

SÍNTESE DE SUPERVISORES DE SISTEMAS A EVENTOS DISCRETOS TEMPORIZADOS

EDUARD MONTGOMERY MEIRA COSTA E ANTONIO MARCUS NOGUEIRA LIMA

*Departamento de Engenharia Elétrica
Universidade Federal da Paraíba
58109-970 Campina Grande, PB, Brasil, Caixa Postal 10.105
Email: {eduard,amnlima}@dee.ufpb.br*

Resumo— Este trabalho utiliza a álgebra de dióides para construção de supervisores para SEDs temporizados. A utilização dessa álgebra dispensa a execução do algoritmo de ponto fixo usado na determinação da suprema sublinguagem controlável. As linguagens utilizadas para descrever e especificar o comportamento do sistema são substituídas por matrizes de incidência temporizadas construídas a partir dos eventos controláveis, dos eventos não controláveis e da especificação de comportamento. O supervisor é sintetizado através de operações matriciais reconhecendo palavras através de uma série formal. O procedimento de síntese proposto produz o mesmo resultado que seria obtido com o procedimento clássico, quando as temporizações são binárias.

Abstract— This paper utilizes the dioid algebra to synthesize timed DES supervisors. The use of this algebra avoids the use of the fix point algorithm used to determine the supremal controllable sublanguage. The timed languages used to describe and to specify the DES behaviour are replaced by timed incidence matrices constructed from the controllable and uncontrollable events, the delays and the behaviour specification. The supervisor is synthesized through matricial and the words being recognized are defined through formal series. The proposed synthesis procedure yields the same results of the classical procedure when the delays are binary.

Key Words— Max Plus Algebra, Timed Discrete Event Systems, Supervisory Control

1 INTRODUÇÃO

Sistemas a Eventos Discretos (SEDs) (Ramadge e Wonham, 1982) são sistemas que apresentam uma evolução dinâmica descrita pela ocorrência de eventos que determinam sua interação com o ambiente e que alteram o estado do sistema. Os SEDs estão presentes em muitas aplicações do cotidiano, como redes de computadores, sistemas de manufatura, supervisão de tráfego aéreo e ferroviário e sistemas operacionais. O estudo dos SEDs requer a utilização de uma representação adequada e que permita projetar um agente de controle automático, denominado de supervisor. A partir de tarefas especificadas para o sistema, o supervisor recebe informações dos eventos do sistema através de sensores, determina a ação de controle e envia comandos para os atuadores que inibem ou habilitam determinados eventos.

De modo geral, a análise e o projeto de SEDs começa pelo estudo do comportamento lógico. Nessa primeira etapa, na qual não se consideram explicitamente os tempos associados aos eventos, o problema de controle é resolvido diretamente com a Teoria de Controle Supervisório (Ramadge e Wonham, 1989). Entretanto, em certas aplicações, é necessário representar explicitamente os tempos associados aos eventos para garantir que a solução do problema de controle seja factível mesmo considerando-se as restrições temporais. Além disso, esses tempos associados aos eventos podem variar durante a operação do sistema e, nesses casos, é necessário modificar a estratégia de controle do SED para preservar o comportamento desejado e não permitir a degradação do desempenho. Desse modo, a inclusão de uma representação temporal explícita nos modelos de SEDs abre a possibilidade de projetar supervisores que sejam adaptáveis a mudanças do sistema e de determinar o melhor desempenho na realização das tarefas especificadas.

Há várias alternativas para incluir uma representação temporal no estudo de SEDs e para realizar a síntese de supervisores de SEDs temporizados. A utilização de tempori-

zadores externos ao autômato e a definição do evento ‘tick’ sincronizado a um relógio global é tratada em (Brandin e Wonham, 1994). Por outro lado, a inclusão da representação do tempo dentro do próprio modelo é tratada em (Cofer e Garg, 1994; Cofer e Garg, 1995) quando o sistema é representado através de um grafo de eventos temporizados. No primeiro caso, há o problema do crescimento no número de estados devido a inclusão do evento ‘tick’. O segundo caso, requer o uso da álgebra de dióides (Gaubert, 1992), e tem sido muito utilizado, especialmente para os SEDs que apresentam periodicidade e necessitam de sincronização (Cohen et al., 1985; Baccelli et al., 1992). Esta álgebra é utilizada na avaliação de desempenho de SEDs (Gaubert, 1993; Gaubert, 1995), utilizando séries formais (Berstel e Reutenauer, 1988; Gaubert, 1994). Devido a sua versatilidade, ela também é utilizada para determinar caminhos em SEDs, utilizando a matriz de incidência do grafo que representa o sistema (Gill, 1962; Gaubert, 1999).

Nesse trabalho é apresentada uma formulação alternativa para a síntese de supervisores para SEDs temporizados, que é restrita ao caso no qual a especificação de comportamento é um sub-autômato do modelo do SED. Essa formulação é baseada na álgebra de dióides e o supervisor é sintetizado através de operações matriciais que dispensam a utilização do algoritmo de ponto fixo apresentado em (Cofer e Garg, 1996) e em (Brandin e Wonham, 1994). A descrição do SED e também sua especificação de comportamento temporizada é feita através de matrizes de incidência temporizadas.

A estrutura desse trabalho é a seguinte: na Seção 2 são apresentados os conceitos das ferramentas necessárias a esta abordagem; na Seção 3 são apresentados os conceitos que dão as bases para a formalização do trabalho; na Seção 4 é apresentado o algoritmo de síntese do supervisor; na Seção 5 é apresentado um exemplo ilustrativo e na Seção 6 são apresentadas as conclusões.

2 PRELIMINARES

Um dióide é definido como:

Definição 1 Um dióide é um conjunto D dotado de duas operações: \oplus (soma) e \otimes (multiplicação), que satisfaz os seguintes axiomas:

Axioma 1: Comutatividade de \oplus : $a \oplus b = b \oplus a$

Axioma 2: Associatividade de \oplus : $(a \oplus b) \oplus c = a \oplus (b \oplus c)$

Axioma 3: Associatividade de \otimes : $(a \otimes b) \otimes c = a \otimes (b \otimes c)$

Axioma 4: Distributividade de \otimes sobre \oplus :

$$\begin{aligned} (a \oplus b) \otimes c &= (a \otimes c) \oplus (b \otimes c) \\ c \otimes (a \oplus b) &= (c \otimes a) \oplus (c \otimes b) \end{aligned}$$

Axioma 5: Elemento nulo em \oplus : $a \oplus \epsilon = a$, $\forall a \in D$ e algum $\epsilon \in D$

Axioma 6: Absorção pelo elemento nulo em \otimes : $a \otimes \epsilon = \epsilon$

Axioma 7: Elemento identidade em \otimes : $a \otimes e = a$

Axioma 8: Idempotência em \oplus : $a \oplus a = a$.

Um dióide é dito ser comutativo se \otimes é comutativo.

2.1 AUTÔMATOS TEMPORIZADOS

O autômato $(\max, +)$ é uma ferramenta que pode ser utilizada na modelagem de SEDs temporizados. A evolução dinâmica desse tipo autômato é simplificada utilizando séries formais aplicadas aos seus datadores. Os datadores definem os tempos em que ocorrem as transições de estado. A álgebra de dióides pode ser aplicada nesse caso definindo-se $D = \mathbb{R}_{\max} = \mathbb{R} \cup -\infty$, $\epsilon = -\infty$ e $e = 0$. O operador \oplus é definido como a operação \max e o operador \otimes é a soma usual $+$.

Definição 2 Um autômato finito $A_{(\max, +)}$ sobre um alfabeto Σ é uma quádrupla $A_{(\max, +)} = (Q, \theta, T, \phi)$ onde Q é um conjunto finito de estados e θ, T, ϕ são mapas $\theta: Q \rightarrow \mathbb{R}_{\max}$, $T: Q \times \Sigma \times Q \rightarrow \mathbb{R}_{\max}$, $\phi: Q \rightarrow \mathbb{R}_{\max}$, denominados atraso inicial, tempo de transição e atraso final, respectivamente.

Nesse autômato um caminho de comprimento n é uma seqüência de estados $p = (q_1, \dots, q_{n+1}) \in Q^{n+1}$, onde uma palavra $s = \sigma_1 \dots \sigma_n$ é reconhecida em p se $P(p, s) := \theta(q_1) \otimes T_{q_1, \sigma_1, q_2} \otimes \dots \otimes T_{q_n, \sigma_n, q_{n+1}} \otimes \phi(q_{n+1}) \neq \epsilon$, com P sendo a função peso do caminho p . Ou seja, uma palavra é reconhecida se ela é completada num tempo finito. A multiplicidade, ou o número de caminhos que aceitam uma palavra $s = \sigma_1 \dots \sigma_n$ é o máximo entre os pesos dos caminhos que reconhecem s , isto é,

$$(A_{(\max, +)} | s) := \bigoplus_{p \in Q^{n+1}} P(p, s) \quad (1)$$

ou

$$(A_{(\max, +)} | s) := \max_{q_1 \dots q_{n+1}} [\theta(q_1) + T_{q_1, \sigma_1, q_2} + \dots + \phi(q_{n+1})]. \quad (2)$$

Dessa forma, o autômato reconhece o datador da palavra $s \mapsto (A_{(\max, +)} | s)$. Em outros termos, um datador $y: \Sigma^* \rightarrow \mathbb{R}_{\max}$ é dito *reconhecível* se existe um autômato $A_{(\max, +)}$ tal que $(y | s) = (A_{(\max, +)} | s)$.

A representação gráfica de $A_{(\max, +)}$ é um multigrafo valorado, definido por vértices construídos do conjunto de

estados Q e por três tipos de arcos: i) Os arcos internos, $i \xrightarrow{\sigma} j$, $\forall i, j \in Q$ e $\sigma \in \Sigma$ tal que $T_{i, \sigma, j} \neq \epsilon$. O arco $i \xrightarrow{\sigma} j$, é valorado pelo escalar $T_{i, \sigma, j}$; ii) Os arcos de entrada $\rightarrow i$, valorados por $\theta(i)$, $\forall i \in Q$ tal que $\theta(i) \neq \epsilon$; iii) Os arcos de saída $i \rightarrow$, valorados por $\phi(i)$, $\forall i \in Q$ tal que $\phi(i) \neq \epsilon$. Logo, $(A_{(\max, +)} | s)$ lê o máximo entre as somas de todos os pesos de um caminho etiquetado s da entrada para a saída do grafo associado à $A_{(\max, +)}$.

Exemplo 1 Seja $\Sigma = \{\alpha, \beta\}$. O autômato com conjunto de estados $Q = \{0, 1, 2\}$, tempos de transições $T(0, \alpha, 1) = 1$, $T(0, \alpha, 2) = 3$, $T(1, \alpha, 2) = 4$, $T(2, \beta, 2) = 1$, $T(2, \beta, 1) = 5$, $T(2, \beta, 0) = 7$, $T(1, \beta, 1) = 1$, $T(1, \beta, 0) = 2$, atrasos final e inicial $\phi(0) = 2$ e $\theta(0) = 0$, respectivamente (os outros valores de ϕ, T, θ são ϵ) está representado na Figura 1. Os valores

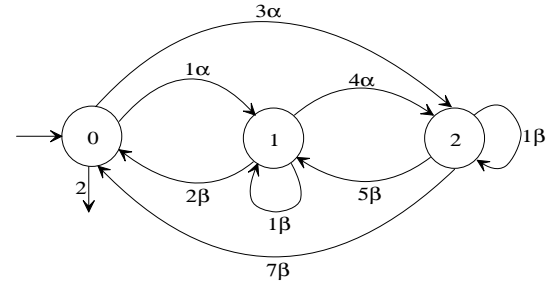


Figura 1. Autômato $(\max, +)$.

iguais a 0 são omitidos, isto é, o arco de entrada $\rightarrow 0$, não valorado, define $\theta(0) = 0$. Assim, encontra-se que $s = \alpha\beta$ é uma palavra reconhecida, pois

$$\begin{aligned} P(0 \xrightarrow{\alpha} 1 \xrightarrow{\beta} 0) &= \theta(0) + T_{0, \alpha, 1} + T_{1, \beta, 0} + \phi(0) = 5 \\ P(0 \xrightarrow{\alpha} 2 \xrightarrow{\beta} 0) &= \theta(0) + T_{0, \alpha, 2} + T_{2, \beta, 0} + \phi(0) = 12 \\ (A_{(\max, +)} | \alpha\beta) &= \max(P(0 \xrightarrow{\alpha} 1 \xrightarrow{\beta} 0), P(0 \xrightarrow{\alpha} 2 \xrightarrow{\beta} 0)) \\ (A_{(\max, +)} | \alpha\beta) &= \max(5, 12) = 12. \end{aligned}$$

As séries formais podem ser utilizadas para a descrição da evolução dinâmica desse tipo de autômato e determinar seus estados reconhecidos. Assim, a linguagem reconhecida por $A_{(\max, +)}$ pode ser descrita através de seus datadores como

$$y = \bigoplus_{s \in \Sigma^*} (y | s) \otimes s \in \mathbb{R}_{\max} \langle\langle \Sigma \rangle\rangle \quad (3)$$

onde $\mathbb{R}_{\max} \langle\langle \Sigma \rangle\rangle$ é o semi-anel das séries formais com variáveis não comutativas em Σ sobre \mathbb{R}_{\max} . Neste contexto, $(y | s)$ denota o coeficiente da palavra s , onde $(y | s) = \epsilon$ se $s \notin L$ e $(y | s) = e$ se $s \in L$. Geralmente, o operador \otimes é omitido. Assim, a notação ab significando $a \otimes b$ será utilizada daqui por diante, para multiplicações entre quaisquer elementos na álgebra de dióides.

Sendo o mapa $\mu: \Sigma \rightarrow \mathbb{R}_{\max}^{Q \times Q}$, onde $\mathbb{R}_{\max}^{Q \times Q}$ é o dióide \mathbb{R}_{\max} sobre matrizes quadradas de dimensão $|Q|$, define-se a matriz $\mu(\sigma)_{qq'} := T_{q, \sigma, q'}$. Identificando θ com um vetor linha contendo os arcos de entrada e ϕ com um vetor coluna contendo os arcos de saída, tem-se que

$$(A_{(\max, +)} | \sigma_1 \dots \sigma_n) = \theta \mu(\sigma_1) \dots \mu(\sigma_n) \phi. \quad (4)$$

Logo, $(A_{(\max, +)} | s) = \theta \mu(s) \phi$, é a descrição do comportamento do autômato, onde $\mu(s)$ é a matriz que contém os tempos de complementação das palavras formadas por $\sigma_1, \dots, \sigma_n \in \Sigma$. Por outro lado, identificando o mapa datador

$$y: \Sigma^* \rightarrow \mathbb{R}_{\max}, s \mapsto (y | s) \quad (5)$$

com a série formal, diz-se que a série y é reconhecível se existe um autômato finito (θ, μ, ϕ) , equivalente a $A_{(\max,+)}$, com $\theta \in \mathbb{R}_{\max}^{1 \times Q}$, $\phi \in \mathbb{R}_{\max}^{Q \times 1}$, $\mu: \Sigma^* \rightarrow \mathbb{R}_{\max}^{Q \times Q}$ e Q finito, tal que $y = \bigoplus_{s \in \Sigma^*} \theta \mu(s) \phi s$.

Exemplo 2 O autômato da Figura 2, tem

$$\theta = [e \ e \ e] \quad \phi = \begin{bmatrix} 2 \\ \epsilon \\ \epsilon \end{bmatrix}$$

$$\mu(\alpha) = \begin{bmatrix} \epsilon & 1 & \epsilon \\ \epsilon & \epsilon & 4 \\ \epsilon & \epsilon & \epsilon \end{bmatrix} \quad \mu(\beta) = \begin{bmatrix} \epsilon & \epsilon & \epsilon \\ 2 & \epsilon & \epsilon \\ \epsilon & 5 & \epsilon \end{bmatrix}.$$

Calculando $(y|s)$ para um caso particular, onde $s = \alpha\beta$, tem-se que $(y|\alpha\beta) = \theta \mu(\alpha\beta) \phi = \theta \mu(\alpha) \mu(\beta) \phi = 5$. Para este autômato, a série

$$y = \bigoplus_{s \in \Sigma^*} (y|s) \otimes s \in \mathbb{R}_{\max} \langle\langle \Sigma \rangle\rangle$$

admite a seguinte expressão

$$y = 2(3\alpha(9\alpha\beta)^*\beta)^* = 2 \oplus 5\alpha\beta \oplus 8\alpha\beta\alpha\beta \oplus 14\alpha\alpha\beta\beta \oplus \dots$$

Desse modo, y apresenta todas as palavras formadas por α e β , iniciadas por α , e reconhecidas pelo autômato da Figura 2, com seus respectivos tempos de complementação.

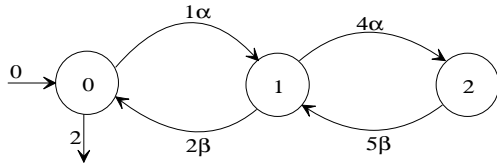


Figura 2. Autômato $(\max,+)$ determinístico.

REPRESENTAÇÃO POR MATRIZ DE INCIDÊNCIA

A matriz de incidência de um autômato $(\max,+)$, difere da matriz de um autômato não temporizado apenas pela introdução do tempo de vida do evento:

Definição 3 A um autômato $A_{(\max,+)}$ com N estados associa-se uma matriz $A_{N \times N}$, denominada de matriz de incidência temporizada, onde

$$A = [a_{ij}], a_{ij} = \begin{cases} t_\sigma \sigma & \text{se } \exists \sigma \text{ do estado } i \text{ para o estado } j; \\ \epsilon & \text{caso contrário,} \end{cases}$$

e $\sigma \in \Sigma$ representa os caminhos de comprimento 1, que provocam a transição do estado i para o estado j , e t_σ representa o tempo de vida do evento σ .

Exemplo 3 A matriz de incidência A associada ao autômato da Figura 3, é dada por

$$A = \begin{bmatrix} \epsilon & 3\alpha & 3\beta \\ 4\eta & \epsilon & \epsilon \\ \epsilon & 2\beta & 2\alpha + \lambda \end{bmatrix}.$$

3 A SÍNTESE DO SUPERVISOR

Aqui é apresentada a maneira de sintetizar um supervisor para um SED temporizado, utilizando as matrizes de incidência do autômato que modela o SED, admitindo que o alfabeto de eventos é $\Sigma = \Sigma_c \cup \Sigma_{uc}$, $\Sigma_c \cap \Sigma_{uc} = \emptyset$.

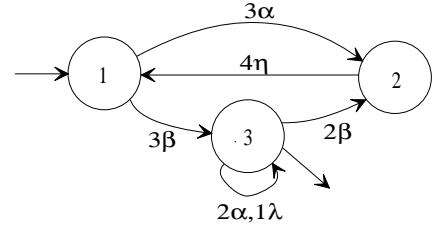


Figura 3. Autômato temporizado $(\max,+)$.

3.1 EVENTOS NA MATRIZ DE INCIDÊNCIA

Considerando $\Sigma = \Sigma_c \cup \Sigma_{uc}$, definem-se as matrizes A_c e A_{uc} , referentes às partes controláveis e não controláveis do autômato que modela o SED.

Definição 4 A um autômato $A_{(\max,+)}$ definido para um alfabeto $\Sigma = \Sigma_c \cup \Sigma_{uc}$, associam-se as matrizes A_c e A_{uc} , com dimensões iguais a A , dadas por

$$A_c = [(a_c)_{ij}], (a_c)_{ij} = \begin{cases} t_{\sigma_c} & \text{se } \exists \sigma_c \text{ do estado } i \text{ para } j; \\ \epsilon & \text{caso contrário} \end{cases}$$

$$A_{uc} = [(a_{uc})_{ij}], [a_{uc}]_{ij} = \begin{cases} t_{\sigma_{uc}} & \text{se } \exists \sigma_{uc} \text{ do estado } i \text{ para } j; \\ \epsilon & \text{caso contrário,} \end{cases}$$

onde $\sigma_c \in \Sigma_c$, $\sigma_{uc} \in \Sigma_{uc}$ e t representa o tempo de vida do respectivo evento.

Exemplo 4 Considerando que o autômato da Figura 3, tem $\Sigma_c = \{\alpha, \eta\}$ e $\Sigma_{uc} = \{\beta, \lambda\}$, então

$$A_c = \begin{bmatrix} \epsilon & 3\alpha & \epsilon \\ 4\eta & \epsilon & \epsilon \\ \epsilon & \epsilon & 2\alpha \end{bmatrix} \quad e \quad A_{uc} = \begin{bmatrix} \epsilon & \epsilon & 3\beta \\ \epsilon & \epsilon & \epsilon \\ \epsilon & 2\beta & \lambda \end{bmatrix}.$$

Nesta representação, quando $t=0$, omite-se na matriz. Observe que $A = A_c \oplus A_{uc}$.

3.2 A CONSTRUÇÃO DO SUPERVISOR

O algoritmo da construção da $SupC(L)$ para SEDs não temporizados (Ramadge e Wonham, 1987) e para SEDs temporizados (Brandin e Wonham, 1994), provê uma forma de construir um supervisor para um SED. Dadas uma linguagem e uma especificação de comportamento, sua execução determina quais os eventos controláveis que devem ser inibidos para evitar seqüências que não pertencem à especificação. Por outro lado, em (Cofer e Garg, 1996) é visto que este controle é feito através da inibição por tempo determinado dos eventos controláveis, de forma ao SED seguir uma seqüência determinada. Aqui, a síntese do supervisor é feita através de operações matriciais, onde o supervisor é especificado por uma matriz de incidência S , que inibe arcos controláveis por tempo indeterminado, definindo a mesma função lógica do supervisor na TCS, além de poder inibir eventos controláveis por tempo determinado, gerando atrasos em algumas transições do sistema, semelhantemente a abordagem de Cofer e Garg (Cofer e Garg, 1996).

Definição 5 Dada uma especificação de comportamento E para um autômato $A_{(\max,+)}$ que é um modelo de um SED, sua matriz de incidência é definida por

$$E = [e_{ij}], e_{ij} = \begin{cases} t_\sigma \sigma & \text{se } \exists \sigma \text{ do estado } i \text{ para o estado } j \text{ em } E; \\ \epsilon & \text{caso contrário} \end{cases}$$

com dimensão $N \times N$ e, $\sigma \in \Sigma$ representa os eventos que provocam a mudança do estado i para o estado j , e t_σ representa o tempo de vida desejado do respectivo evento.

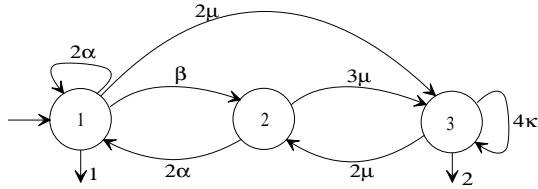


Figura 4. Autômato do Exemplo 7.

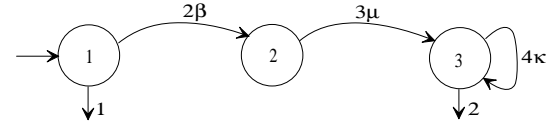


Figura 5. Supervisor do Exemplo 7.

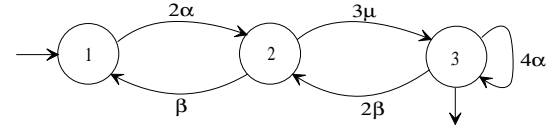


Figura 6. Autômato do Exemplo 8.

Considerando que os tempos de vida dos eventos não podem ser reduzidos, mas apenas atrasados, então:

Definição 6 Uma especificação de comportamento E dada a um SED modelado por um autômato $A_{(\max,+)}$, é válida se satisfizer a seguinte condição:

$$t(e_{ij}) = \begin{cases} t(e_{ij}) \oplus t(a_{ij}) & \text{se } e_{ij} \neq \epsilon \wedge a_{ij} \in \Sigma_c \\ t(a_{ij}) & \text{se } e_{ij} \neq \epsilon \wedge a_{ij} \in \Sigma_{uc} \end{cases}$$

onde $t(e_{ij})$ e $t(a_{ij})$ são os tempos de vida dos eventos nos elementos e_{ij} e a_{ij} , respectivamente.

Disso, determina-se se a especificação de comportamento E é solução não restritiva para o problema de controle do SED. Para isto, o que segue é necessário:

Definição 7 Seja (D, \oplus, \otimes) um dióide. Para $D = \mathbb{R}_{\max} \langle \langle \Sigma \rangle \rangle$, isto é, D é um conjunto de linguagens temporizadas, então, a operação \oplus é definida por $t_\sigma \sigma \oplus t'_\sigma \sigma' = t_\sigma \sigma + t'_\sigma \sigma'$, como definido para as expressões regulares, significando a união das linguagens temporizadas, e a operação \otimes é definida por $t_\sigma \sigma \otimes t'_\sigma \sigma' = (t_\sigma + t'_\sigma) \sigma \sigma'$, representando a concatenação entre linguagens temporizadas, onde o tempo total é a soma dos tempos individuais.

Exemplo 5 Sejam $L_1 = \{2\alpha\beta\}$ e $L_2 = \{3\kappa\}$. Então $L_1 \oplus L_2 = (2\alpha\beta \oplus 3\kappa) = (2\alpha\beta + 3\kappa) = \{2\alpha\beta, 3\kappa\}$ e $L_1 \otimes L_2 = \{5\alpha\beta\kappa\}$.

Definição 8 Seja A uma matriz de incidência de um autômato $A_{(\max,+)}$. O operador $\Upsilon: A_{(\max,+)} \rightarrow \mathbb{N}$, define o número de arcos em $A_{(\max,+)}$, que compõem os termos da matriz A .

Exemplo 6 No autômato da Figura 3, $\Upsilon(A) = 6$, que é o número de arcos do autômato representado por A . Neste mesmo autômato, $\Upsilon(A_c) = 3$ e $\Upsilon(A_{uc}) = 3$.

Assim, com este operador, tem-se:

Lema 1 Um supervisor S para um autômato $A_{(\max,+)}$ trim é definido pela especificação de comportamento E trim se e só se $\Upsilon(E_{uc}) = \Upsilon(A_{uc})$.

Corolário 1 Dada uma especificação E trim e a matriz de incidência A_{uc} de um autômato G trim, $S = E \oplus A_{uc}$ se e só se $\Upsilon(E_{uc}) = \Upsilon(A_{uc})$.

Logo, o Lema 1 sendo satisfeito, o supervisor é definido por E .

Exemplo 7 Do autômato visto na Figura 4, onde $\Sigma = \{\alpha, \beta, \kappa, \mu\}$, $\Sigma_{uc} = \{\kappa\}$ e $\Sigma_c = \{\alpha, \beta, \mu\}$, as matrizes de incidência A , A_{uc} e E são dadas por

$$A = \begin{bmatrix} 2\alpha & \beta & 2\mu \\ 2\alpha & \epsilon & 3\mu \\ \epsilon & 2\mu & 4\kappa \end{bmatrix}, \quad A_{uc} = \begin{bmatrix} \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & 4\kappa \end{bmatrix},$$

$$E = \begin{bmatrix} \epsilon & 2\beta & \epsilon \\ \epsilon & \epsilon & 3\mu \\ \epsilon & \epsilon & 4\kappa \end{bmatrix},$$

Para a especificação dada, as condições da Definição 6, são satisfeitas, e $S = E \oplus A_{uc} = E$, pois $\Upsilon(E_{uc}) = \Upsilon(A_{uc}) = 1$, que satisfaz o Lema 1. Este supervisor é visto na Figura 5.

Quando o Lema 1 não é satisfeito, é necessário encontrar a $SupC(L)$ para E . Para isto, é necessário o que se segue:

Definição 9 Seja A uma matriz de incidência, cujos elementos a_{ij} definem caminhos de comprimento 1. Então, define-se a matriz $A^n = A \otimes A \otimes \dots \otimes A$ (n vezes) como sendo uma matriz de caminhos, onde a_{ij}^n representa caminhos de comprimento n , que são percorridos no autômato representado por A , com um tempo total t_{ij} . Esses caminhos são formados por eventos controláveis e não controláveis, que levam o autômato do estado i para o estado j . Se não há um caminho com n eventos de i para j , $a_{ij} = \epsilon$.

Exemplo 8 Do autômato mostrado na Figura 6, tem-se

$$A = \begin{bmatrix} \epsilon & 2\alpha & \epsilon \\ \beta & \epsilon & 3\mu \\ \epsilon & 2\beta & 4\alpha \end{bmatrix}.$$

A matriz de caminhos de comprimento 2 é

$$A^2 = A \otimes A = \begin{bmatrix} 3\alpha\beta & \epsilon & 5\alpha\mu \\ \epsilon & 5\mu\beta + 3\beta\alpha & 7\mu\alpha \\ 3\beta\beta & 6\alpha\beta & 8\alpha\alpha + 5\beta\mu \end{bmatrix}.$$

Usando a Definição 9, pode-se avaliar quais os caminhos que iniciam com um evento qualquer, seguidos só por eventos não controláveis.

Definição 10 Seja $\Sigma = \Sigma_c \cup \Sigma_{uc}$, e seja um autômato G construído com símbolos de Σ . A matriz de caminhos C_{uc}^n , que define as palavras que mudam o estado do autômato de i para j , iniciados por um evento qualquer (σ_c ou σ_{uc}), e seguidos sempre de eventos não controláveis, é definida por $C_{uc}^n = A \otimes (A_{uc})^{n-1}$.

Assim, os primeiros eventos das seqüências dos termos de C_{uc}^n são elementos de A (σ_{uc} ou σ_c) e os demais são σ_{uc} . Disso, utiliza-se a mesma idéia para definir caminhos em um dado autômato que iniciem com eventos de E , seguidos de σ_{uc} , e determinar se é possível sintetizar um supervisor para a $SupC(L)$.

Definição 11 Dada a matriz de incidência E , que é uma matriz de caminhos de comprimento 1, define-se $B_{uc}^n = E \otimes (A_{uc})^{n-1}$, como sendo uma matriz de caminhos, onde o primeiro elemento de cada seqüência é um elemento de E , e os demais, de A_{uc} .

Com esta definição, pode-se avaliar se é possível inibir alguma seqüência que leve à ocorrência de $\sigma_{uc} \notin E$. Em B_{uc}^n , um evento $\sigma_{uc} \notin E$ de $A_{uc}(i, j)$, aparece como último evento nas seqüências dos termos de uma coluna j garantindo a condição $\Sigma(\mathbf{H}(x)) \cap \Sigma_{uc}$ (Ramadge e Wonham, 1987; Brandin e Wonham, 1994). Estes eventos são

antecedidos pelos termos de $B_{uc}^{n-1}(k, i)$, para alguns $k=1, \dots, N$. Deve-se observar que $B_{uc}^1 = B_c^1 = E \otimes (A_{uc})^0 = E \otimes I = E$, com I sendo a matriz identidade. Disso, tem-se:

Teorema 1 *Dada a especificação de comportamento E e a matriz de incidência dos eventos não controláveis A_{uc} , do autômato G trim, se $\Upsilon(E_{uc}) \neq \Upsilon(A_{uc})$, então a $SupC(L)$ será determinada recursivamente por:*

1. Para $n = 1, S^1 = E$.
2. Para $n = n + 1$, enquanto $(n \leq N) \wedge \exists \sigma_{uc} \notin E$ então,

$$B_{uc}^n = E \otimes (A_{uc})^{n-1}$$

$$S^n = [s_{ij}^n], s_{ij}^n = \begin{cases} e_{ij} & \text{se } \sigma_{uc}^1 \sigma_{uc}^2 \dots \sigma_{uc}^n \in B_{uc}^n \wedge \sigma_{uc}^n \in E; \\ \epsilon & \text{se } \sigma_{uc}^n \notin E \wedge \sigma^1 \in \Sigma_c \vee e_{ij} \text{ não é alcançável} \end{cases}$$

onde σ_{uc}^n é o n -ésimo evento da seqüência do termo B_{uc}^n , que pode ser $\sigma_{uc}^n \notin E$.

3. Se $(n > N) \wedge (\exists \sigma_{uc} \notin E \text{ em } S^n)$, então $S = [\epsilon]$.

Assim, se o último evento de uma seqüência em B_{uc}^n não pertence à especificação, a inibição do primeiro evento elimina esta seqüência. A existência de outros eventos não pertencentes à especificação, é eliminada recursivamente em B_{uc}^n , para $n=2, 3, \dots, N$, construindo o supervisor. Se algum $\sigma_{uc}^n \notin E$ não for eliminado até $n=N$, o supervisor não é possível para E . As condições da Definição 6 garantem os atrasos aplicados aos eventos controláveis. As mudanças no procedimento para a síntese do supervisor para SEDs temporizados é mínima em relação ao algoritmo de síntese para SEDs não temporizados, sendo incluído apenas as condições de temporização.

4 ALGORITMO DE SÍNTESE

Algoritmo 1 *Construção da $SupC(L)$*

1. Faça $\forall e_{ij}$,

$$t(e_{ij}) = \begin{cases} t(e_{ij}) \oplus t(a_{ij}) & \text{se } e_{ij} \neq \epsilon \wedge a_{ij} \in \Sigma_c \\ t(a_{ij}) & \text{se } e_{ij} \neq \epsilon \wedge a_{ij} \in \Sigma_{uc} \end{cases}$$

onde $t(e_{ij})$ e $t(a_{ij})$ são os tempos de vida dos eventos nos elementos e_{ij} e a_{ij} , respectivamente.

2. Faça $S=E$.

3. Se $\Upsilon(E_{uc})=\Upsilon(A_{uc})$ pare. Caso contrário, faça

a) $n=2$, $xdif(k, n-1)=i$ e $ydif(k, n-1)=j$ (onde $A_{uc} \neq E_{uc}$), para $k=1, \dots, M$ (M número de elementos diferentes entre A_{uc} e E_{uc}).

b) Calcule B_{uc}^n .

c) Faça para $k=1$ até M

i. Procure os elementos em $B_{uc}^n(i, ydif(k, n-1))$, onde $\sigma_{uc}^n \notin E$ (σ_{uc}^n sendo o último elemento da seqüência).

(1) Se $\sigma_{uc}^n \notin E$ e $\sigma^1 \in \Sigma_c$, faça $S(i, xdif(k, n-1))=\epsilon$.

(2) Se $\sigma_{uc}^n \notin E$ e $\sigma^1 \in \Sigma_{uc}$, faça $xdif(k, n)=i$ e $ydif(k', n)=ydif(k, n-1)$, onde k' é o número de elementos não possíveis de controlar em B_{uc}^n .

ii. Se $\forall i, j=1, \dots, N$ (N número de estados), $j=ydif(k, n-1)$ e $S(i, j)=\epsilon$, faça $S(i, j)=\epsilon$.

d) Para $k=1$ até M , faça

i. Se $S(xdif(k, n-1), ydif(k, n-1))=\epsilon$, calcule

$$y = \bigoplus_{s \in \Sigma^*} \theta_{\mu_S}(s) \phi_s$$

(teste de acessibilidade e coacessibilidade).

Caso contrário, faça $n=n+1$.

(1) Se $n < N$, retorne ao passo 2.b.

Esse algoritmo para a síntese do supervisor, apresenta uma ordem de complexidade de $O(N^2)$, onde N é a dimensão da matriz de incidência A , ou o número de estados do autômato. Nele, se $y \neq \epsilon$, o supervisor encontrado é trim (reconhece alguma seqüência). As matrizes $xdif$ e $ydif$ guardam os valores de i e j de E , respectivamente, para os eventos $\sigma_{uc} \notin E$. Sempre que uma seqüência em B_{uc}^n tem o último evento $\sigma_{uc} \notin E$ e o primeiro evento não controlável, estas matrizes são atualizadas com os valores de i e j de B_{uc}^{n-1} . Enquanto $n \leq N$, o algoritmo é repetido, buscando a $SupC(L)$.

5 EXEMPLO

Exemplo 9 *No autômato visto na Figura 7, com $\Sigma=\{\alpha, \beta, \kappa, \eta, \lambda, \mu\}$ e $\Sigma_{uc}=\{\alpha, \kappa, \lambda\}$, a matriz de incidência A_{uc} e a especificação E (vista na Figura 8) são dadas por*

$$A_{uc} = \begin{bmatrix} \epsilon & 2\alpha & \epsilon & \epsilon & \epsilon & \epsilon \\ \lambda & \epsilon & \kappa & \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & \epsilon & \kappa & \epsilon & \epsilon \\ \epsilon & \epsilon & 2\kappa & 3\alpha & \epsilon & \epsilon \\ \epsilon & \epsilon & \epsilon & 2\lambda & \epsilon & \alpha \\ 2\alpha & \epsilon & \epsilon & \epsilon & 2\kappa & \lambda \end{bmatrix} \text{ e}$$

$$E = \begin{bmatrix} \epsilon & 2\alpha & \epsilon & \epsilon & \epsilon & \epsilon \\ \lambda & \epsilon & \kappa & \epsilon & \epsilon & \epsilon \\ \epsilon & \epsilon & \epsilon & \kappa & \epsilon & \epsilon \\ 5\eta & \epsilon & 2\kappa & 3\alpha & 8\beta & \epsilon \\ \epsilon & \epsilon & \epsilon & 2\lambda & \epsilon & \alpha \\ 2\alpha & \epsilon & \epsilon & \epsilon & \epsilon & \lambda \end{bmatrix}.$$

Essa especificação satisfaz as condições da Definição 6. Então, tem-se $S^1=E$. Como $\Upsilon(E_{uc})=10$ e $\Upsilon(A_{uc})=11$, vê-se que $A_{uc}(6, 5)=2\kappa$ não está em E . Assim, calcula-se B_{uc}^2 :

$$E \otimes A_{uc} = \begin{bmatrix} 3\alpha\lambda & \epsilon & 3\alpha\kappa & \epsilon & \epsilon & \epsilon \\ \epsilon & 3\lambda\alpha & \epsilon & 2\kappa\kappa & \epsilon & \epsilon \\ \epsilon & \epsilon & 3\kappa\kappa & 4\kappa\alpha & \epsilon & \epsilon \\ \epsilon & 7\eta\alpha & 5\alpha\kappa & 3\kappa\kappa+6\alpha\alpha+10\beta\lambda & \epsilon & 9\beta\alpha \\ 3\alpha\alpha & \epsilon & \epsilon & \epsilon & 3\alpha\kappa & 2\alpha\lambda \\ 3\lambda\alpha & 4\alpha\alpha & \epsilon & \epsilon & 3\lambda\kappa & 2\lambda\lambda \end{bmatrix}$$

Nessa matriz, o termo (5, 5) contém a seqüência $3\alpha\kappa$, onde κ é o evento $A_{uc}(6, 5) \neq E(6, 5)$ antecedido de $\alpha \in \Sigma_{uc}$, determinando $S^2 = S^1$. Então, no cálculo de B_{uc}^3 , encontram-se os termos (4, 5)= $11\beta\alpha\kappa$, (5, 5)= $4\alpha\lambda\kappa$ e (6, 5)= $4\lambda\lambda\kappa$, onde o primeiro desses termos define a seqüência iniciada pelo evento controlável β e, cuja seqüência leva do estado 4 para o estado 5, através da ocorrência de 3 eventos. Assim, a inibição de β em $E(4, 5)$, define S^3 . Observe que a inibição de β no estado 4, torna os estados 5 e 6 não alcançáveis. Logo, todos os eventos nas respectivas linhas/colunas tornam-se ϵ . Assim, o supervisor, visto na Figura 9, reconhece a série

$$y = \bigoplus_{s \in \Sigma^*} \theta_{\mu_S}(s) \phi_s$$

onde $\theta=[e \ \epsilon \ \epsilon]$ e $\phi=[\epsilon \ \epsilon \ \epsilon]^T$ e s sendo as palavras que levam ao estado 4, é dado por

$$S^3 = \begin{bmatrix} \epsilon & 2\alpha & \epsilon & \epsilon \\ \lambda & \epsilon & \kappa & \epsilon \\ \epsilon & \epsilon & \epsilon & \kappa \\ 5\eta & \epsilon & \kappa & 3\alpha \end{bmatrix}.$$

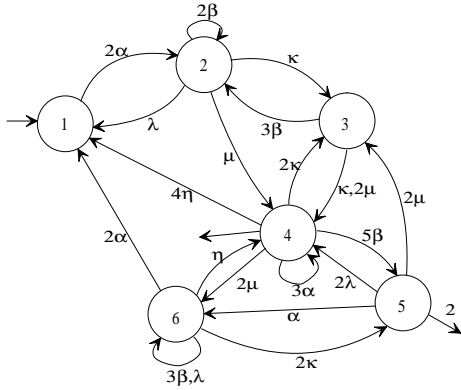


Figura 7. Autômato do Exemplo 9.

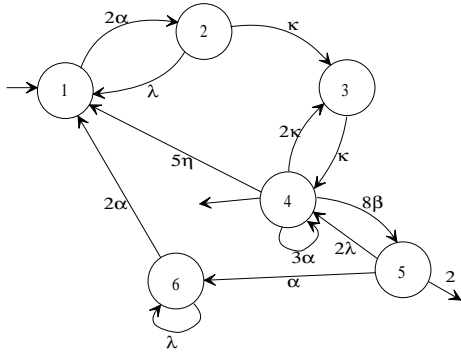


Figura 8. Especificação para o Exemplo 9.

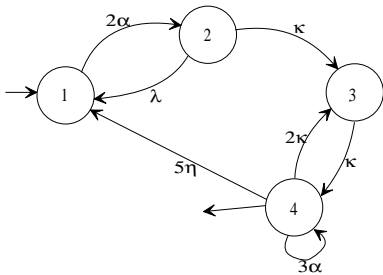


Figura 9. Supervisor do Exemplo 9.

6 CONCLUSÃO

Esse trabalho mostra que é possível sintetizar um supervisor para um SED temporizado, utilizando autômatos $(\max, +)$ e álgebra de dióides. A formulação do problema de controle supervísório no contexto desta álgebra requer que tanto o autômato que modela o SED quanto a especificação de comportamento sejam representados através de matrizes de incidência temporizadas constituídas de elementos definidos no alfabeto de eventos do SED através do semianel $\mathbb{R}_{\max} \langle\langle \Sigma \rangle\rangle$. A solução obtida fornece resultados comparáveis aos da abordagem não temporizada, quando os tempos de vida dos eventos são todos iguais a e , produzindo resultados semelhantes ao algoritmo da $SupC(L)$ não temporizada desenvolvido em (Ramadge e Wonham, 1987) e à abordagem temporizada (Brandin e Wonham, 1994).

Logo, o algoritmo aqui apresentado pode ser utilizado como uma alternativa à construção de supervisores para SEDs não temporizados e temporizados. Esse algoritmo apresenta uma ordem de complexidade de $O(N^2)$, sendo $N-1$ o máximo número de iterações, onde N é a dimensão da matriz de incidência. Embora nesse trabalho a síntese do supervisor esteja limitada ao tratamento de SEDs em que a especificação de comportamento é definida por um sub-autômato do modelo do sistema, uma abordagem mais ampla está sendo fundamentada, a qual inclui testes de bloqueio, controlabilidade e fechamento, além de ser aplicado à especificações não sub-autômatos.

Referências

- Baccelli, F., Cohen, G., Olsder, G. e Quadrat, J. (1992). *Synchronisation and Linearity. An Algebra for Discrete Event Systems*, John Wiley Sons.
- Berstel, J. e Reutenauer, C. (1988). *Rational Series and their Languages*, Springer.
- Brandin, B. e Wonham, W. (1994). Supervisory control of timed discrete-event systems, *IEEE Transactions on Automatic Control* **39**(2): 329–342.
- Cofer, D. e Garg, V. (1994). Supervisory control of timed event graphs, *IEEE - 0-7803-2129-4/94*.
- Cofer, D. e Garg, V. (1995). Control of event separation times in discrete event systems, *Research supported in part by NSF grant CCR-9110605, a General Motors Fellowship, and an IBM grant*.
- Cofer, D. e Garg, V. (1996). Supervisory control of real time discrete-event systems using lattice theory, *IEEE Transactions on Automatic Control* **41**(2): 199–209.
- Cohen, G., Dubois, D., Quadrat, J. e Viot, M. (1985). A linear system theoretic view of discrete event process and its use for performance evaluation in manufacturing, *IEEE Transactions on Automatic Control* **30**(3): 210–220.
- Gaubert, S. (1992). *Théorie des Systèmes Linéaires dans les Dioïdes*, PhD thesis, École Nationale Supérieure des Mines de Paris.
- Gaubert, S. (1993). Performance evaluation of timed automata, *Technical report*, Institut National de Recherche en Informatique et en Automatique.
- Gaubert, S. (1994). Rational series over dioids and discrete event systems, *Proc. of the 11th Int. Conf. on Analysis and Optimization of Systems: Discrete Event Systems, Sophia Antipolis, Lectures Notes in Control and Information Sciences 199*, Springer.
- Gaubert, S. (1995). Performance evaluation of $(\max, +)$ automata, *IEEE Transactions on Automatic Control* **40**(12): 2014–2025.
- Gaubert, S. (1999). Introduction aux systèmes dynamiques à Événements discrets. Notes de Cours.
- Gill, A. (1962). *Introduction to the Theory of Finite-State Machines*, McGraw-Hill Electronic Sciences Series, McGRAW-HILL Book Company.
- Ramadge, P. e Wonham, W. (1982). Supervision of discrete event processes, *Proceedings of 21st Conference on Decision and Control* pp. 1228–1229.
- Ramadge, P. e Wonham, W. (1987). On the supremal controllable sublanguage of a given language, *SIAM Journal on Control and Optimization* **25**(3): 637–659.
- Ramadge, P. e Wonham, W. (1989). The control of discrete event systems, *Proceedings of the IEEE* **77**(1): 81–98.