Discrete Event Dyn Syst (2015) 25:601–603
DOI 10.1007/s10626-013-0161-7

ERRATUM

Erratum to: Diagnosis of discrete event systems using decentralized architectures

Yin Wang · Tae-Sic Yoo · Stéphane Lafortune

Published online: 5 April 2013

Erratum to: Discrete Event Dyn Syst DOI 10.1007/s10626-006-0006-8

It was pointed out to us by S. Takai and T. Yamamoto (S. Takai and T. Yamamoto, August 2012. Private communication) that Theorem 9 in the original article *Diagnosis of discrete event systems using decentralized architectures* was incorrect in that the stated condition for violation of negative codiagnosability is sufficient but not necessary in general. We are very grateful to them for bringing this issue to our attention. The lack of necessity in Theorem 9 also carries over to Theorems 16 and 17 in the original article. In this paper, we explain how to adjust these three theorems so that necessity and sufficiency hold.

In the original article, we called a one-level verifier state $(q_1, l_1, q_2, l_2, q_3, l_3)$ a (l_1, l_2, l_3)-state, and a strongly connected component (SCC) a (l_1, l_2, l_3)-SCC, if every state in the SCC is a (l_1, l_2, l_3)-state. Here, we extend the label l_i to include symbol "?" in addition to "P" and "N"; the new symbol "?" means either "P" or "N". For example, a (?,P,N)-state can be either a (P,P,N)-state or a (N,P,N)-state. The corrected statements and revised proofs of Theorems 9, 16, and 17 are as follows.

The online version of the original article can be found at http://dx.doi.org/10.1007/s10626-006-0006-8.

Y. Wang (✉)
HP Labs, Palo Alto, CA 94304, USA
e-mail: yinw@eecs.umich.edu

T.-S. Yoo
Idaho National Laboratory, Idaho Falls, ID 83403, USA
e-mail: Tae-Sic.Yoo@inl.gov

S. Lafortune
University of Michigan, Ann Arbor, MI 48109-2122, USA
e-mail: stephane@eecs.umich.edu

Theorem 9 *The language generated by system G is not negative-codiagnosable if and only if the one-level verifier V_1 of G has a path with two SCCs (that may or may not be distinct), a (P,?,N)-SCC and a (?,P,N)-SCC, where each SCC has a transition whose event corresponding to the "P" in the 3-tuple is not ε.*

Proof We consider the general case where the two SCCs are distinct. The proof can be straightforwardly adapted to the special case where the two SCCs are the same, i.e., the case of a (P,P,N)-SCC.

(i) Two SCCs along a path $\Rightarrow$ not negative-codiagnosable. Without loss of generality, we assume the (P,?,N)-SCC is *before* the (?,P,N)-SCC along the path. Based on Proposition 7, we can induce a trace triple $s_1 t_1^n u_1 v_1^m, s_2 t_2^n u_2 v_2^m, st^n uv^m$ from this path, where trace triple (s_1, s_2, s) corresponds to the prefix of the path that reaches the (P,?,N)-SCC from the initial state, (t_1, t_2, t) corresponds to the edges within the (P,?,N)-SCC, (u_1, u_2, u) corresponds to the path from the (P,?,N)-SCC to the (?,P,N)-SCC, and (v_1, v_2, v) corresponds to the edges within the (?,P,N)-SCC. We know that $st^n uv^m$ must be negative, while s_1 and $s_2 t_2^n u_2$ are positive. Furthermore, we can select t_1 and v_2 such that $t_1, v_2 \neq \varepsilon$. Then the triple $s_1 t_1^n u_1 v_1^m, s_2 t_2^n u_2 v_2^m, st^n uv^m$ violates the definition of negative-codiagnosability.

(ii) Not negative-codiagnosable $\Rightarrow$ two SCCs along a path. Not negative-codiagnosable implies there exist negative trace u and positive traces s_1 and s_2 with arbitrarily long extensions t_1 and t_2 such that $\mathcal{P}_1(u) = \mathcal{P}_1(s_1 t_1)$ and $\mathcal{P}_2(u) = \mathcal{P}_2(s_2 t_2)$. By Proposition 7, these three traces must form a path in V_1. Since both t_1 and t_2 can be arbitrarily long and V_1 has only a finite number of states, there must be two SCCs corresponding to t_1 and t_2, respectively. The SCC corresponding to t_1 is a (P,?,N)-SCC since s_1 is positive, and the other corresponding to t_2 is a (?,P,N)-SCC since s_2 is positive. Each SCC must have an edge with non-ε corresponding to the "P" component in the 3-tuple in order to induce the arbitrarily long extensions. □

Theorem 16 *The language generated by system G is not* Cond-Disj-Codiag *if and only if the two-level verifier V_2 of G has three SCCs (that may or may not be distinct) along a path, a (N,P,N,?,?)-SCC, a (N,?,N,P,?)-SCC, and a (N,?,N,?,P)-SCC, where each SCC has at least one transition whose event label corresponding to the "P" in the 5-tuple is not ε.*

Proof

(i) A path with three SCCs $\Rightarrow$ not conditionally positive-codiagnosable. Based on Proposition 15, this three-SCC path eventually reaches a (N,P,N,P,P) state. We can induce a 5-tuple trace $u_1, v_1 w_1, u_2, v_2 w_2, st$ from this path, where u_1, u_2 are negative, v_1, v_2, s are positive, and w_1, w_2, t are arbitrarily long. Now, positive trace st is indistinguishable from negative trace u_1 at site 1 and indistinguishable from negative trace u_2 at site 2. Furthermore, site 2's estimate of u_1 contains a trace with positive prefix v_1 and site 1's estimate of u_2 contains a trace with prefix v_2. Thus it is not conditionally positive-codiagnosable.

(ii) Not conditionally positive-codiagnosable $\Rightarrow$ a path with three SCCs. If the system is not conditionally positive-codiagnosable, then there exist arbitrarily long positive trace st, negative traces u_1 and u_2 that have the same projection at site 1 and 2, respectively, and arbitrarily long positive traces $v_1 w_1$ and $v_2 w_2$ such that $\mathcal{P}_2(u_1) = \mathcal{P}_2(v_1 w_1)$ and $\mathcal{P}_1(u_2) = \mathcal{P}_1(v_2 w_2)$. By Proposition 15, these five traces should form an arbitrarily long path in V_2. This path must contain three SCCs, (N,P,N,?,?)-SCC, (N,?,N,P,?)-SCC, and (N,?,N,?,P)-SCC, corresponding to arbitrarily long traces $v_1 w_1$, $v_2 w_2$, and st, respectively. The non-ε edges in each SCC are due to the the non-empty suffixes w_1, w_2, and t. □

Theorem 17 *The language generated by system G is not* Cond-Conj-Codiag *if and only if the two-level verifier V_2 of G has two SCCs (that may or may not be distinct) along a path, a (P,N,?,N,N)-SCC and a (?,N,P,N,N)-SCC, where each SCC has at least one transition whose event corresponding to the "P" in the 5-tuple is not ε.*

The proof is similar to that of Theorem 16 and omitted.