

CODIAGNOSTICABILIDADE DE SISTEMAS A EVENTOS DISCRETOS EM REDE COM INFORMAÇÕES DE TEMPORIZAÇÃO ENTRE OCORRÊNCIAS DE EVENTOS E ATRASOS NOS CANAIS DE COMUNICAÇÃO

GUSTAVO DA SILVA VIANA*, MARCOS V. S. ALVES*, JOÃO CARLOS BASILIO*

*Universidade Federal do Rio de Janeiro
COPPE - Programa de Engenharia Elétrica
21949-900, Rio de Janeiro, R.J., Brasil

Emails: gustavo.viana@poli.ufrj.br, mvalves@poli.ufrj.br, basilio@poli.ufrj.br

Abstract— We address, in this paper, the problem of codiagnosability of time networked discrete event systems (TNDES) subject to time delays and losses of observations of events between the measurement sites (MS) and local diagnosers (LD), and, for this purpose, we first introduce a new time DES model that represents the dynamic system behavior of the plant based on the, *a priori*, knowledge of the minimal firing time for each transition of the plant and on the maximal delays in the communication channels that connect the MS and the LD, and also intermittent packet losses in the communication network. We, then, convert this time DES model in an untimed one, and, based on the untimed model, we present necessary and sufficient conditions for TNDES codiagnosability and an automaton-based algorithm for its verification. Examples illustrate all the results presented in the paper.

Keywords— Discrete event systems, communication networks, failure diagnosis.

Resumo— Neste artigo, é abordado o problema da codiagnosticabilidade em sistemas a eventos discretos (SEDs) em redes sujeitas a atrasos temporais de comunicação (SEDRT) e perdas de observação de eventos entre os locais de medição (LM) e os diagnosticadores locais (DL). Para tanto, será proposto um novo modelo temporizado que representa o comportamento do sistema dinâmico da planta com base no conhecimento *a priori* do tempo de disparo mínimo de cada transição da planta, os atrasos máximos nos canais de comunicação que ligam os LM e os DL e também as perdas intermitentes de pacotes na rede de comunicação. Este modelo temporizado será então convertido em um modelo não temporizado equivalente e, com base nesse modelo não temporizado, condições necessárias e suficientes para a codiagnosticabilidade de SEDRT e um algoritmo baseado em autômato para sua verificação são apresentados. Exemplos ilustram todos os resultados apresentados no artigo.

Palavras-chave— Sistemas a eventos discretos, comunicação em rede, diagnose de falhas.

1 Introdução

A maioria dos trabalhos na área de diagnose de falhas de sistemas a eventos discretos (SED) considera que toda a informação é enviada para o diagnosticador de forma imediata e sem interrupções (Sampath et al., 1995; Debouk et al., 2000a). No entanto, devido à complexidade das plantas, os diagnosticadores são muitas vezes implementados de forma distribuída e, conseqüentemente, com o desenvolvimento da tecnologia de rede, tornou-se cada vez mais comum na indústria, implementações de sistemas de comunicação utilizando comunicação em rede (Shu e Lin, 2014).

Em sistemas de diagnose de falhas baseados em redes de comunicação, o intenso tráfego de dados nos canais de comunicação, ou a longa distância entre os locais de medição e os diagnosticadores, podem atrasar as informações enviadas através dos canais. Portanto, quando múltiplos canais de comunicação são implementados, o diagnosticador pode observar eventos com algum atraso após a sua ocorrência e também em ordem diferente de sua ocorrência de fato na planta, sendo levado a tomar decisões erradas sobre uma ocorrência de falha. Além disso, no envio de informações, podem ocorrer perdas de pacotes.

O problema de diagnose de falha de SED com

atrasos em redes de comunicação foi abordado pela primeira vez em Debouk et al. (2000b) e Qiu e Kumar (2008). Em Debouk et al. (2000b), não é suposto qualquer atraso entre os locais de medição e o diagnosticador, e em Qiu e Kumar (2008), cada diagnosticador local pode trocar informações com os outros diagnosticadores locais para inferir a ocorrência da falha e o atraso de comunicação entre dois diagnosticadores locais é considerado igual. Como em Debouk et al. (2000b), não se consideram atrasos entre os locais de medição e diagnosticadores. O problema de SED sujeitos a perdas de observações de eventos foi abordado em Carvalho et al. (2012) (no contexto do diagnose de falha) sem considerar redes de comunicação.

Em Nunes et al. (2016), foi introduzida a definição de codiagnosticabilidade com comunicação em rede de SED sujeitos a atrasos na comunicação de eventos, tal que o atraso corresponde ao tempo que a informação leva para ir da LM para a DL. Tal atraso é medido em passos (Lin, 2014), *i.e.*, $k \in \mathbb{N}$ passos de atraso significa que podem ocorrer, no máximo, k eventos até que a informação do evento executado pela planta chegue ao diagnosticador local.

Neste artigo, é abordado o problema da codiagnosticabilidade de SEDs com redes de comunicação sujeitas a atrasos temporais (SEDRT) e

perdas de observação de eventos entre os locais de medição (LM) e os diagnosticadores locais (DL). Para tanto, será proposto um novo modelo temporizado que representa o comportamento do sistema dinâmico da planta com base no conhecimento, *a priori*, do tempo de disparo mínimo para cada transição da planta, e os atrasos máximos nos canais de comunicação que ligam LM a DL e também as perdas intermitentes de pacotes na rede de comunicação. Esse modelo temporizado é então convertido em um modelo não temporizado equivalente e, com base nesse modelo não temporizado, são apresentadas condições necessárias e suficientes para a codiagnosticabilidade de SEDRT, além de um algoritmo baseado em autômato para sua verificação. Tais resultados são significativamente superiores aos obtidos em (Nunes et al., 2016), uma vez que, como um tempo mínimo de disparo é associado a cada transição da planta, pode-se, então, definir o conjunto de todas as ordens de observações possíveis para uma mesma sequência da planta com maior precisão do que nas abordagens em que o atraso é limitado apenas pelo número de eventos na sequência.

O modelo TDES (Timed Discrete Event System) proposto em Brandin e Wonham (1994), em que os autores introduzem um evento adicional chamado *tick*, para representar “tick do relógio global”, também poderia ser usado neste trabalho. Entretanto, a principal limitação da representação proposta em Brandin e Wonham (1994) é o caso de sistemas que têm características temporais variadas: devido ao comportamento rápido do sistema, o *tick* estará associado a um pequeno intervalo de tempo e, como consequência, o modelo não temporizado pode ter um elevado número de estados para representar uma dinâmica lenta do sistema no modelo.

Este artigo está estruturado da seguinte forma. A seção 2 apresenta alguns conceitos preliminares. Na seção 3, é introduzido o conceito de sistema temporizado em rede para modelar o comportamento temporal da planta e os canais de comunicação. Na seção 4, é apresentado o modelo da planta sujeito a atrasos de comunicação e em seguida, as perdas intermitentes de observação são incluídas no modelo. Um algoritmo para verificação da codiagnosticabilidade de sistemas a eventos discretos temporizados em rede sujeitos a atrasos de comunicação da ocorrência de eventos e perdas intermitentes de observação é apresentado na seção 5. As conclusões são apresentadas na seção 6.

2 Preliminares

2.1 Notações e definições

No decorrer deste artigo, $G = (X, \Sigma, f, \Gamma, x_0, X_m)$ denotará um autômato determinístico, tal que X é o espaço de estados finitos, Σ é o conjunto de eventos, $f : X \times \Sigma \rightarrow X$ é a função de transição,

parcialmente definida no conjunto de eventos, Γ é a função de eventos ativos, *i.e.*, $\Gamma(x) = \{\sigma \in \Sigma : (\exists y \in X)[f(x, \sigma) = y]\}$, x_0 é o estado inicial do sistema, e X_m o conjunto dos estados marcados. A função de transição quando estendida para $f : X \times \Sigma^* \rightarrow X$, tal que Σ^* é o fecho de Kleene de Σ , permite definir linguagens gerada e marcada por G , ($L(G)$ e $L_m(G)$, respectivamente) da seguinte forma: $L(G) = \{s \in \Sigma^* : f(x_0, s)!\}$ e $L_m(G) = \{s \in \Sigma^* : f(x_0, s) \in X_m\}$, em que $f(x, s)!$ significa que $f(x, s)$ é definido, *i.e.*, existe $y \in X$ tal que $f(x, s) = y$. Considere que \bar{L} denota o fecho de prefixo de L . Se $L = \bar{L}$, então L é dita ser prefixo-fechada.

Define-se a operação *Trim* como $Trim(G) := Coac[Ac(G)] = Ac[Coac(G)]$ tal que *Coac* e *Ac* denotam a parte coacessível e acessível de um autômato, respectivamente (Cassandras e Lafortune, 2008). A parte acessível de G é a operação sobre G que elimina todos os estados de G que não são acessíveis a partir do estado inicial x_0 e suas transições relacionadas, e a parte coacessível de G é obtida eliminando-se todos os estados de G a partir dos quais não é possível alcançar um estado marcado. Para uma dada sequência $s \in L$, define-se pós-linguagem de L após s como $L/s = \{t \in \Sigma^* : st \in L\}$. É suposto aqui que a linguagem gerada por G é sempre viva, uma vez que qualquer linguagem não viva pode ser tornada viva adicionando-se auto-laços de eventos não observáveis nos estados x para os quais $\Gamma(x) = \emptyset$.

A projeção natural é o mapeamento $P : \Sigma_\ell^* \rightarrow \Sigma_s^*$ em que $\Sigma_s \subset \Sigma_\ell$, e a projeção inversa é o mapeamento $P^{-1} : \Sigma_s^* \rightarrow 2^{\Sigma_\ell^*}$, sendo definidas de maneira usual (Cassandras e Lafortune, 2008). A composição paralela e o produto entre G_1 e G_2 , como definido em Cassandras e Lafortune (2008), serão denotados como $G_1 || G_2$ e $G_1 \times G_2$, respectivamente.

2.2 Codiagnosticabilidade de Sistemas a Eventos Discretos

Considere que o conjunto de eventos de G seja particionado como $\Sigma = \Sigma_o \dot{\cup} \Sigma_{uo}$, em que Σ_o e Σ_{uo} denotam o conjunto de eventos observáveis e não observáveis, respectivamente, e $\Sigma_f \subseteq \Sigma_{uo}$ seja o conjunto de eventos de falha. Por simplicidade, suponha que há apenas um evento de falha, *i.e.*, $\Sigma_f = \{\sigma_f\}$. Um sequência s é dita ser de falha se $\Sigma_f \in s$, e dita normal se $\Sigma_f \notin s$. O conjunto de todas sequências normais geradas pelo sistema é a linguagem prefixo-fechada $L_N \subset L$. Portanto, o conjunto de todas as sequências de falha é dado por $L \setminus L_N$. G_N é o subautômato de G que gera a linguagem L_N .

Neste artigo, é adotado o esquema de diagnose descentralizada apresentado no Protocolo 3 de Debouk et al. (2000a), o qual levou ao conceito de codiagnosticabilidade. A ideia por trás do problema da codiagnosticabilidade é que toda

sequência s , tal que $\Sigma_f \in s$, deve ser diagnosticada por, pelo menos, um diagnosticador local.

Definição 1 (Debouk et al., 2000a) Considere que L e $L_N \subset L$ sejam linguagens prefixo-fechadas geradas por G e G_N , respectivamente, e $P_{o_i} : \Sigma^* \rightarrow \Sigma_{o_i}^*$, $i = 1, \dots, n$, sejam operações de projeção. Portanto, L é codiagnosticável em relação às projeções P_{o_i} e Σ_f se

$$(\exists z \in \mathbb{N})(\forall s \in L \setminus L_N)(\forall st \in L \setminus L_N, \|t\| \geq z) \Rightarrow (\exists i \in \{1, 2, \dots, n\})(P_{o_i}(st) \neq P_{o_i}(\omega), \forall \omega \in L_N),$$

em que $\|\cdot\|$ é o comprimento de uma sequência.

O fato de nenhum dos módulos locais poder sozinho diagnosticar a ocorrência de falha está implícito na definição de codiagnosticabilidade, uma vez que, do contrário não seria necessário usar uma estrutura descentralizada. De acordo com a definição 1, L é codiagnosticável em relação a P_{o_i} e Σ_f se, e somente se, para todas as sequências de falha $s_F = st$ de comprimento arbitrariamente longo após a ocorrência do evento de falha, não existem sequências $s_{N_i} \in L_N$, tal que $P_{o_i}(s_{N_i}) = P_{o_i}(s_F)$, para todo $i \in \{1, 2, \dots, n\}$.

A verificação de codiagnosticabilidade de L é o primeiro passo para a diagnose de falha de um SED. Neste artigo, será utilizado o algoritmo proposto por Moreira et al. (2011) para construir um autômato verificador para a verificação de codiagnosticabilidade de SED em rede sujeitos a atrasos de comunicação e perdas intermitentes de observação de eventos usando informações de tempo.

3 Formulação do problema

Neste artigo, é considerada a arquitetura em rede para uma planta distribuída introduzida por Nunes et al. (2016), formada com m locais de medição LM_j , $j = 1, \dots, m$ e n diagnosticadores locais DL_i , para $i = 1, \dots, n$. Cada local de medição LM_j pode detectar um subconjunto $\Sigma_{LM_j} \subset \Sigma_o$ dos eventos observáveis do sistema. Nesta configuração, apenas os eventos detectados pelo local de medição LM_j podem ser comunicados através do canal ch_{ij} para o diagnosticador local DL_i .

Diferentemente da abordagem adotada em Nunes et al. (2016), em que os atrasos de comunicação foram representados por passos, neste artigo considera-se que cada canal ch_{ij} tem um tempo de atraso máximo $T_{ij} \in \mathbb{R}_+$, em que \mathbb{R}_+ denota o conjunto de números reais positivos. $\Sigma_{o_{ij}} \subseteq \Sigma_{LM_j}$ denota o conjunto de eventos comunicados ao diagnosticador local DL_i , através do canal de comunicação ch_{ij} . Se o canal de comunicação ch_{kw} , entre um local de medição LM_w e um diagnosticador local DL_k , não existir, então $\Sigma_{o_{kw}} = \emptyset$. Assim, o conjunto de eventos observáveis de DL_i , Σ_{o_i} , é dado por $\Sigma_{o_i} = \bigcup_{j=1}^m \Sigma_{o_{ij}}$ e o conjunto de eventos observáveis de todo o sistema é $\Sigma_o = \bigcup_{i=1}^n \Sigma_{o_i}$.

Para abordar o problema da codiagnosticabilidade em SEDRT, as seguintes hipóteses devem ser feitas.

- A1.** $L(G)$ é codiagnosticável em relação a P_{o_i} , $i = 1, \dots, n$ e Σ_f , em que G é o autômato que modela a planta.
- A2.** Há apenas um canal de comunicação ch_{ij} entre o local de medição LM_j e o diagnosticador local DL_i , comunicando os eventos em $\Sigma_{o_{ij}}$.
- A3.** Cada canal ch_{ij} , é modelado is por uma fila “first-in first-out” (FIFO), e sujeito a um tempo máximo de atraso previamente conhecido, $T_{ij} \in \mathbb{R}_+$, T_{ij} finito.
- A4.** Os conjuntos de eventos $\Sigma_{LM_i} \cap \Sigma_{LM_j} = \emptyset$, $i, j \in \{1, 2, \dots, m\}$, $i \neq j$.

Em relação à perda de dados nos canais de comunicação, a seguinte hipótese é feita:

- A5.** $\Sigma_o = \Sigma_{ilo} \dot{\cup} \Sigma_{nilo}$, em que Σ_{ilo} (resp. Σ_{nilo}) denota o conjunto de eventos sujeitos (resp. não sujeitos) a perdas de comunicação.

As consequências do atraso na transmissão de observação de eventos em sistemas de diagnose de falhas dependem da dinâmica da planta, uma vez que se a dinâmica da planta é suficientemente lenta, atrasos na transmissão podem não gerar consequências adversas. Portanto para melhor considerar os efeitos do comportamento dinâmico em sistemas reais, é razoável considerar que uma transição de estado de G pode não ocorrer imediatamente depois de uma anterior, isto é, o sistema precisa ficar por algum tempo em um estado antes de uma nova transição ocorrer. Isto leva à definição de sistema temporizado em rede.

Definição 2 (Sistema temporizado em rede) Um sistema temporizado em rede $TNetSys$ é uma tripla $TNetSys = (G, t_{min}, T)$, em que $G = (X, \Sigma, f, \Gamma, x_0, X_m)$ é um autômato finito, $t_{min} : X \times \Sigma \rightarrow \mathbb{R}_+$ é a função de tempo mínimo que associa um tempo mínimo de disparo de cada transição de G e é definida, sobre o par $(x, \sigma) \in X \times \Sigma$ se, e somente se, a transição $f(x, \sigma)$ é definida. $T : n \times m$ é a matriz de atraso máximo, em que cada T_{ij} é o tempo máximo de atraso de cada canal ch_{ij} tal que $T_{ij} \in \mathbb{R}_+$, se existe um canal de comunicação entre o local de medição LM_j e o diagnosticador local DL_i , e $T_{ij} = \infty$, caso contrário.

De acordo com a definição 2, uma transição de estado de $TNetSys$ apenas ocorre quando o tempo decorrido desde a última transição é maior que o conhecido tempo mínimo de ativação $t_{min}(x, \sigma) \in \mathbb{R}_+$. Note que este modelo não é restritivo uma vez que o tempo mínimo de ativação $t_{min}(x, \sigma)$ pode ser tão pequeno quanto possível. O exemplo a seguir ilustra um sistema temporizado em rede.

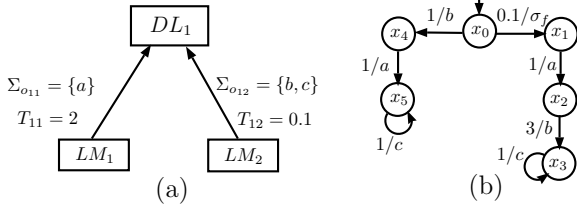


Figura 1: $TNetSys = (G, t_{min}, T)$ do exemplo 1: estrutura com atraso de comunicação (a) e autômato G com função de tempo mínimo t_{min} (b).

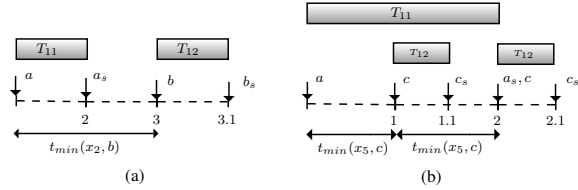


Figura 2: Linha do tempo após a ocorrência do evento a no estado x_1 (a) e no estado x_4 (b) de $TNetSys$ do exemplo 1.

Exemplo 1 Considere o sistema temporizado em rede $TNetSys = (G, t_{min}, T)$ mostrado nas figuras 1(a) e 1(b). Na figura 1(a), é mostrada a estrutura com atraso de comunicação da arquitetura de um SEDRT. Note que ela consiste de dois locais de medição e apenas um diagnosticador local. O local de medição LM_1 comunica apenas o evento a ao diagnosticador local DL_1 através do canal ch_{11} , com atraso máximo $T_{11} = 2$ unidades de tempo (u.t.) e o local de medição LM_2 comunica os eventos b e c para o diagnosticador local DL_1 com atraso máximo $T_{12} = 0.1$ u.t. Portanto $T = [T_{11} \ T_{12}] = [2 \ 0.1]$. A figura 1(b) descreve o autômato G juntamente com a função de tempo mínimo t_{min} , que é dada por: $t_{min}(x_0, \sigma_f) = 0.1$, $t_{min}(x_0, b) = t_{min}(x_1, a) = t_{min}(x_3, c) = t_{min}(x_4, a) = t_{min}(x_5, c) = 1$ e $t_{min}(x_2, b) = 3$. Observe que o rótulo sobre a transição $f(x_4, a) = x_5$ significa que $t_{min}(x_4, a) = 1$, i.e., o evento a ocorre pelo menos 1 u.t. depois de G entrar no estado x_4 , que, no exemplo corresponde à ocorrência do evento b . Os rótulos restantes são interpretados de forma semelhante.

Quando o sistema gera a sequência $s_1 = \sigma_f abc^p$, em que $p \in \mathbb{N}$, o evento a não pode ser observado pelo diagnosticador local DL_1 após a ocorrência do evento b , uma vez que a observação do evento a pode atrasar no máximo $T_{11} = 2$ u.t. e o evento b ocorre pelo menos 3 u.t. após a ocorrência do evento a como mostrado na linha do tempo representada na Figura 2(a), onde a_s e b_s indicam a observação bem-sucedida dos eventos a e b , respectivamente, pelo diagnosticador local DL_1 . Além disso, não há mudança na ordem de observação dos eventos b e c , uma vez que são transmitidos através do mesmo canal de comuni-

cação. Portanto, após a ocorrência da sequência s_1 , o diagnosticador observa $s_{1s} = a_s b_s c_s^p$, em que c_s denota a observação bem-sucedida de eventos c e, portanto, a ordem das sequências de eventos observadas não é afetada devido à dinâmica do sistema. Quando o sistema executa $s_2 = bac^q$, em que $q \in \mathbb{N}$, os eventos b e a não são observados fora da ordem de suas ocorrências, uma vez que $t_{min}(x_4, a) > T_{12}$. Entretanto, é possível que o diagnosticador observe $s_{2s} = b_s a_s c_s^q$ ou $s'_{2s} = b_s c_s a_s c_s^{q-1}$, uma vez que $t_{min}(x_5, c) < T_{11}$. Por inspeção, pode-se notar que mesmo com esta mudança na ordem de observação de eventos, o diagnosticador local DL_1 pode diagnosticar a ocorrência do evento de falha. A linha do tempo, representada na figura 2(b), mostra a possibilidade de o evento a ser observado após a observação de um evento c .

Observação 1 Considere que o atraso na comunicação do sistema temporizado do exemplo 1 fosse medido em passos como proposto em Nunes et al. (2016) e Lin (2014), supondo que os atrasos nos canais ch_{11} e ch_{12} são de no máximo 1 e 0 passos, respectivamente. Então a mudança na ordem de observação dos eventos a e b depois da sequência $s_1 = \sigma_f abc^n$ também deve ser levada em consideração, i.e., o diagnosticador local pode observar $s'_1 = b_s a_s c_s^n$, que é igual a uma possível observação de uma sequência normal $s_2 = bac^n$, $s'_2 = b_s a_s c_s^n$. Sendo assim, de acordo com a abordagem utilizando passos, o diagnosticador local não é capaz de diagnosticar a ocorrência do evento de falha, que é incorreto, uma vez que como visto no exemplo 1, tal observação não é possível.

4 Modelagem

Para caracterizar o comportamento de SEDRT na presença de atrasos de observações, será apresentado um algoritmo para a computação de um autômato, $G_i = (X_i, \Sigma_i, f_i, \Gamma_i, x_{0_i}, \emptyset)$, formado a partir de $TNetSys$, cujos estados têm dois componentes: (i) o primeiro componente representa o estado correspondente de G , e (ii) o segundo componente representa os eventos observáveis que foram gerados por G para alcançar o estado x e cujas observações estão sendo transmitidas ao diagnosticador local DL_i juntamente com o tempo mínimo decorrido entre as ocorrências desses eventos observáveis. A construção de G_i é baseada na manipulação das sequências compostas por eventos e números reais que formam a segunda componente dos estados de G_i . Para tanto, pode se definir o conjunto $\mathcal{Q}_i := \{q = q_1 q_2 \dots q_l : \forall w \in \{1, 2, \dots, l\}, (q_w \in \Sigma_{o_i}) \vee (q_w \in \mathbb{R}_+^0)\}$, em que \mathbb{R}_+^0 denota o conjunto de números reais não negativos e $i \in \{1, 2, \dots, n\}$. Note que os elementos de \mathcal{Q}_i são sequências formadas por eventos observáveis em Σ_{o_i} e números em \mathbb{R}_+^0 . As seguintes operações sobre o conjunto \mathcal{Q}_i podem ser definidas.

Definição 3 (Operações sobre o conjunto \mathcal{Q}_i)

(a) A operação de concatenação é o mapeamento $conc : \mathcal{Q}_i \times \mathcal{Q}_i \rightarrow \mathcal{Q}_i$ em que, para cada $q = q_1 \cdots q_l$ e $p = p_1 \cdots p_k$ pertencente a \mathcal{Q}_i ,

$$conc(q, p) = \begin{cases} q_1 \cdots q_{l-1} (q_l + p_1) p_2 \cdots p_k, \\ \quad \text{se } q_l, p_1 \in \mathbb{R}_+^0 \\ q_1 \cdots q_l p_1 \cdots p_k, \text{ caso contrário.} \end{cases}$$

(b) A operação de eliminação é o mapeamento $cut : \mathcal{Q}_i \rightarrow \mathcal{Q}_i$ em que, $\forall q = q_1 q_2 \cdots q_l \in \mathcal{Q}_i$,

$$cut(q) = \begin{cases} q_w q_{w+1} \cdots q_l, \text{ se } (\exists w \leq l)[(q_w \in \Sigma_{o_i}) \\ \quad \wedge (q_j \in \mathbb{R}_+^0, \forall j \in \{1, \dots, w-1\})] \\ 0, \text{ se } q_j \in \mathbb{R}_+^0, \forall j \in \{1, 2, \dots, l\}. \end{cases}$$

(c) A operação de adição é o mapeamento $add : \mathcal{Q}_i \times X \times \Sigma \rightarrow \mathcal{Q}_i$ em que, $\forall q = q_1 q_2 \cdots q_l \in \mathcal{Q}_i$, $x \in X$ e $\sigma \in \Sigma$,

$$add(q, x, \sigma) = \begin{cases} cut(conc(q, t_{min}(x, \sigma)\sigma)), \text{ se } (\sigma \\ \quad \in \Sigma_{o_i}) \wedge (f(x, \sigma)!) \\ cut(conc(q, t_{min}(x, \sigma))), \text{ se } (\sigma \\ \quad \in \Sigma_{uo_i}) \wedge (f(x, \sigma)!) \\ \text{indefinido, caso contrário.} \end{cases}$$

(d) A operação de remoção é o mapeamento

$$rem : \mathcal{Q}_i \times \mathbb{N} \rightarrow \mathcal{Q}_i$$

em que, $\forall q = q_1 q_2 \cdots q_l \in \mathcal{Q}_i$,

$$rem(q, k) = \begin{cases} cut(q_2 \cdots q_l), \text{ se } (k = 1) \\ conc(q_1 \cdots q_{k-1}, q_{k+1} \cdots q_l), \\ \quad \text{se } (1 < k < l) \\ cut(q_1 \cdots q_{l-1}), \text{ se } (k = l) \\ \text{indefinido, caso contrário.} \end{cases}$$

(e) A função índice de local de medição ms é o mapeamento

$$ms : \Sigma_{o_i} \rightarrow \{1, 2, \dots, m\}$$

em que, para todo $\sigma \in \Sigma_{o_i}$,

$$ms(\sigma) = \begin{cases} j : \text{se } \sigma \in \Sigma_{o_{ij}} \text{ para algum } i \in \{1, \dots, n\} \\ \text{indefinido, caso contrário.} \end{cases}$$

Para distinguir um evento $\sigma \in \Sigma_{o_{ij}}$, que é transmitido do local de medição LM_j para o diagnosticador local DL_i através do canal de comunicação ch_{ij} , a partir de sua observação pelo diagnosticador local DL_i , é criado um evento σ_{s_i} que modela a observação bem-sucedida de σ por DL_i e forma o seguinte conjunto:

$$\Sigma_{o_{ij}}^s = \{\sigma_{s_i} : \sigma \in \Sigma_{o_{ij}}\}. \quad (1)$$

Considere que

$$\Sigma_{o_i}^s = \bigcup_{j=1}^m \Sigma_{o_{ij}}^s, \quad (2)$$

denote o conjunto de eventos observáveis que são comunicados com sucesso para o diagnosticador

local DL_i . Então, a seguinte função bijetora pode ser definida:

$$\psi_i : \begin{cases} \Sigma_{o_i}^s \rightarrow \Sigma_{o_i}, \\ \sigma_{s_i} \mapsto \psi_i(\sigma_{s_i}) = \sigma. \end{cases}$$

O comportamento do sistema na presença de atrasos será, portanto, modelado por uma linguagem definida sobre o seguinte conjunto estendido de eventos:

$$\Sigma_i := \Sigma \cup \Sigma_{o_i}^s \quad (3)$$

A partir das definições dos conjuntos anteriores, pode-se, então, apresentar um algoritmo para a construção do autômato G_i .

Algoritmo 1 (Construção do autômato G_i)

Entradas:

- Sistema temporizado em rede TNetSys = (G, t_{min}, T) ;
- $\Sigma_{o_{ij}}$, para todo $j \in \{1, \dots, m\}$: os conjuntos de eventos observáveis comunicados por cada local de medição LM_j ao diagnosticador local DL_i ;
- Σ_{o_i} : o conjunto de todos os eventos observáveis pelo diagnosticador local DL_i ;

Saída:

- $G_i = (X_i, \Sigma_i, f_i, \Gamma_i, x_{o_i}, \emptyset)$.

Passo 1: Defina o estado inicial $x_{o_i} = (x_0, 0)$ e $X_i = \emptyset$;

Passo 2: Defina $\Sigma_{o_{ij}}^s$, para todo $j \in \{1, \dots, m\}$, $\Sigma_{o_i}^s$ e Σ_i de acordo com as equações (1), (2) e (3), respectivamente;

Passo 3: Crie a fila FIFO F e adicione x_{o_i} a F ;

Passo 4: Enquanto $F \neq \emptyset$ faça:

◦ 4.1: $(x, q) = F_1$, em que F_1 é o primeiro elemento da fila F , e remova F_1 de F .

◦ 4.2: $X_i \leftarrow X_i \cup \{(x, q)\}$;

◦ 4.3: Considere $q = q_1 q_2 \cdots q_l$. Defina o conjunto de índices $I_{o_i} = \{k \in \{1, \dots, l\} : q_k \in \Sigma_{o_i}\}$;

◦ 4.4: Para todo $y \in I_{o_i}$, faça $minst(y)$: a soma dos números reais à direita de q_y em q ;

◦ 4.5: Para todo $\sigma \in \Gamma(x)$:

(a) Faça $\tilde{x}_i = f_i((x, q), \sigma) =$

$$\begin{cases} (f(x, \sigma), add(q, x, \sigma)), \text{ se } (minst(y) + \\ \quad + t_{min}(x, \sigma)) < T_{iw}, w = ms(q_y), \forall y \in I_{o_i}; \\ \text{indefinido, caso contrário.} \end{cases}$$

(b) Se $(\tilde{x}_i \notin X_i) \wedge (\tilde{x}_i \notin F)$, e adicione \tilde{x}_i para o fim da fila F ;

◦ 4.6: Para todo $\Sigma_{o_{ij}}$, $j = 1, \dots, m$:

(a) $y = \min\{j \in I_{o_i} : \sigma_j \in \Sigma_{o_{ij}}\}$;

(b) $\hat{x}_i = f_i((x, q), \psi_i^{-1}(q_y)) =$

$$\begin{cases} (x, rem(q, y)), \text{ se } y \text{ é definido;} \\ \text{indefinido, caso contrário.} \end{cases}$$

(c) Se $(\hat{x}_i \notin X_i) \wedge (\hat{x}_i \notin F)$, adicione \hat{x}_i para o fim da fila F ;

Passo 5: Defina $\Gamma_i(x_i) = \{\sigma \in \Sigma_i : f_i(x_i, \sigma) \text{ é definido}\}$, para todo $x_i \in X_i$;

Baseado no algoritmo 1, o seguinte teorema relacionado à observação da linguagem gerada por G_i pode ser enunciado.

Teorema 1 A linguagem gerada por G_i , $L(G_i)$, modela todas as possíveis ordens de observação das sequências de $L(G)$ em relação a T_{ij} , para $j = 1, \dots, m$, e Σ_{o_i} .

Prova: A prova é por construção e será omitida devido à falta de espaço.

Com o objetivo de considerar perdas intermitentes de observação, defina $\Sigma_{i,ilo}^s = \psi^{-1}(\Sigma_{i,ilo})$ e $\Sigma_{i,nilo}^s = \psi^{-1}(\Sigma_{i,nilo})$, em que ψ^{-1} denota função inversa de ψ . Uma vez que o conjunto de observação de eventos G_i é dado por $\Sigma_{i,o} = \Sigma_{o_i}^s$, pode-se fazer a seguinte partição do conjunto de eventos observáveis de G_i :

$$\Sigma_{o_i}^s = \Sigma_{i,ilo}^s \dot{\cup} \Sigma_{i,nilo}^s \quad (4)$$

em que os eventos de $\Sigma_{i,ilo}^s$ e $\Sigma_{i,nilo}^s$ denotam o sucesso na observação de eventos de $\Sigma_{i,ilo}$ e $\Sigma_{i,nilo}$, respectivamente, pelo diagnosticador DL_i .

Pode-se definir então os seguintes conjuntos: (i) o conjunto de eventos não observáveis que modela a perda intermitente na observação de eventos $\sigma_s \in \Sigma_{i,ilo}^s$ como $\Sigma_{i,ilo}' = \{\sigma_s' : \sigma_s \in \Sigma_{i,ilo}^s\}$ e; (ii) o conjunto $\Sigma_i' = \Sigma_i \cup \Sigma_{i,ilo}'$. A função dilatação $D_{s_i} : \Sigma_i^* \rightarrow 2^{(\Sigma_i')^*}$ é definida de forma recursiva: $D_{s_i}(\varepsilon) = \{\varepsilon\}$; $D_{s_i}(\sigma) = \{\sigma\}$, se $\sigma \in \Sigma_i \setminus \Sigma_{i,ilo}^s$, $D_{s_i}(\sigma) = \{\sigma, \sigma'\}$, se $\sigma \in \Sigma_{i,ilo}^s$; $D_{s_i}(s_i\sigma) = D_{s_i}(s_i)D_{s_i}(\sigma)$, $\forall s_i \in \Sigma_i^*$, $\forall \sigma \in \Sigma_i$. A operação dilatação D_{s_i} é estendida para linguagens de maneira direta: $D_{s_i}(L) = \bigcup_{s \in L} D_{s_i}(s)$. Portanto, pode-se obter o autômato G_i' que gera a linguagem $D_{s_i}[L(G_i)]$, da seguinte forma:

$$G_i' = (X_i, \Sigma_i', f_i', \Gamma_i', x_{0_i}, \emptyset), \quad (5)$$

em que $\Gamma_i'(x_i) = D_{s_i}[\Gamma_i(x_i)]$, $\forall x_i \in X_i$, e $f_i'(x_i, \sigma') = f_i(x_i, \sigma)$, se $\sigma' \in \Sigma_{i,ilo}'$, e $f_i'(x_i, \sigma) = f_i(x_i, \sigma)$, se $\sigma \in \Sigma_i \setminus \Sigma_{i,ilo}^s$. Note que, se $\Sigma_{i,ilo} = \emptyset$, $G_i' = G_i$, que implica que $D_{s_i}[L(G_i)] = L(G_i)$.

5 Verificação da codiagnosticabilidade em rede

Nesta seção apresenta-se um método para verificação da codiagnosticabilidade em rede (CR). Primeiramente, é necessário introduzir uma função para obter todas as sequências de G_i cujas projeções fornecem todas as possibilidades de ordens de observação de uma sequência $s \in L(G)$ devido aos atrasos na comunicação. Considere

$\chi_i : L(G) \rightarrow 2^{\Sigma_i^*}$, em que $\chi_i(s) = P_i^{-1}(s) \cap L(G_i)$ e $P_i : \Sigma_i^* \rightarrow \Sigma^*$ é uma projeção. Note que $L(G_i) = \bigcup_{s \in L} \chi_i(s)$.

Pode-se então definir codiagnosticabilidade em rede da linguagem gerada por um SEDRT da seguinte forma.

Definição 4 Considere L e $L_N \subset L$ como as linguagens prefixo-fechadas geradas por G e G_N , respectivamente, e $I_n = \{1, \dots, n\}$. Então, L é dito ser codiagnosticável em rede em relação a $\chi_i : \Sigma^* \rightarrow 2^{\Sigma_i^*}$, D_{s_i} , projeções $P_{s_i}' : \Sigma_i^* \rightarrow \Sigma_{o_i}^*$, para $i = 1, \dots, n$, e Σ_f

$$(\exists z \in \mathbb{N})(\forall s \in L \setminus L_N)(\forall st \in L \setminus L_N, ||t|| \geq z) \Rightarrow (\exists i \in I_n)[P_{s_i}'[D_{s_i}(\chi_i(st))] \cap P_{s_i}'[D_{s_i}(\chi_i(\omega_i))] = \emptyset, \forall \omega_i \in L_N].$$

Definição 5 (função de renomeação de eventos)

$$\rho_i : \Sigma_{i_N}' \rightarrow \Sigma_{i_p}' \quad (6)$$

$$\sigma \mapsto \rho_i(\sigma) = \begin{cases} \sigma_{\rho_i}, & \text{se } \sigma \in (\Sigma \cup \Sigma_{i,ilo}^s) \setminus \Sigma_f \\ \sigma, & \text{se } \sigma \in \Sigma_{o_i}^s. \end{cases}$$

em que $\Sigma_{i_N}' = \Sigma_i' \setminus \Sigma_f$, para $i = 1, \dots, n$. O domínio da função ρ_i pode ser estendido para $\Sigma_{i_N}^*$ de maneira usual, i.e., $\rho_i(s\sigma) = \rho_i(s)\rho_i(\sigma)$, para todo $s \in \Sigma_{i_N}^*$ e $\sigma \in \Sigma_{i_N}'$. A função ρ_i pode ser também aplicada à linguagem K como $\rho_i(K) = \bigcup_{s \in K} \rho_i(s)$.

Algoritmo 2 (Verificação da CR)

Entrada:

- $G_i' = (X_i, \Sigma_i', f_i', \Gamma_i', x_{0_i}, \emptyset)$, para $i = 1, \dots, n$

Saída:

- $G_V = (X_V, \Sigma_V, f_V, \Gamma_V, x_{0_V}, \emptyset)$.

Passo 1: Compute o autômato $G_{i_N}' = (X_{i_N}', \Sigma_{i_N}', f_{i_N}', \Gamma_{i_N}', (x_{i_0}, N), \emptyset)$, em que $\Sigma_{i_N}' = \Sigma_i' \setminus \Sigma_f$, para $i = 1, \dots, n$, que modela o comportamento normal do autômato G_i' como apresentado em Moreira et al. (2011);

Passo 2: Compute o autômato $G_{i_F}' = (X_{i_F}', \Sigma_{i_F}', f_{i_F}', \Gamma_{i_F}', (x_{i_0}, N), \emptyset)$, para $i = 1, \dots, n$, que modela o comportamento de falha de G_i' como apresentado em Moreira et al. (2011).

Passo 3: Construa o autômato $G_{i,\rho}' = (X_{i,\rho}', \Sigma_{i,\rho}', f_{i,\rho}', \Gamma_{i,\rho}', (x_{i_0}, N), \emptyset)$ em que $\Sigma_{i,\rho}' = \rho_i(\Sigma_{i_N}')$, e $f_{i,\rho}'(x_{i_N}, \sigma_{\rho_i}) = f_{i_N}'(x_{i_N}, \sigma)$ tal que $\sigma_{\rho_i} = \rho_i(\sigma)$, para todo $\sigma \in \Sigma_{i_N}'$ e $x_{i_N} \in X_{i_N}'$.

Passo 4: Compute o autômato $\bar{V}_i = G_{i,\rho}' \parallel G_{i_F}' = (Y_{V_i}, \Sigma_{V_i}, f_{V_i}, \Gamma_{V_i}, y_{V_i,0}, \emptyset)$, para $i = 1, \dots, n$, em que $\Sigma_{V_i} = \Sigma_{i,\rho}' \cup \Sigma_{i_F}'$.

Passo 5: Encontre todos os caminhos cíclicos $cl_i = (y_{V_i}^k, \sigma_k, y_{V_i}^{k+1}, \sigma_{k+1}, \dots, \sigma_\ell, y_{V_i}^k)$, tal que $\ell \geq k > 0$ em \bar{V}_i que satisfaçam a seguinte condição:

$$\exists j \in \{k, k+1, \dots, \ell\} \text{ tal que, para algum } y_{V_i}^j = (x_i^j, N, y_i^j, x_i^j, F), \wedge (\sigma_j \in \Sigma_i') \quad (7)$$

em que $x_i^j, y_i^j \in X_i$.

Passo 6: Compute os autômatos $V_i = (Y_{V_i}, \Sigma_{V_i}, f_{V_i}, \Gamma_{V_i}, y_{V_i,0}, Y_{V_i,m})$, em que $Y_{V_i,m}$ é formado por estados de \bar{V}_i que pertencem a componentes fortemente conexas que contenham caminhos cíclicos cl_i que violem a condição (7).

Passo 7: Compute o autômato verificador $G_V = V_1 \parallel \dots \parallel V_n = (X_V, \Sigma_V, f_V, x_{V,0}, X_{V_m})$, em que $\Sigma_V = \bigcup_{i=1}^n \Sigma_{V_i}$.

Passo 8: Verifique a existência de um caminho cíclico $cl = (x_V^k, \sigma_k, x_V^{k+1}, \sigma_{k+1}, \dots, \sigma_\ell, x_V^k)$ in G_V , $\ell \geq k > 0$, que satisfaça a seguinte condição:

$$x_V^q \in X_{V_m}, \forall q \in \{k, k+1, \dots, \ell\}, \text{ e para algum } q \in \{k, k+1, \dots, \ell\}, \sigma_q \in \Sigma.$$

Se a resposta for “sim”, então L será não codiagnosticável em rede em relação a $\chi_i, D_{s_i}, P'_{s_i}$, para $i = 1, \dots, n$, e Σ_f . Caso contrário, L é codiagnosticável em rede.

Teorema 2 A linguagem L é codiagnosticável em rede em relação a $\chi_i, D_{s_i}, P'_{s_i}$, para $i = 1, \dots, n$, e Σ_f se, e somente se, não existir um caminho cíclico $cl = (x_V^k, \sigma_k, x_V^{k+1}, \sigma_{k+1}, \dots, x_V^\ell, \sigma_\ell, x_V^k)$, $\ell \geq k > 0$ em G_V que satisfaça a seguinte condição: $x_V^q \in X_{V_m}, \forall q \in \{k, k+1, \dots, \ell\}$, e para algum $q \in \{k, k+1, \dots, \ell\}, \sigma_q \in \Sigma$.

Prova: A prova é por construção e será omitida devido à falta de espaço.

Exemplo 2 Considere $TNetSys = (G, t_{min}, T)$ mostrado nas figuras 3(a) e 3(b). A partir da figura 3(a), pode ser vista uma estrutura com atraso de comunicação que possui dois diagnosticadores locais, DL_1 e DL_2 , e três locais de medição, LM_1, LM_2 e LM_3 . Considere que $\Sigma_{LM_1} = \{b\}$, $\Sigma_{LM_2} = \{c\}$ e $\Sigma_{LM_3} = \{a\}$, sejam os conjuntos de eventos que os locais de medição LM_1, LM_2 e LM_3 , respectivamente, detectam. Admita que o conjunto de eventos observáveis do diagnosticador local DL_1 seja $\Sigma_{o_1} = \{b, c\}$. Sendo assim, a ocorrência dos eventos em Σ_{o_1} são transmitidas por meio dos canais de comunicação ch_{11} e ch_{12} , i.e., $\Sigma_{o_{11}} = \{b\}$ e $\Sigma_{o_{12}} = \{c\}$.

Suponha agora que o conjunto de eventos observáveis do diagnosticador local DL_2 seja $\Sigma_{o_2} = \{a, c\}$. Portanto, a ocorrência dos eventos em Σ_{o_2} são transmitidas por meio dos canais de comunicação ch_{22} e ch_{23} , i.e., $\Sigma_{o_{22}} = \{c\}$ e $\Sigma_{o_{23}} = \{a\}$. A figura 3(b) representa o diagrama de transição de estados do autômato G , em que $\Sigma = \{a, b, c, \sigma_f\}$, e a função de tempo mínimo é dada por: $t_{min}(x_0, \sigma_f) = 0.1$, $t_{min}(x_1, a) = t_{min}(x_2, c) = t_{min}(x_4, c) = t_{min}(x_0, b) = t_{min}(x_5, a) = t_{min}(x_6, c) = t_{min}(x_7, a) = t_{min}(x_8, c) = 1$, $t_{min}(x_2, b) = 2$ e $t_{min}(x_3, b) = 0.5$. Finalmente, a matriz de atraso máximo é $T = [T_{11} \ T_{12} \ T_{13}; T_{21} \ T_{22} \ T_{23}]$, em que $T_{11} = 0.3$, $T_{12} = 0.9$, $T_{13} = \infty$, $T_{21} = \infty$, $T_{22} = 0.5$ e $T_{23} = 0.1$. Não é difícil checar que

$L(G)$ é codiagnosticável em relação à P_{o_i} e Σ_f , $i = 1, 2$. A partir disso, é necessário agora verificar se $TNetSys$ permanece codiagnosticável na presença de atrasos na comunicação e perdas intermitentes de observação.

O autômato G_1 , mostrado na figura 4, é computado de acordo com o algoritmo 1 da seguinte maneira. No passo 1, define-se o estado inicial de G_1 como o par $(x_0, 0)$, em que x_0 é o estado inicial de G . No passo 2, o conjunto de eventos $\Sigma_{o_{11}}^s = \{b_{s_1}\}$, $\Sigma_{o_{12}}^s = \{c_{s_1}\}$, $\Sigma_{o_1}^s = \{b_{s_1}, c_{s_1}\}$ e $\Sigma_1 = \{\sigma_f, a, b, c, b_{s_1}, c_{s_1}\}$ são criados. Passo 3 forma a fila $F = [(x_0, 0)]$ e, no passo 4, são estabelecidas novas transições e estados. No passo 4, obtém-se $(x, q) = (x_0, 0)$, e faz-se $F = []$ e $X_1 = (x_0, 0)$ nos passos 4.1 e 4.2. No passo 4.3, são criados os conjuntos de índices I_{o_1} que gravam os índices dos eventos observáveis dentro de q . Sendo assim, $I_{o_1} = \emptyset$, uma vez que $q = 0$. No passo 4.4, computa-se para todo $y \in I_{o_1}$, o tempo mínimo decorrido após a ocorrência do evento q_y na planta, que é denotado como $minst(y)$. Uma vez que $q = 0$, não é necessário computar o passo 4.4 na primeira iteração. No passo 4.5, são definidas duas transições do estado $(x_0, 0)$ rotuladas por eventos σ_f e b , uma vez $\Gamma(x_0) = \{\sigma_f, b\}$ e $q = 0$, e os estados alcançados por essas transições são, respectivamente, $(x_1, 0)$ e (x_5, b) . Então, adicionam-se esses estados à fila F , que se torna $F = [(x_1, 0), (x_5, b)]$. Note que, o evento b (resp. σ_f) é adicionado (resp. não adicionado) à segunda componente do estado alcançado (x_5, b) (resp. $(x_1, 0)$) porque é um evento observável (resp. não observável). Como $q = 0$, nenhuma transição é definida no passo 4.6. As iterações são executadas até que F se torne vazia. Note que na sequência $s = \sigma_f a c b b_{s_1} c_{s_1} \in L(G_1)$ ocorre uma mudança na ordem de observação do diagnosticador local DL_1 . A planta gera os eventos c e b , nesta ordem, entretanto, o diagnosticador local DL_1 observa o evento b_{s_1} antes do evento c_{s_1} .

O próximo passo na verificação da codiagnosticabilidade em rede é a computação, de acordo com o algoritmo 1, do autômato G_2 , cujo diagrama de transição de estado correspondente foi omitido devido à falta de espaço. Observe que, o conjunto de eventos observáveis e não observáveis do diagnosticador local DL_1 são $\Sigma_{1_o} = \{b_{s_1}, c_{s_1}\}$ e $\Sigma_{1_{uo}} = \{a, b, c, \sigma_f\}$, respectivamente, e o conjunto de eventos observáveis e não observáveis do diagnosticador local DL_2 são $\Sigma_{2_o} = \{a_{s_2}, c_{s_2}\}$ e $\Sigma_{2_{uo}} = \Sigma_{1_{uo}} = \{a, b, c, \sigma_f\}$, respectivamente.

Suponha que o evento b esteja sujeito à perda intermitente de observação apenas pelo diagnosticador local DL_1 , e o evento a esteja sujeito à perda intermitente de observação apenas pelo diagnosticador local DL_2 . Sendo assim, para DL_1 , $\Sigma_{1,ilo} = \{b\}$ e $\Sigma_{1,nilo} = \{c\}$, para DL_2 , $\Sigma_{2,ilo} = \{a\}$ e $\Sigma_{2,nilo} = \{c\}$. Os autômatos G'_1 e G'_2 que modelam o atraso de comunicação e a perda in-

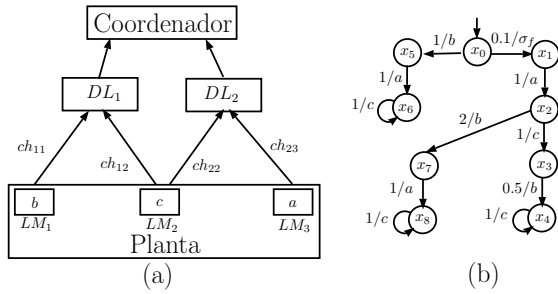


Figura 3: $TNetSys = (G, t_{min}, T)$ para o exemplo 2: estrutura com atraso de comunicação (a) e autômato G com a função de tempo mínimo t_{min} (b).

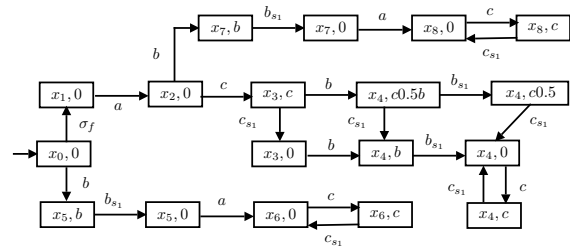


Figura 4: Autômato G_1 do exemplo 2.

termite na observação dos eventos em $\Sigma_{1,ilo}$ e $\Sigma_{2,ilo}$, respectivamente, são computados de acordo com a equação (5). Aplica-se, então o algoritmo 2 para computar V_1 e V_2 . Após computar V_1 e V_2 , pode-se obter $G_V = V_1 || V_2$. É mostrado na figura 5 apenas o caminho de G_V que contém o caminho cíclico $cl = (cc_{s_2}c_{\rho_2}c_{\rho_1}c_{s_1})^l$, em que $l \in \mathbb{N}$. Portanto, linguagem L não é codiagnosticável em rede com relação à $\chi_i, D_{s_i}, P'_{s_i}, i = 1, 2, \dots, n$ e Σ_f .

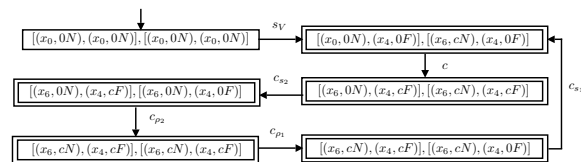


Figura 5: Caminho de G_V com caminho cíclico cl do exemplo 2.

6 Conclusão

Neste artigo, é abordado o problema da codiagnosticabilidade em SEDs com redes sujeitas a atrasos temporais de comunicação e perdas de observação de eventos. Ao invés de medir o atraso na comunicação por passos, considera-se as características temporais do comportamento dinâmico da planta para melhor representar os efeitos dos atrasos na comunicação. É apresentada uma condição necessária e suficiente para a codiagnosticabilidade em rede da linguagem gerada pelo sistema em relação aos atrasos de comunicação e perdas intermitentes e é proposto um algoritmo para verificar essa propriedade.

Referências

- Brandin, B. A. e Wonham, W. M. (1994). Supervisory control of timed discrete-event systems, *IEEE Transactions on Automatic Control* **39**(2): 329–342.
- Carvalho, L. K., Basilio, J. C. e Moreira, M. V. (2012). Robust diagnosis of discrete event systems against intermittent loss of observations, *Automatica* **48**(9): 2068–2078.
- Cassandras, C. G. e Lafortune, S. (2008). *Introduction to Discrete Events Systems*, 2nd edn, Springer, New York, NY : USA.
- Debouk, R., Lafortune, S. e Teneketzis, D. (2000a). Coordinated decentralized protocols for failure diagnosis of discrete event systems, *Discrete Event Dynamic Systems: Theory and Applications* **10**(1): 33–86.
- Debouk, R., Lafortune, S. e Teneketzis, D. (2000b). On the effect of communication delays in failure diagnosis of decentralized discrete event systems, *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, Vol. 3, IEEE, pp. 2245–2251.
- Lin, F. (2014). Control of networked discrete event systems: dealing with communication delays and losses, *SIAM Journal on Control and Optimization* **52**(2): 1276–1298.
- Moreira, M. V., Jesus, T. C. e Basilio, J. C. (2011). Polynomial time verification of decentralized diagnosability of discrete event systems, *IEEE Transactions on Automatic Control* **56**(7): 1679–1684.
- Nunes, C. E. V., Moreira, M. V. e Alves, M. V. S. (2016). Network codiagnosability of discrete-event systems subject to event communication delays, *13th International Workshop on Discrete Event Systems (WODES)*, Xian, China, pp. 217–223.
- Qiu, W. e Kumar, R. (2008). Distributed diagnosis under bounded-delay communication of immediately forwarded local observations, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* **38**(3): 628–643.
- Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K. e Teneketzis, D. (1995). Diagnosability of discrete-event systems, *IEEE Transactions on Automatic Control* **40**(9): 1555–1575.
- Shu, S. e Lin, F. (2014). Decentralized control of networked discrete event systems with communication delays, *Automatica* **50**(8): 2108 – 2112.