi$, and $\{y(k)\}_{k \in \mathbb{N}} = g(\sigma)$ with $\sigma \in \mathcal{E}(\Sigma)$, is such that (i) and (ii) of Theorem 3.1 are verified.

Proof: (i) of Theorem 3.1 follows directly from equations (3) and from assumptions (a) and (c). (ii) follows from the fact that Σ_e is weakly observable on $\{y(k)\}_{k \in \mathbb{N}}$ (by (b)), and from the fact that by virtue of (c), Theorem 3.1 can be applied to Σ_e. ∎

Given the system Σ, there are several ways one can construct a system Σ_e that satisfies items (a) and (c) of Corollary 5.1, but one is not guaranteed that also (b) will be satisfied. In the following algorithms, a possible procedure is proposed for constructing a system Σ_e for which (a) and (c) hold on a fixed lattice $(\chi, \leq)$.
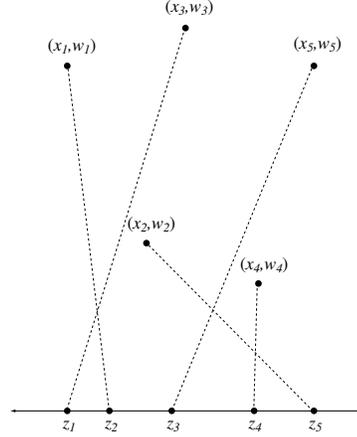
Algorithm 5.1: (procedure for constructing H)

Fig. 2. An example state of the RoboFlag Drill for 5 robots. Here $\alpha = \{3, 1, 5, 4, 2\}$.

- At each k let $a_y(k) = \{\alpha \in \mathcal{U} \mid y(k+1) \in h(\alpha, y(k))\}$. Define $A_y(k) := [\bigwedge a_y(k), \bigvee a_y(k)]$;
- compute $A_y(k) \cap \mathcal{U}$, and define H such that $y(k+1) = H(\alpha_i, y(k))$ if and only if $\alpha_i \in A_y(k) \cap \mathcal{U}$;
- define $y(k+1) \in \tilde{H}(w, y(k))$ if and only if $w \in A_y(k)$.

By this construction, it follows that if $A_y(k) \cap \mathcal{U} = a_y(k)$, then $H(\alpha_i, y(k)) = h(\alpha_i, y(k))$ for any $\alpha_i \in A_y(k) \cap \mathcal{U}$. If instead $A_y(k) \cap \mathcal{U}$ contains more elements than the ones in $a_y(k)$, these are added to the set of α such that $y(k+1) \in h(\alpha, y(k))$. It is also by construction that $\tilde{H}|_{\mathcal{U} \times \mathcal{Z}} = H$. Clearly, property 2a. of Theorem 3.1 is satisfied.

Algorithm 5.2: (procedure for constructing F)

- At each k let $a_y(k) = \{\alpha \in \mathcal{U} \mid y(k+1) \in h(\alpha, y(k))\}$. Define $A_y(k)' := [\bigwedge f(a_y(k), y(k)), \bigvee f(a_y(k), y(k))]$;
- if for any $\alpha' \in A_y(k)' \cap \mathcal{U}$ there is $\alpha \in a_y(k)$ such that $\alpha' \in f(\alpha, y(k))$, and there is a natural extension $\tilde{f}$ that satisfies 2b. of Theorem 3.1 then define $F(\alpha, y(k), y(k+1)) := f(\alpha, y(k))$ for $\alpha \in a_y(k)$, and $\tilde{F} := \tilde{f}$;
- else define $F(\alpha, y(k), y(k+1)) := A_y(k)' \cap \mathcal{U}$ for any $\alpha \in a_y(k)$, and define $\tilde{F}(w, y(k), y(k+1)) := A_y(k)'$ for any $w \in A_y(k)$.

It follows by this construction that $\tilde{F}|_{\mathcal{U} \times \mathcal{Z} \times \mathcal{Z}} \cap \mathcal{U} = F$. Property 2b. of Theorem 3.1 is trivially satisfied in the case $F \neq f$, as any subset $[L, U]$ of $A_y(k)$ is mapped to $A_y(k)'$. Also property 2c. is verified, as for any $\alpha' \in A_y(k)' \cap \mathcal{U}$, there is $\alpha \in a_y(k)$ such that $\alpha' \in F(\alpha, y(k), y(k+1))$. By construction one can verify that (a) of Corollary 5.1 is also satisfied.

In the next section, this procedure is applied to a nondeterministic multi-robot system.

VI. EXAMPLE: THE ROBOFLAG DRILL

In this section, a nondeterministic version of the RoboFlag Drill proposed in [12] is presented. Some number of blue robots with positions $(z_i, 0) \in \mathbb{R}^2$ must defend their zone $\{(x, w) \mid w \leq 0\}$ from an equal number of incoming

red robots. The positions of the red robots are $(x_i, w_i) \in \mathbb{R}^2$. An example for 5 robots is illustrated in Figure 2. The red robots move straight toward the blue defensive zone. The blue robots are assigned each to a red robot and they coordinate to intercept the red robots. Let N represent the number of robots in each team. The robots start with a random assignment $\alpha : \{1,...,N\}$ into $\{1,...,N\}$. At each step, each blue robot communicates with its neighbors and decides to either switch assignments with its left or right neighbor or keep its assignment.

The system can be described with a guarded command program that is a way of specifying transition functions. Such programs are constituted by a set of clauses. Each clause is of the form *guard : rule*. When a guard becomes true the corresponding rule is executed (for more details see [8]). The red robot dynamics is described by the N clauses $w_i - \delta > 0 : w_i' = w_i - \delta$ for $i \in \{1,...,N\}$. The blue robot dynamics Σ are described by the $3N$ clauses

$$z_i < x_{\alpha_i} \quad : \quad z_i' = z_i + \delta, \tag{4}$$
$$z_i > x_{\alpha_i} \quad : \quad z_i' = z_i - \delta, \tag{5}$$
$$switch_{(i,i+1)} \quad : \quad (\alpha_i', \alpha_{i+1}') = (\alpha_{i+1}, \alpha_i), \tag{6}$$

for $i \in \{1,...,N\}$, where $switch_{(i,j)}$ is such that

$$switch_{(i,i+1)} \Rightarrow x_{\alpha_i} \geq x_{\alpha_{i+1}} \tag{7}$$
$$switch_{(i,i+1)} \wedge switch_{(j,j+1)} = false, \; i \neq j \tag{8}$$
$$\big((x_{\alpha_1} \geq x_{\alpha_2}) \vee .... \vee (x_{\alpha_{N-1}} \geq x_{\alpha_N})\big) \Rightarrow$$
$$(switch_{(1,2)} \vee ... \vee switch_{(N-1,N)} = true. \tag{9}$$

Primed variables denote the value of a variable after it has been updated. Guards (6) establish that two close robots will exchange their assignments if *switch* is true. In particular, (7), (8), (9), along with (4),(5),(6), guarantee that, if there are some close robots with conflicting assignments, there is one and only one pair of robots among them that will switch the assignments. This renders the assignment protocol in commands (6) nondeterministic, as at each step we do not know which pair of robots switches assignments. It is possible to show (see [9] for details) that the assignment protocol converges to the equilibrium value $(1,...,N)$. For the blue robots we assume that $z_i < z_{i+1}$, and that $x_i \leq z_i \leq x_{i+1}$ for all time. With this assumption, it is possible to show that system Σ is weakly observable (see [11]). We define $x = (x_1,...,x_N)$, $z = (z_1,...,z_N)$, $\alpha = (\alpha_1,...,\alpha_N)$.

The commands reported in (6) determine the function $f : \mathcal{U} \times \mathcal{Z} \to 2^{\mathcal{U}}$ that updates the discrete variables α, with $\mathcal{U} = \text{perm}(N)$, while the commands in (5) and (4) determine the function $h : \mathcal{U} \times \mathcal{Z} \to \mathcal{Z}$, with $\mathcal{Z} = \mathbb{R}^N$. Therefore the blue robot system is defined by $\Sigma = (f, h, \text{perm}(N), \mathbb{R}^N)$.

State estimation problem. We consider the problem of estimating the current assignment $\alpha(k)$, given the motions of the blue robots $z(k)$ and of the red robots $(x(k), w(k))$, and by knowing the dynamics of the system Σ.

In view of constructing the LU estimator, we choose as lattice $(\chi, \leq)$ the same as the one used in [12]. Briefly,

$(\chi, \leq)$ is the set of vectors in $\mathbb{N}^N$ with entries in $[1, N]$, and the order is established componentwise First, we show in the following claim that on such a lattice there are no extensions of f that satisfy the requirements for the construction of the LU estimator as given in Theorem 3.1.

Claim 6.1: Consider the system $\Sigma = (f, h, \text{perm}(N), \mathbb{R}^N)$ represented in equations (4-5-6). Then there does not exist any extension $\tilde{f}$ of f on $(\chi, \leq)$ that satisfy 2b. and 2c. of Theorem 3.1.

Proof: (Sketch.) This is shown for $N = 4$ for simplicity, the same can be proved for any N. Assume $\alpha(k) = (4,3,2,1)$. This implies, from commands (4)-(5) read from right to left, that the set of all α compatible with $y(k), y(k+1)$ is given by $A_y \cap \mathcal{U}$, where $A_y = [l_y, u_y] = [(2,3,1,1),(4,4,3,4)]$. By computing $f([l_y, u_y] \cap \mathcal{U})$ one note that if $f([l_y, u_y] \cap \mathcal{U}) \subseteq [\bigwedge \tilde{f}(l_y), \bigvee \tilde{f}(u_y)]$, it must be $\bigwedge \tilde{f}(l_y) \leq (2,1,1,1)$ and $\bigvee \tilde{f}(u_y) \geq (4,4,4,4)$. This in turn implies that $(2,3,1,4) \in [\bigwedge \tilde{f}(l_y), \bigvee \tilde{f}(u_y)] \cap \mathcal{U}$, but it is not in $\tilde{f}([l_y, u_y] \cap \mathcal{U}) \cap \mathcal{U}$. As a consequence $\tilde{f} : [l_y, u_y] \cap \mathcal{U} \to [\bigwedge \tilde{f}(l_y), \bigvee \tilde{f}(u_y)] \cap \mathcal{U}$ cannot be onto. ∎

A. LU Estimator Construction

Corollary 5.1 is used along with the Algorithm 5.2, as 2a. of Theorem 3.1 is satisfied with $H = h$.

From commands (4)-(5)-(6), we deduce that at each step a switch of the assignment α can be either observable if it leads to a change in the velocity or not observable otherwise. Let $v(k) = z(k+1) - z(k)$ denote the velocity and omit the dependence of F on y. Define

$$F(\alpha) := f(\alpha) \text{ if } v(k) \neq v(k-1) \tag{10}$$
$$F(\alpha) := [l_y(k), u_y(k)] \cap \mathcal{U} \text{ otherwise.} \tag{11}$$

One can check that (a) and (b) of Corollary 5.1 are satisfied. An extension $\tilde{F}$ that satisfies (c) of Corollary 5.1 is

$$\tilde{F}(x) = w, \; (w_i, w_{i+1}) := (x_{i+1}, x_i), \text{ if } v_i(k) \neq v_i(k-1) \tag{12}$$
$$\tilde{F}(x) := [l_y(k), u_y(k)] \text{ otherwise.} \tag{13}$$

B. Simulation results

The performance of the LU estimator of Corollary 5.1 is reported in Figure 3. In the figure, $W(k) = \frac{1}{N} \sum_{i=1}^{N} |m_i(k)|$, where $|m_i(k)|$ is the cardinality of the sets $m_i(k)$ that are the sets of possible α_i for each component obtained from the sets $[L_i, U_i]$ by removing iteratively a singleton occurring at component i by all other components. When $[L(k), U(k)] \cap \text{perm}(N)$ has converged to α, then $m_i(k) = \alpha_i(k)$. This function represents the estimation error at each step. The function $E(k) = \frac{1}{N} \sum_{i=1}^{N} |\alpha_i(k) - i|$, which gives an idea of the speed of convergence of the assignment to the equilibrium value $(1,...,N)$. $|[L(k), U(k)] \cap \mathcal{U}|$ converges to 1, as predicted by Corollary 5.1, but $|[L(k), U(k)]|$ is not a monotonic function of k as it was in the deterministic case. This is due to the nondeterministic nature of the transition functions. The choice of F has a considerable impact on the convergence speed of the estimator, and the procedure given in the previous section not necessarily gives

the best choice. The map F we chose is rough and does not take other information that we have on the system into account. The most information F can model, the fastest is the convergence rate.
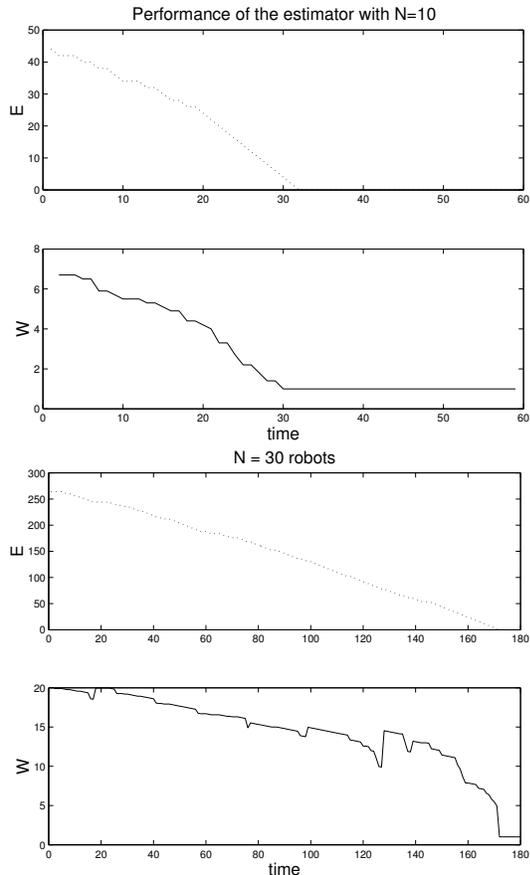

Fig. 3. Example with N=10(up) and N=30(down). Dashed plots show the stabilization function of the α assignment (E). Solid plots show the function W for the estimator.

A way of measuring the complexity of the LU estimator is to count the number of updated variables. The estimator roughly updates the $2N$ variables L_i and U_i. System Σ updates $2N$ variables as well, that is α_i and z_i. Therefore, it can be informally said that the system Σ and its LU estimator have the same complexity.

VII. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

The problem of estimating the discrete variables in a class of nondeterministic hybrid systems, where the continuous variables are available for measurement has been presented. Using partial order theory, a discrete state estimator on a lattice was constructed, which updates the lower and upper bounds of the set of discrete states compatible with the output sequence. For weakly observable systems, there always exist a lattice where the proposed estimator can be constructed. This shows that the lattice approach to

estimation is general. However, some partial order structures are to be preferred to others for complexity reasons. We showed that it is possible to add nondeterminism to a system to obtain a new system that admits the construction of the estimator on the chosen lattice. The drawback is that the LU estimator on the chosen lattice can have a slower convergence rate than a LU estimator constructed on a more complicated lattice. This is a compromise between complexity and performance. Our ideas were applied to a multi-robot example, whose discrete state set is so large as to render previously proposed methods inapplicable.

B. Future Work

The results obtained in this study are still preliminary and point to a large number of future research directions. In particular, the existence of an observable extension Σ_e is to be investigated. A major research trust will be to consider the case in which the continuous variables need to be estimated as well.

VIII. ACKNOWLEDGMENTS

REFERENCES

[1] A. Balluchi, L. Benvenuti, M. D. Di Benedetto, and A. Sangiovanni-Vincentelli. Design of observers for hybrid systems. In *Lecture Notes in Computer Science 2289,C. J. Tomlin and M. R. Greensreet Eds. Springer*, pages 76–89, 2002.

[2] A. Balluchi, L. Benvenuti, M. D. Di Benedetto, and A. Sangiovanni-Vincentelli. Observability of hybrid systems. In *Proc. of 42nd IEEE Conference on Decision and Control*, pages 1159–1164, 2003.

[3] A. Bemporad, G. Ferrari-Trecate, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Transactions on Automatic Control*, 45:1864–1876, 1999.

[4] P. E. Caines. Classical and logic-based dynamic observers for finite automata. *IMA J. of Mathematical Control and Information*, pages 45–80, 1991.

[5] A. R. Cassandra, L. P. Kaelbling, and M. L Littman. Acting optimally in partially observable stochastic domains. In *Proc. 12th Conference on Artificial Intelligence*, pages 1023–1028, Seattle, WA, 1994.

[6] B. A. Davey and H. A. Priesteley. *Introduction to Lattices and Order*. Cambridge University Press, 2002.

[7] C. M. Özveren and A. S. Willsky. Observability of discrete event dynamic systems. *IEEE Transactions on Automatic Control*, 35(7):797–806, 19.

[8] E. W. Dijkstra. Guarded commands, non-determinacy and a calculus for the derivation of programs. In *Proceedings of the international conference on Reliable software*, pages 2 – 2.13, Los Angeles, California, 1975. http://portal.acm.org.

[9] E. Klavins. A formal model of a multi-robot control and communication task. In *Conference on Decision and Control*, Hawaii, 2003.

[10] P. J. Ramadge. Observability of discrete event systems. In *Proc. 25th Conference on Decision and Control*, pages 1108–1112, Athens, Greece, 1986.

[11] D. Del Vecchio and E. Klavins. Observation of guarded command programs. In *Conference on Decision and Control*, Hawaii, 2003.

[12] D. Del Vecchio and R. M. Murray. Discrete state estimators for a class of hybrid systems on a lattice. In *7th International Workshop on Hybrid Systems: Computation and Control*, University of Pennsylvania, Philadelphia, USA, 2004.

[13] D. Del Vecchio and R. M. Murray. Observability of a class of hybrid systems on a lattice. In *43rd Conference on Decision and Control (submitted)*, 2004.

[14] R. Vidal, A. Chiuso, and S. Soatto. Observability and identifiability of jump linear systems. In *Decision and Control Conference*, Las Vegas, 2002.