Intersection Based Decentralized Diagnosis: Implementation and Verification

Maria Panteli and Christoforos N. Hadjicostis

Abstract—We consider decentralized diagnosis in discrete event systems that are modeled as non-deterministic finite automata and are observed, through distinct natural projection maps, at multiple observation sites. Specifically, we consider a scenario where two or more observers (each with its own map) are allowed to communicate their assessments (state estimates and matching normal/failure conditions) to a centralized location which then makes an overall decision based on the intersection of the local state estimates and their matching conditions. This intersection based decentralized diagnosis (IBDD) scheme can be implemented with polynomial complexity, both at the observation sites and at the centralized location. It is shown that IBDD can be verified with an algorithm of polynomial complexity that constructs verifiers for the observation sites, and analyzes properties of their parallel composition and product with the system.

I. INTRODUCTION

Failure diagnosis is an important aspect of monitoring and control of discrete event systems, including transportation systems, medical diagnostics, heating/ventilation systems, and communication networks. In a centralized (monolithic) setting, detection and isolation of failures in a given (non-deterministic) finite automaton that models the system of interest is performed by a single entity, called observer or diagnoser, typically designed as a (deterministic) finite automaton that is driven by observable events [1]–[3]. Following [3], many researchers have investigated algorithms for fault diagnosis and the verification of diagnosability. In particular, language diagnosability as defined in [3] can be verified with polynomial-time algorithms [4], [5]. The approach in [4] assumes a deterministic system and constructs, for each failure type, a verifier that can be used to check fault detection and isolation of the corresponding failure type. The system is considered diagnosable for a particular failure type if all cycles of the corresponding verifier have identical state labels. The two algorithms mentioned above have the same computational complexity in terms of the number of events of the system, whereas the algorithm in [4] is of lower complexity in terms of the number of states (at the cost of dealing with deterministic automata and a single failure type at a time). Related approaches, that overcome the so-called state-explosion problem, also include the work in [6], which assumes asynchronous diagnosis using Petri-nets and [7], which studies failure diagnosis of discrete event systems with linear-time temporal logic specifications.

Diagnosability has also been investigated in decentralized architectures. The authors in [8] propose three protocols for coordinated decentralized diagnosis. Protocols 1 and 2 assume unidirectional communication from the local diagnosers to a coordinator, whereas Protocol 3 assumes no communication between them or to any coordinator. Polynomial complexity algorithms for the verification of Protocol 3 and variations of it can be found in [9]–[13]. In this paper we describe and propose a verification algorithm of polynomial complexity for a protocol that resembles Protocol 2 in [8]. Fault diagnosis under this protocol is performed locally using the (basic) diagnoser, and upon the request of the coordinator, diagnostic information is communicated to the coordinator who reaches an overall diagnosis decision by taking the intersection of the local estimates (state estimates along with matching normal/failure conditions).

This paper proposes a polynomial time algorithm for the verification of diagnosability using intersection based decentralized diagnosis (IBDD), inspired by the polynomial-time verification approach for centralized diagnosability in [5], with adaptations to the decentralized architecture and the intersection operation. Local estimates are combined with the use of the parallel composition of local verifiers and the product with the system model. Using this composition automaton, one can obtain necessary and sufficient conditions for IBDD diagnosability, as described later in the paper. The verification of these properties has complexity polynomial in the number of states of the finite automaton and exponential in the number of observation sites.

II. PRELIMINARIES

A. Model

The system under diagnosis is modeled as a non-deterministic finite automaton, which is a particularly useful model in planning systems, such as robotics, navigation and control [14], [15]. We also include notation for deterministic systems because it makes connections with earlier work clearer. The system is defined as $G = (Q, \Sigma, \delta, q_0)$ where
$Q$ denotes the finite set of states, $\Sigma$ denotes the finite set of events, $q_0 \in Q$ (or $\{q_0\} \subseteq Q$) is the initial state and $\delta: Q \times \Sigma \rightarrow 2^Q$ is the transition function (with $2^Q$ denoting the power set of $Q$). System $G$ is deterministic if $|\delta(\cdot, \cdot)| \leq 1$ (where $|S|$ denotes the cardinality of set $S$). An event $\sigma \in \Sigma$ causes a transition from the current state $q \in Q$ to the state $q' \in Q$ if $q' \in \delta(q, \sigma)$. The language generated by $G$, denoted by $L(G)$ or sometimes by $L$, is a subset of $\Sigma^*$, where $\Sigma^*$ denotes the Kleene closure of the set $\Sigma$. The event set $S$ is partitioned as $\Sigma = \Sigma_1 \cup \Sigma_2$, where $\Sigma_1$ is the set of observable events and $\Sigma_2$ is the set of unobservable events.

The event observation can be conveniently described by the so-called natural projection with respect to the set of observations $\omega$. $\Sigma$ is the set of failure events, following the sequence of observations $\omega$. For failure diagnosis, we consider some specific events that indicate faults or abnormal conditions and need to be detected (more generally, they need to be classified or identified) after a finite (bounded) delay. Let $\Sigma_f \subseteq \Sigma$ denote the set of failure events and assume without loss of generality that $\Sigma_f \subseteq \Sigma_{uo}$ (since an observable failure event is immediately diagnosable). The set of failure events is partitioned into disjoint sets of different failure types $\Sigma_f = S_{F_1} \cup \ldots \cup S_{F_k}$. This partition is denoted by $\Pi_f$. We also denote the set of traces $s \in L$ that end with a failure event of type $F_i$ as $\Psi(S_{F_i}) = \{s \sigma_{F_i} \in L: s \in (\Sigma \setminus \Sigma_f)^* \sigma_{F_i}, \in S_{F_i}\}$ and the postlanguage of $L$ after $s$ as $L \setminus s = \{t \in \Sigma^*: st \in L\}$. 

B. Centralized Diagnosis

System diagnosis in a centralized (monolithic) architecture aims to detect or precisely identify the occurrence, if any, of failure events, following the sequence of observations generated by the system. The following assumptions on the language $L(G)$ are typical [3] and are also assumed in our development here:

(A1) $G$ has no cycles of unobservable events;

(A2) $L(G)$ is live i.e., a transition is defined at each state $q \in Q$.

Diagnosability as introduced in [3] can be tested with the diagnoser $G_d = (Q_d, \Sigma_d, \delta_d, q_{0_d})$. This is a deterministic finite automaton built from $G$ that provides an estimate of the current state of system $G$ and its matching failure condition(s) (if any). The latter is given in terms of a subset of the set $\Delta_f = \{F_1, F_2, \ldots, F_k\}$, that are assigned to each state estimate, and correspond to faulty behavior associated with a fault in fault class $F_i$, and the label $N$, that corresponds to normal behavior of the system.

The state space $Q_d$ is a subset of $2^{Q \times \Delta}$, where $Q$ is the state space of the system and $\Delta$ is the complete set of possible labels $\Delta = \{N\} \cup 2^\Delta$. More specifically a state $q_d \in Q_d$ is of the form $q_d \subseteq Q \times \Delta$, i.e., $q_d$ can be written as

$$q_d \subseteq \{(q_1, l_1), \ldots, (q_n, l_n)\},$$

with $q_i \in Q$ and $l_i \in \Delta$ (repetitions of $q_i$ are permitted). The transition function $\delta_d: 2^{Q \times \Delta} \times \Sigma_2 \rightarrow 2^{Q \times \Delta}$ satisfies $q'_d = \delta_d(q_d, \sigma)$ and updates both the possible states and matching normal/failure condition(s) as defined in [3]. Verification of diagnosability using the diagnoser depends on the presence of $F_i$-uncertain states which are defined as follows [8].

Definition 1: A state $q_d \in Q_d$ of the diagnoser is called $F_i$-uncertain if $\exists (x, l), (y, l') \in q_d$, such that $x, y \in Q, F_i \in l$ and $F_i \not\in l'$. If the state estimate of the diagnoser contains only $N$ labels, the system is diagnosed to normal behavior and if all the labels correspond to $F_i$ the system is diagnosed to faulty behavior associated with fault class $F_i$. Otherwise, the estimate corresponds to an $F_i$-uncertain state such that faulty behavior of type $F_i$ cannot be isolated and the fault cannot be diagnosed (at least not at this point, but it might be possible that the failure can be diagnosed once more observations become available). In general, system $G$ is assumed diagnosable if its diagnoser contains no cycles composed exclusively of $F_i$-uncertain states for some failure type $F_i$. 

C. Decentralized Diagnosis

In decentralized architectures, there are multiple observation sites (each with its own natural projection map), and diagnosis needs to be performed based on the information available at each observation site, and also on the type of protocol that is used to exchange information between sites. For codiagnosability, where no communication to a coordinator is assumed, the local decisions need to be considered individually. Codiagnosability requires that for any arbitrarily long trace following a fault, there exists at least one local site that, based on its own (local) information, can detect the fault with certainty within a bounded interval of observations [13].

Example 1: Consider the system shown in Fig.1a. Assume $\Sigma_f = \{f\}$, and let $\Sigma_{o_1} = \{a, c\}, \Sigma_{o_2} = \{b, c\}$ be the sets of observable events for diagnosers $D_1$ and $D_2$ respectively. Diagnoser $D_1$ is shown in Fig.1b and diagnoser $D_2$ is shown in Fig.1c. The faulty trace $abbec$ under projection $P_{\Sigma_{o_1}}$ leads diagnoser $D_1$ to the cycle consisting of the (single) $F_i$-uncertain state $(3F, 4N)$. The same trace under projection $P_{\Sigma_{o_2}}$ leads diagnoser $D_2$ to the cycle consisting of the (single) $F_i$-uncertain state $(3F, 5N)$. Thus, neither of the local sites can detect with certainty the occurrence of the fault within a finite number of observations. Hence, the condition of codiagnosability is violated.

---

1With some abuse of notation, the initial state is defined for simplicity as a singleton subset of $Q$ but this can be easily relaxed to be an arbitrary subset of $Q$.

2For example, state $q_d$ could be the set $\{(q_1, N), (q_1, F_1), (q_2, F_2)\}$.

3It is possible that cycles with $F_i$-uncertain states are not executable after the occurrence of faults; since such cycles do not violate diagnosability we need to check against this possibility [3].
If communication to a coordinator is allowed, possible combinations of local estimates can be considered. These can be represented by the intersection of the estimates of the local diagnosers [8], [16]. Decentralized diagnosability (under this intersection operation) can be verified by searching for cycles with $F_i$-uncertain states within an automaton composed from the local diagnosers and the given system. Specifically, the authors in [16] proposed the offline pre-computation of possible intersections of local estimates with the use of a look-up table. In the case of two observation sites with natural projection maps $P_{\Sigma_{o1}}$ and $P_{\Sigma_{o2}}$, [8], [16] determines multiple-request diagnosability by constructing $D_c \times (D_1 \parallel \parallel D_2)$, where $D_c$ is the centralized diagnoser (based on $\Sigma_o = \Sigma_{o1} \cup \Sigma_{o2}$) and $D_i$, $i = 1,2$, is the local diagnoser based on $\Sigma_{oi}$. If the product automaton $D_c \times (D_1 \parallel \parallel D_2)$ contains no cycles of states corresponding to ambiguous entries of the look-up table, the system is considered multiple-request diagnosable [16].

Example 2: For the system $G$ and diagnosers $D_1$ and $D_2$ in Fig.1, the finite automaton $D_c \times (D_1 \parallel \parallel D_2)$ as defined in [16] is shown in Fig.2, where states are represented by triplets $\{q_{D_c}, q_{D_1}, q_{D_2}\}$. This FSM contains, amongst others, the cycle consisting of local estimates $(3F, 4N)$ of $D_1$ and $(3F, 5N)$ of $D_2$. The intersection of these estimates gives $\{3F\}$ which is not an $F_i$-uncertain state. Therefore, under this scheme, the system is considered multiple-request diagnosable.

III. PROBLEM FORMULATION

We consider a system $G = (Q, \Sigma, \delta, q_0)$ (modeled as a non-deterministic finite automaton) that is observed by multiple observers, each with its own natural projection map with respect to its set of observable events. More specifically, we assume $m$ observers, each characterized by the set of (locally) observable events $\Sigma_{o1}, \Sigma_{o2}, \ldots, \Sigma_{om}$. The set of observable events of the system is defined as the union of the local sets of observable events, i.e., $\Sigma_o = \Sigma_{o1} \cup \cdots \cup \Sigma_{om}$. Similarly, we define $\Sigma_{uo1} = \Sigma \setminus \Sigma_{o1}$ for $i = 1,2,\ldots, m$ and $\Sigma_{uo} = \Sigma \setminus \Sigma_o$. We assume that $\Sigma_f \subseteq \Sigma_{uo}$ (without loss of generality, since otherwise, at least one observation site will be able to immediately identify the fault). For fault diagnosis, a local diagnoser $D_j$ is assigned to each observer and performs diagnosis based on the events it observes under natural projection $P_{\Sigma_{o1}}$. Note that we refer to a local diagnoser but, in practice, online diagnosis can be performed more efficiently via an iterative algorithm that keeps track of the possible states and matching normal/failure condition(s), following any particular sequence of observations (without explicitly constructing the overall diagnoser, but only the parts of it that are needed following the sequence of observations). As in [16], we assume a coordinator that makes a global diagnosis decision by requesting (perhaps at multiple instances) information from the local diagnosers and processing it accordingly. While performing diagnosis, we assume that the following hold.

(A3) There is reliable communication between the local sites and the coordinator, i.e., information is communicated to the coordinator correctly and in order.

(A4) Local sites are aware of the system model and can recursively estimate the possible states and matching normal/failure condition(s); however, the coordinator has limited memory and capacity (and is not necessarily aware of the system model).

We investigate decentralized diagnosis under the intersection operation. In particular, our proposed IBDD scheme has the following specifications.

SP1. At any given time instant, following a sequence of events $s \in \Sigma^*$ that generates the sequence of observations $\omega_j = P_{\Sigma_{o1}}(s)$ at the $j^{th}$ observation site, the diagnostic information at local site $j$ consists of all possible states that can be reached along with their matching label $l \in \Delta$. Each local site sends its diagnostic information when requested by the coordinator.

SP2. The coordinator requests information from local sites periodically and takes the intersection of the local state/conditions to form an estimate of the state of the system and matching normal/failure condition(s), in order to reach
a diagnosis decision.

Informally, the diagnosability of this protocol captures the ability of the coordinator to detect (identify) faults following a finite number of observations after their occurrence. The formal definition of diagnosability using IBDD is as follows.

**Definition 2:** A non-deterministic finite automaton $G$ observed by $m$ observers with natural projection maps $P_{\Sigma_1}, \ldots, P_{\Sigma_m}$, under $\text{SP1-SP2}$, is diagnosable using IBDD with respect to fault class $F_i$ if the following condition is true: for any arbitrarily long trace following the occurrence of a fault in class $F_i$, the intersection of the local estimates after a finite number of observations becomes $F_i$-certain, i.e.,

\[ \exists n_0 \in \mathbb{N}, \forall \sigma, t \in L(G), s \in (\Sigma \setminus \Sigma^*)^*, \sigma \in \Sigma_{F_i}, t > n_0, \]

\[ \bigcap_{j=1}^{m} \delta_{d_j}(q_{j0}, P_{\Sigma_{o_j}}(\sigma, t)) \neq \{(q, l), (q', V)\} \]

for some $q, q' \in Q$, with $F_i \in I$ but $F_i \notin q'$.

where $d_{j+1}$ is the next state transition function for the diagnosis $D_j$ at the $j^{th}$ observation site.

**Remark 1:** In the above definition, the intersection of local estimates is defined for states $q^V_j \equiv \delta_{d_j}(q_{j0}, P_{\Sigma_{o_j}}(\sigma, t))$, $j = 1, 2, \ldots, m$, in the standard way

\[ q^V_j \cap q^V_{j+1} \cap \ldots \cap q^V_m = \{(q_j, l_j) \mid (q_j, l_j) \in q^V_j, j = 1, 2, \ldots, m\}. \]

For example, for states $q^V_1 = (3F, 4N)$ and $q^V_2 = (3F, 5N)$, the intersection of local estimates $(3F, 4N) \cap (3F, 5N)$ gives the singleton state $(3F)$. For states $q^V_1 = (2F, 4N)$ and $q^V_2 = (4N, 2F)$, the intersection of local estimates gives the non-singleton state $(2F, 4N) \Box$

Online operation for IBDD is actually rather straightforward: each local site recursively maintains a set of pairs of possible states and fault conditions (maintaining this list requires polynomial complexity in the number of states and exponential complexity in the number of fault classes — because the total number of elements maintained is, at most $|Q| \times 2^{V_r}$), i.e., all the elements of $Q \times \Delta$. When prompted by the coordinator, the local site communicates its state estimates (along with their associated failure labels) to the coordinator, who then computes the intersection of these state estimates. Note that IBDD is different from the notion of codiagnosability. More specifically, codiagnosability implies IBDD but not the other way around.

**IV. VERIFICATION OF IBDD**

**A. Verifier Automaton for each Observation Site**

For each observation site $j$, we construct a verifier which is a non-deterministic automaton that can track the pairs of current system state and matching normal/failure condition(s) that are consistent with the sequence of observations at the $j^{th}$ site.

**Definition 3:** Consider a system $G = (Q^G, \Sigma, \delta^G, q^G_0)$ Let $\Sigma_f \subseteq \Sigma$ denote the set of failure events, assume $\Sigma_f = \Sigma_{F_1} \cup \ldots \cup \Sigma_{F_k}$ is the partition into different failure types, and denote by $\Delta = \{N \cup 2^{\Sigma_f}\}$ the complete set of possible labels where $\Delta_f = \{F_1, F_2, \ldots, F_k\}$. Consider the natural projection map $P_{\Sigma_{o_j}}$ for a set of events $\Sigma_{o_j} \subseteq \Sigma$.

The Verifier automaton $V^j$ is a non-deterministic automaton constructed from the given non-deterministic system $G$ as

\[ V^j = (Q^V_j, \Sigma_{o_j}, \delta^V_j, q^V_0) \]

where

1. $Q^V_j = Q^G \times \Delta \times \Delta^G \times \Delta$ is the set of states.
2. The transition rule $\delta^V_j$ for the verifier automaton is (for $(q_i, l_i, q'_i, l'_i) \in Q^V_j$ and $\sigma \in \Sigma_{f_i}$)

\[ \delta^V_j((q_i, l_i, q'_i, l'_i), \sigma) = \{(q_{i+1}, l_{i+1}, q'_{i+1}, l'_{i+1}) \mid \exists s_1, s_2 \in \Sigma^*, P_{\Sigma_{o_j}}(s_1) = P_{\Sigma_{o_j}}(s_2) = \sigma, q_{i+1} \in \delta^G_i(q_i, s_1), q'_{i+1} \in \delta^G_i(q'_i, s_2), l_{i+1} = f(l_i, s_1), l'_{i+1} = f(l'_i, s_2)\}, \]

where, for $l_i$ and $s_i$, $h = 1, 2$, the label function $f: \Delta \times \Sigma^* \to \Delta$ is defined as

\[ f(l_i, s_i) = \begin{cases} l_i \cup L(s_h), & l_i \neq N, \\ L(s_h), & l_i = N \text{ and } \exists f \in \Sigma_f, f \in s_h, \\ N, & l_i = N \text{ and } \forall f \in \Sigma_f, f \notin s_h, \end{cases} \]

with

\[ L(s_h) = \{ F_i \mid \exists \sigma_{f_i} \in \Sigma_{F_i}, \text{s.t. } f \in s_h \} \]  (1)

3. The initial state of $V^j$ is given by

\[ q^V_0 = \{(q_{00}, l_0, q'_0, l'_0) \mid \exists s_0, s_0 \in (\Sigma \setminus \Sigma_{\sigma_j})^* \text{ s.t.} q_0 \in \delta^G_i(q_{00}, s_0), q'_0 \in \delta^G_i(q'_0, s_0), l_0 = f(s_0), l'_0 = f(s_0)\}, \]

where $f(s_0)$ for $h = 1, 2$ is given by

\[ f(s_{h0}) = \begin{cases} N, & \text{if } \forall \sigma \in \Sigma_f, f \notin s_0, \\ \{ F_i \mid \exists \sigma_{f_i} \in \Sigma_{F_i}, \text{s.t. } f \in s_0 \}, & \text{otherwise.} \end{cases} \]

By construction, $\delta^V_j$ is defined whenever $\delta^G$ is defined. Thus, it tracks two strings in $L(G)$ that are equivalent under the projection $P_{\Sigma_{o_j}}$, i.e., they produce the same sequence of observations, but might have different failure information as the observation sequence evolves. The labels associated to each state of the verifier track the occurrence of faults from different classes, but neglect the order in which they appear as well as the number of occurrences of faults in each class.

**Definition 4:** A subset of states $\{v^j_1, v^j_2, \ldots, v^j_n\} \subseteq Q^V_j$ forms a path denoted by $(v^j_1, v^j_2, \ldots, v^j_n)$ if there exist observations $\sigma_1, \sigma_2, \ldots, \sigma_{n-1} \in \Sigma_{f_j}$ such that $v^j_1 \stackrel{\sigma_1}{\rightarrow} v^j_2 \stackrel{\sigma_2}{\rightarrow} \ldots \stackrel{\sigma_n}{\rightarrow} v^j_n$.

**Definition 5:** A path $(v^j_1, v^j_2, \ldots, v^j_n)$ forms a cycle if $v^j_1 = v^j_n$. A cycle $(v^j_1, v^j_2, \ldots, v^j_n)$ is $F_i$-confused if for all $v^j_k = (q^j_1, l^j_1, q^j_2, l^j_2)$, $k = 1, \ldots, n$, $F_i \in l^j_k$ but $F_i \notin l^j_n$, or vice versa.

Verification of diagnosability (in a centralized architecture) is captured in the following lemma.

**Lemma 1:** System $G$ is diagnosable under observable events $\Sigma_{o_j}$ and partition $H$ on $\Sigma_f$ iff the verifier automaton $V^j$ has no $F_i$-confused cycles.

**Proof:** Lemma 1 addresses centralized diagnosability; its proof is similar to Theorem 1 of [5] and omitted.

**Example 3:** Consider again the system and diagnosers shown in Fig.1. For observable sets $\Sigma_{o_1} = \{a, c\}$ and $\Sigma_{o_2} = \{b, c\}$, we construct respectively the verifiers $V^1$ and
V\(^2\) of system G as shown in Fig.3. Consider again the trace \(abca^*\); we observe that \(V^1\) reaches (amongst others) the cycles \(\langle 3F,4N \rangle\) and \(\langle 4N,3F \rangle\), and \(V^2\) the cycles \(\langle 3F,5N \rangle\) and \(\langle 5N,3F \rangle\). These are \(F_i\)-confused cycles and, by Lemma 1, we conclude that the system is not diagnosable, neither w.r.t. \(\Sigma_{o_1}\) at observation site 1 nor w.r.t. \(\Sigma_{o_2}\) at observation site 2.

For IBDD, the local estimates need to be combined to make a global diagnosis decision. It turns out, that via an appropriate composition of the local verifiers in Definition 3, IBDD can be verified with polynomial complexity. The algorithm proposed for this is presented in the following section, followed by its proof of correctness.

### B. Algorithm for Verification of IBDD

The algorithm proposed for verifying IBDD first constructs the parallel composition of local verifiers \(V^j\). If we focus for simplicity on two verifiers, we have

\[
V^1 \parallel V^2 = (Q^{V^1||V^2}, \Sigma^{V^1||V^2}, \delta^{V^1||V^2}, q_0^{V^1||V^2}),
\]

where

1. \(Q^{V^1||V^2} = Q^{V^1} \times Q^{V^2}\);
2. \(\Sigma^{V^1||V^2} = \Sigma_{o_1} \cup \Sigma_{o_2}(\equiv \Sigma_o)\);
3. \(\delta((q^{V^1}, q^{V^2}), \sigma)\) defined for all \(q^{V^1} \in Q^{V^1}, q^{V^2} \in Q^{V^2}, \sigma \in \Sigma_{o_1} \cup \Sigma_{o_2}\) by
   
   \[
   \begin{cases}
   \delta^{V^1}(q^{V^1}, \sigma) \times \delta^{V^2}(q^{V^2}, \sigma), & \text{for } \sigma \in \Sigma_{o_1} \cap \Sigma_{o_2}, \\
   \delta^{V^1}(q^{V^1}, \sigma) \times q^{V^2}, & \text{for } \sigma \in \Sigma_{o_1} \setminus \Sigma_{o_2}, \\
   q^{V^1} \times \delta^{V^2}(q^{V^2}, \sigma), & \text{for } \sigma \in \Sigma_{o_2} \setminus \Sigma_{o_1}, \\
   \text{undefined,} & \text{otherwise};
   \end{cases}
   \]

4. \(q_0^{V^1||V^2} = q_0^{V^1} \times q_0^{V^2}\).

Following the execution of a string \(s \in L(G)\), the states reached in the parallel composition of verifiers provide pairs of compatible states reached by the local verifiers following the projection of \(s\) under the local set of observable events, i.e., for \(s \in L(G)\),

\[
(q^{V^1}, q^{V^2}) \in (\delta^{V^1||V^2}((q_0^{V^1}, q_0^{V^2}), P_{\Sigma_{o_1} \cup \Sigma_{o_2}}(s)))
\]

if and only if

\[
q^{V^1} \in \delta^{V^1}(q_0^{V^1}, P_{\Sigma_{o_1}}(s)), \quad q^{V^2} \in \delta^{V^2}(q_0^{V^2}, P_{\Sigma_{o_2}}(s)).
\]

Note that transitions of observable events that appear possible in \(V^1||V^2\) might not necessarily be executable by the system. The algorithm eliminates these by computing the product of a modified version of \(V^1||V^2\), denoted by \(\tilde{V}^1||\tilde{V}^2\), with the system, i.e. \(G \times (\tilde{V}^1||\tilde{V}^2)\). Automaton \(V^1||V^2\) includes all events in \(\Sigma\) in its set of events (unlike automaton \(V^1||V^2\) which only includes the observable events \(\Sigma_o \equiv \Sigma_{o_1} \cup \Sigma_{o_2}\)). Specifically, automaton \(\tilde{V}^1||\tilde{V}^2\) appends a self loop transition for each unobservable event \(\sigma_u \in \Sigma_{uo}\) (note that \(\Sigma_{uo} = \Sigma \setminus \Sigma_o\)) at each state of \(V^1||V^2\), i.e.

\[
\delta^{\tilde{V}^1||\tilde{V}^2}((q^{V^1}, q^{V^2}), \sigma_u) = (q^{V^1}, q^{V^2}) \quad \text{for } \sigma_u \in \Sigma_{uo}.
\]

Thus, the state space of \(\tilde{V}^1||\tilde{V}^2\) remains the same as \(V^1||V^2\), but the event space is augmented in order to include unobservable events in \(\Sigma_{uo}\).

### C. Necessary and Sufficient Conditions for IBDD

We verify IBDD diagnosability by checking for \(F_i\)-uncertain states in \(G \times \tilde{V}^1||\tilde{V}^2\), where \(F_i\)-uncertainty is now defined under the intersection-based scheme. The automaton
Definition 6: A state \((q^{V_1}, q^{V_2})\) of \(\tilde{V}_1 || \tilde{V}_2\) is Intersection Based \(F_i\)-uncertain if the intersection of its local estimates includes a subset of the form \(\{(q, l), (q', l')\}\) with \(q, q' \in Q\) and \(F_i \in l\) but \(F_i \notin l'\) (or vice versa).

Remark 2: Note that a state \(q^{V_1}\) of a (local) verifier \(V_j^i\) consists of exactly two state estimates of the system. Therefore, the above definition holds only whenever \(q^{V_1}, q^{V_2}\) are both \(F_i\)-uncertain and consist of the same pair of state estimates and label. Also important is the observation that labels that become faulty with label \(F_i\) in a local verifier \(V_j^i\), and thus also in the parallel composition automaton \(V_1 || V_2\), have to remain faulty with label \(F_i\) (due to the way function \(f\) is defined in Eq. (1)). □

Since diagnosability is considered in the case when multiple requests are allowed, we are particularly interested in Intersection Based \(F_i\)-uncertain states that exist within cycles of the Product Verifier. Such cycles are called Intersection Based \(F_i\)-confused cycles and are defined as follows.

Definition 7: A cycle of the Product Verifier \((\langle (v_1^n, v_2^n), (v_2^n, v_1^n) \rangle)_{|\tilde{V}_1 || \tilde{V}_2|}\) is Intersection Based \(F_i\)-confused if each of the pairs \((v_1^1, v_2^1), (v_1^2, v_2^2), \ldots, (v_1^n, v_2^n)\) are Intersection Based \(F_i\)-uncertain states.

By Remark 2, we note that \(\langle (v_1^1, v_2^1), (v_1^2, v_2^2), \ldots, (v_1^n, v_2^n) \rangle_{|\tilde{V}_1 || \tilde{V}_2|}\) is Intersection Based \(F_i\)-confused if \(\langle v_1^1, v_2^1, v_1^2, v_2^2, \ldots, v_1^n, v_2^n \rangle_{|\tilde{V}_1 || \tilde{V}_2|}\) are \(F_i\)-confused cycles and \(v_2^k\) consist of the same pairs of \(F_i\)-uncertain state estimates as \(v_2^k\) for \(k = 1, \ldots, n\).

Considering the above, we can now state the following theorem, the proof of which is omitted due to space limitations.

Theorem 1: A non-deterministic system \(G\) observed by \(n = 2\) observers under natural projections with observable events \(\Sigma_{o_1}\) and \(\Sigma_{o_2}\) respectively, under SP1–SP2, is \(F_i\)-diagnosable under IBDD with respect to \(\Sigma_{o_1}, \Sigma_{o_2}\), and partition \(\Pi_f\) on \(\Sigma_f\) if and only if the automaton \(G \times \tilde{V}_1 || \tilde{V}_2\) has no intersection based \(F_i\)-confused cycles.

Example 4: Consider again the system \(G\) in Fig. 1a. Automaton \(G \times (\tilde{V}_1 || \tilde{V}_2)\) contains (amongst others) the cycles \(\langle 3F, 3F4N, 5N3F \rangle, \langle 3F, 4N3F, 5N3F \rangle, \langle 3F, 3F4N, 3F5N \rangle, \langle 3F, 4N3F, 3F5N \rangle, \langle 3F4N, 4N3F \rangle, \langle 5N3F, 3F5N \rangle\), where \(\langle 3F4N, 4N3F \rangle\) and \(\langle 5N3F, 3F5N \rangle\) are \(F_i\)-confused cycles in \(V_1\) and \(V_2\) respectively (refer to Fig. 3). However, the intersection \(\langle 3F4N, 4N3F \rangle \cap \{5N, 3F\}\) gives \(\{3F\}\) which is not an \(F_i\)-uncertain state. Hence, \(G \times (\tilde{V}_1 || \tilde{V}_2)\) has no intersection based \(F_i\)-confused cycles, and the system is IBDD.

V. Conclusions

In this work, we have introduced the notion of intersection based decentralized diagnosis (IBDD), compared it to the notion of codiagnosability, and argued that the former is superior to the latter (at the cost of sending state and matching normal/fault condition information to the coordinator, rather than pure diagnosis decisions). The implementation of IBDD requires polynomial complexity at the local observation sites (in order to recursively maintain a list of possible states and fault conditions following a sequence of observations) whereas the coordinator only needs to take the intersection of states provided by the local observation sites (without necessarily having knowledge of the system model). For the verification of IBDD, we have proposed an algorithm that constructs local verifier automata (for local diagnosis) and combines them via the parallel composition of local verifiers and the product with the system model (for the global diagnosis decision). The algorithm is applicable to arbitrary non-deterministic systems (deterministic systems are a special case) and has polynomial complexity. Future extensions of this work include investigating the efficiency of the algorithm under different communication rules.

REFERENCES