III SEMINÁRIO NACIONAL DE CONTROLE E AUTOMAÇÃO
INDUSTRIAL, ELÉTRICA E DE TELECOMUNICAÇÕES
A CONSTRUÇÃO DO CONHECIMENTO, EMPREGABILIDADE E EMPREENDEDORISMO

AUTÔMATOS COM TEMPORIZAÇÃO VARIÁVEL E APLICAÇÃO À SÍNTESE DE SUPERVISORES DE SISTEMAS A EVENTOS DISCRETOS TEMPORIZADOS

Eduard Montgomery Meira Costa

Universidade Federal da Bahia / Escola Politécnica / Departamento de Engenharia Elétrica

GRUPO: ☒ **A** ☐ **B** ☐ **C** ☐ **D** ☐ **E**

TEMA**:** Modelos e métodos matemáticos na gestão e análise de desempenho de sistemas de automação e controle.

PALAVRAS CHAVE: Autômatos Temporizados, Sistemas a Eventos Discretos, Controle Supervisório, Álgebra de Dióides, Automação Industrial.

RESUMO **-** Esse artigo utiliza o autômato com temporização variável para modelar sistemas a eventos discretos que apresentam transições de estados cujo tempo associado é variável. Essas variações nos tempos de vida podem ser exemplificadas por sistemas que apresentam desgastes em seus recursos. A formalização para construir o supervisor é baseada na álgebra de dióides e nas matrizes de incidência. A utilização desse autômato permite representar alguns sistemas em uma forma compacta, nos quais os autômatos (max,+) podem ter um número infinito de estados.

ABSTRACT **-** This paper employs the time varying automaton to model discrete event systems that presents state transitions with a variable time associated. These variations in event lifetimes can be exemplified by systems that presents waste in their resources. The formalization to construct the supervisor is based in dioid algebra and incidence matrix. The utilization of this automaton allows to represent some systems in a compact way, where (max,+) automata can have a infinite number of states.

INTRODUÇÃO

Autômatos [1] são modelos de máquinas de estado que podem representar algumas classes de sistemas. Dentre esses sistemas, encontram-se os Sistemas a Eventos Discretos (SEDs) [2], que são sistemas cuja evolução dinâmica é descrita pela ocorrência de eventos que alteram o estado do sistema. Os autômatos permitem estruturar modelos de sistemas em que é necessário estudar seu funcionamento lógico através de sua evolução dinâmica descrita na forma de linguagens formais [3]. Exemplo disto é a formalização do problema de controle de SEDs, que é denominada de Teoria de Controle Supervisório [4]. Nesta teoria, considera-se o conjunto de eventos particionado em eventos controláveis e não controláveis, onde dado o modelo do sistema e uma especificação funcional através de uma linguagem formal, constrói-se um supervisor. O supervisor é o agente que executa as ações de controle sobre os eventos controláveis, definindo a linguagem que o autômato do sistema deve seguir.

Quando se necessita expressar uma representação temporal em um SED, um paradigma que pode ser utilizado é o autômato temporizado [5,6]. Esse autômato permite incluir representações de tempo em sua estrutura, tal que os símbolos ocorrem de acordo com instantes de tempo específicos, determinados em um ou mais relógios globais. No caso do autômato temporizado, a linguagem é representada por um conjunto de pares (*tempo*, *símbolo*), sendo denominada de linguagem temporizada [7,8].

Outro formalismo utilizado para a representação temporal nos autômatos são os grafos de transições de atividades (GTAs) [9], que apresentam uma estrutura formal semelhante aos autômatos não temporizados. Porém, os GTAs incluem as definições dos tempos máximo e mínimo associados aos símbolos, em que um símbolo é definido como uma tripla (t_{min}, *evento*, t_{max}), em que t_{min} é o menor tempo em que um símbolo torna-se habilitado e t_{max} é o maior tempo em que o símbolo pode ocorrer. A evolução dinâmica dos GTAs pode ser representada através dos grafos de transições temporizadas (GTTs) [10], que incluem o símbolo *t* (*tick*) que é sincronizado a um relógio global. Esta representação gráfica apresenta um aumento de estados que ocorre devido à inclusão deste símbolo. Este tipo de autômato é utilizado por Brandin e Wonham [11] para controle de SEDs temporizados (SEDTs). Os GTAs em conjunto com os GTTs permitem representar sistemas que apresentam transições de estados com limites máximos e mínimos de ocorrências. A linguagem associada é uma linguagem formal que inclui o símbolo *t*.

Para a representação de sistemas em que os tempos são definidos como apenas o limite mínimo em que um símbolo pode ocorrer, o que representa um modelo livre de controle, encontra-se o autômato (max,+) [12], que é similar ao GTA, porém com as definições dos tempos associados aos símbolos apresentados diretamente nos arcos (representação gráfica das funções de transição). Este tipo de autômato temporizado é utilizado em conjunto com sua representação matricial [13] para tratar o problema de controle de SEDTs [14,15].

Quando o sistema apresenta variações nos tempos de vida dos eventos em processos que se repetem, como situações de desgastes de peças, o autômato com temporização variável (*ATV*) [16] se apresenta como um formalismo para modelar estes sistemas e sintetizar supervisores [17]. Assim, esse artigo apresenta a utilização do *ATV* para a modelagem e síntese de supervisores para um SED.

FORMALISMO PRELIMINAR

O formalismo para construir supervisores para SEDTs com tempos de vida dos eventos variáveis necessita da definição de um dióide, que é a álgebra utilizada para este contexto.

Definição 1 *Um dióide é um conjunto D dotado de duas operações: ⊕ (adição) e ⊗ (multiplicação), que é idempotente. O elemento nulo é 'ε' e o elemento identidade é 'e'.*

Um dióide é dito ser comutativo se ⊗ é comutativo. Quando se considera $D=\mathrm{R}_{max}=\mathrm{R}\cup\{-\infty\}$, tem-se $\epsilon=-\infty$, $e=0$, ⊕ é a operação usual max (máximo) e ⊗ é a operação usual + (soma). Nesse caso, *D* é comutativo, e denomina-se a álgebra de álgebra (max,+), onde os elementos $\epsilon=-\infty$, $e=0$ satisfazem:

$$\forall a\in \mathrm{R},\ a \oplus \epsilon = a \oplus (-\infty) = \max\{a, -\infty\} = a = \max\{-\infty, a\} = (-\infty) \oplus a = \epsilon \oplus a \quad (1)$$

$$\forall a\in \mathrm{R},\ a \otimes e = a+0 = a = 0+a = e \otimes a \quad (2)$$

Da equação (2), vê-se que $D=\mathrm{R}_{max}$ é um dióide comutativo.

Na álgebra (max,+), *D* pode ser definido sobre matrizes quadradas de dimensão *n*, isto é, $D=\mathrm{R}_{max}^{n\times n}$. As propriedades do dióide (max,+) são igualmente satisfeitas, considerando que para duas matrizes $\mathbf{A},\mathbf{B}=\mathrm{R}_{max}^{n\times n}$,

$$(\mathbf{A} \oplus \mathbf{B})_{i,j} = \mathbf{A}_{i,j} \oplus \mathbf{B}_{i,j} \quad (3)$$

$$(\mathbf{A} \otimes \mathbf{B})_{i,j} = \bigoplus_{i=1}^{n} (\mathbf{A}_{i,j} \otimes \mathbf{B}_{i,j}) \quad (4)$$

Com esta formalização, pode-se definir o autômato com temporização variável, o que é feito a seguir:

Definição 2 *Um autômato finito sobre um alfabeto* Σ *é uma sextupla ATV=(*Q; Σ; q_0; t_0; t_i; t_f*), em que:*

- $Q=\{q_0, ..., q_n\}$ *é um conjunto finito de estados;*
- $\Sigma=\{\sigma_1, ..., \sigma_m\}$ *é um alfabeto;*
- q_0 *é o estado inicial;*
- $t_0 : q_0 \rightarrow \mathrm{R}_{max}$ *é a função de atraso inicial;*
- $t_i : t_{i-1} \times Q \times \Sigma \times Q \rightarrow \mathrm{R}_{max}$ *é a função de tempo de transição dependente do último tempo decorrido e*
- $t_f : t_{i-1} \times Q \rightarrow \mathrm{R}_{max}$ *é a função de atraso final dependente do último tempo decorrido.*

Um *ATV* é representado graficamente por vértices formados pelo conjunto de estados *Q* e pelos arcos a seguir:

1. Internos, $q_j \xrightarrow{\sigma} q_{j+1}, \forall q_j, q_{j+1} \in Q$ e $\sigma\in\Sigma$ tais que $t_i \neq \epsilon$. O arco $q_j \xrightarrow{\sigma} q_{j+1}$, é valorado por $t_i = f(t_{i-1})$.

de forma que o tempo de transição t_i do arco que aponta de um estado q_a para o estado q_b é uma função *f* do tempo decorrido anteriormente t_{i-1} pela ocorrência de um evento, e que levou o autômato do estado q_c para o estado q_a;

2. O arco de entrada $\rightarrow q_0$, valorado por $t_0 \neq \epsilon$;

3. Os arcos de saída $q_j \rightarrow$, valorados por t_f, $\forall q_f \in Q$, tais que $t_f = f(t_{i-1}) \neq \epsilon$, com t_f definido de forma igual à t_i.

Da definição do *ATV*, vê-se que o autômato (max,+) é um caso restrito, em que se consideram todas as funções de tempos de vida dos eventos definidas como constantes. Assim, sua semântica é definida de maneira similar, considerando as variações nos tempos das transições: 1) Há um relógio global que está sempre sendo incrementado; 2) O tempo de vida t_i de um evento σ, é uma função *f* do último tempo decorrido t_{i-1} de um evento σ', que levou o *ATV* para o estado em que o evento σ é definido, e seu valor calculado é o tempo mínimo necessário para sua habilitação; 3) Para iniciar a execução do autômato (alcançar o estado inicial) é transcorrido um tempo t_0 no relógio global; 4) Estando no estado inicial (ou em qualquer outro), os contadores dos eventos definidos neste estado (valores calculados de cada função de tempo de vida de evento - $t_i(\sigma)$) vão sendo decrementados; 5) Quando um contador de um dos eventos definidos no estado é zerado, o evento se torna habilitado e pode ocorrer a qualquer instante; 6) Se com a incrementação do relógio global, for zerado o contador de um outro evento definido neste estado, ele também se torna habilitado; 7) A ocorrência de um evento habilitado reinicializa seu contador e muda o estado do autômato, desabilitando os outros eventos; 8) Quando um estado marcado é alcançado, ao transcorrer o tempo de atraso final, o *ATV* reconhece este estado, reconhecendo assim a palavra que o levou do estado inicial até este estado; 9) Eventos iguais com diferentes funções de tempos de vida definem não determinismo no *ATV*.

Exemplo 1 *Seja* $\Sigma = \{\alpha, \beta\}$. *O autômato com conjunto de estados* $Q = \{1, 2, 3\}$, *atraso inicial* $t_0=2$, *funções de tempos das transições* $f_1 = t^2 - t$, $f_2 = 3t$, $f_3 = 4$, $f_4 = 1+2t$ *e função de atraso final* $f_5=5-2/t$ (*os outros valores de f são* ϵ) *está apresentado na Fig. 1. O valor de t é interpretado como* t_{i-1}, *isto é, o último tempo decorrido para atingir um determinado estado. Assim, o tempo de cada transição em um dado instante de tempo é sempre calculado em relação ao tempo decorrido para atingir o último estado alcançado do autômato. Assim, o valor do primeiro tempo de vida da transição no arco* $q_1 \xrightarrow{\alpha} q_2$ *é* $t_1=f_1(2)=2$, *desde que* $t_0 = 2$. t_1 *é o tempo mínimo para a ocorrência do evento* α *que muda o estado do autômato de* q_1 *para* q_2. *O tempo final para reconhecer a palavra vazia* ε *é* $t_f = f_5(2) = 4$, *tal que seu reconhecimento tem um tempo total* $t= 6$. *Na ocorrência de* α, *o autômato vai para o estado* 2, *onde o tempo de transição para* α *é* $t_2=f_2(2)=6$ *e para* β *é* $t_2 = f_4(2) = 5$. *Se ocorre* α, *o autômato vai para o estado* 3, *onde o tempo de transição para* β *é* $t_3=f_3(5)= 4$. *Ainda no estado* 2, *se ocorre* β, *o autômato vai para o estado* 1, *onde o novo tempo de transição para* α *é* $t_3=f_4(6)=30$ *e para reconhecer a palavra* $\alpha\beta$, *o atraso final é* $t_f=4{,}6$, *tal que seu reconhecimento tem um tempo total* $t=13{,}6$.

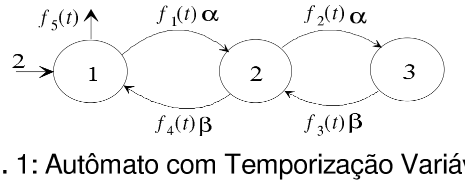

Fig. 1: Autômato com Temporização Variável

O reconhecimento de uma palavra se dá quando um estado marcado é atingido e o tempo de atraso final é transcorrido. Este reconhecimento pode ser descrito por meio das seqüências de eventos através dos datadores definidos sobre o conjunto $R_{max} = R \cup \{-\infty\}$, do dióide (max,+) [18], ou através das linguagens associadas às suas matrizes de incidência. Estas são definidas a seguir.

Definição 3 *Seja ATV um autômato com temporização variável. Define-se a matriz de incidência, denotada por* $\mathbf{A}$, *como*

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

em que $f\,(t_\sigma)$ *é a função do tempo de vida do evento* σ *que leva o autômato do estado i para o estado j. Se mais de um evento é definido do estado i para j, então* $a_{i;j} = \bigoplus_k f(t_{\sigma^k})\sigma^k$. *O estado inicial é definido como sendo o estado* 1, *representado pelo vetor linha* $\theta(\mathbf{A}) = [t_0\ \epsilon\ \ldots\ \epsilon]$, *e os estados marcados são representados pelo vetor coluna* $\phi(\mathbf{A}) = [f(t_{f1})\ \ f(t_{f2})\ \ldots\ f(t_{fn})]^T$.

Nessa representação, se um estado k não é marcado, $f(t_{fk})=\epsilon$. A notação θ_j, refere-se a j-ésima coluna do vetor θ, e ϕ_i, a i-ésima linha do vetor ϕ.

Exemplo 2 *O autômato ATV apresentado na Fig. 1, tem sua matriz de incidência definida por*

$$\mathbf{A} = \begin{bmatrix} \epsilon & (t^2\text{-}t)\alpha & \epsilon \\ (1+2t)\beta & \epsilon & (3t)\alpha \\ \epsilon & 4\beta & \epsilon \end{bmatrix}$$

$$\theta(\mathbf{A}) = [\ 2 \quad \epsilon \quad \epsilon\ ],\quad \phi(\mathbf{A}) = \begin{bmatrix} 5\text{-}2/t \\ \epsilon \\ \epsilon \end{bmatrix}$$

Para a construir as linguagens dos *ATV*s usando as matrizes de incidência, define-se:

Definição 4 *Seja* $\mathbf{A}$ *uma matriz de incidência. A matriz* $\mathbf{A}^n = \mathbf{A} \otimes \mathbf{A} \otimes \ldots \otimes \mathbf{A}$, *é uma matriz de caminhos, onde cada elemento* $a^n_{i,j}$ *representa um ou mais caminhos de comprimento n, formado por símbolos do alfabeto* Σ, *que levam o autômato que ela representa, do estado i para o estado j, com um tempo total* $f(t_s) = f(t_{\sigma 1}) \otimes \ldots \otimes f(t_{\sigma n})$, $s=\sigma^1\ldots\sigma^n$. *Os vetores de estado inicial e de estados marcados da matriz de caminhos* $\mathbf{A}^n$ *são os mesmos de* $\mathbf{A}$.

Se não há um caminho com n eventos que muda o estado do autômato do estado i para o estado j, então $a^n_{i,j} = \epsilon$. A matriz $\mathbf{A}^n$ contém palavras s de comprimento n que são percorridas em um tempo $f(t_s) = f(t_{\sigma 1}) \otimes \ldots \otimes f(t_{\sigma n})$.

Exemplo 3 *Do ATV mostrado na Fig. 2, sua representação matricial é dada por*

$$\mathbf{A} = \begin{bmatrix} \epsilon & 2t\alpha \\ (2\text{-}t)\beta & \epsilon \end{bmatrix},\quad \theta(\mathbf{A}) = [\ 2 \quad \epsilon\ ],\quad \phi(\mathbf{A}) = \begin{bmatrix} 3t \\ \epsilon \end{bmatrix}.$$

A matriz de caminhos $\mathbf{A}^2$ *desse autômato é*

$$\mathbf{A}^2 = \mathbf{A} \otimes \mathbf{A} = \begin{bmatrix} ((2t_1) + (t_2\text{-}2))\,\alpha\beta & \epsilon \\ \epsilon & ((t_2\text{-}2) + (2t_3))\,\beta\alpha \end{bmatrix}$$

com $\theta(\mathbf{A}^2)=\theta(\mathbf{A})$ e $\phi(\mathbf{A}^2)=\phi(\mathbf{A})$. Nessa matriz de caminhos, cada seqüência representa uma palavra de comprimento 2*, que muda o estado do ATV, do estado i para o estado j. Observe que cada termo (i, j) de $\mathbf{A}^2$ não contém uma equação apenas em t. Isto porque cada tempo t_i é calculado separadamente, de acordo com a evolução do ATV.*

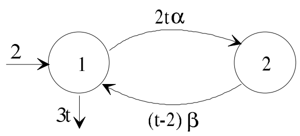

Fig. 2: Autômato para exemplo de B_{uc}^{n}.

Com essa definição, tem-se:

Definição 5 *Para uma dada matriz de incidência* $\mathbf{A}$*, sua linguagem é definida como*

$$L(\mathbf{A}) = \bigoplus_i (\theta(\mathbf{A}) \otimes \mathbf{A}^i) = \bigoplus_i \bigoplus_{j=1}^{n} (\theta_1(\mathbf{A}) \otimes a_{1,j}^i), \qquad (5)$$

onde $\theta_1(\mathbf{A})$ é o elemento da primeira coluna do vetor de estado inicial $\theta(\mathbf{A})$ e $a_{1,j}^i$ é o elemento da linha 1*, coluna* j *da matriz de caminhos $\mathbf{A}^i$.*

Definição 6 *Para uma matriz de incidência* $\mathbf{A}$*, sua linguagem marcada é definida como*

$$L_m(\mathbf{A}) = \bigoplus_i (\theta(\mathbf{A}) \otimes \mathbf{A}^i \otimes \phi(\mathbf{A})) = \bigoplus_i \bigoplus_{j=1}^{n} (\theta_1(\mathbf{A}) \otimes a_{1,j}^i \otimes \phi_j(\mathbf{A})) \qquad (6)$$

onde $\theta_1(\mathbf{A})$ é o elemento da primeira coluna do vetor de estado inicial $\theta(\mathbf{A})$, $a_{1,j}^i$ é o elemento da linha 1*, coluna marcada j da matriz de caminhos $\mathbf{A}^i$ e $\phi_j(\mathbf{A})$ é o elemento da j-ésima linha do vetor de estados marcados $\phi(\mathbf{A})$.*

Exemplo 4 *Considere o autômato apresentado na Fig. 2. Para determinar sua linguagem, utilizam-se sua matriz de incidência, a matriz de caminhos $\mathbf{A}^2$ encontrada no Exemplo 3, e calcula-se*

$$\mathbf{A}^3 = \begin{bmatrix} \epsilon & ((2t_1)+(t_2\text{-}2)+(2t_3))\,\alpha\beta\alpha \\ ((t_2\text{-}2)+(2t_3)+(t_4\text{-}2))\,\beta\alpha\beta & \epsilon \end{bmatrix}$$

$$\vdots$$

em que, para cada uma dessas matrizes citadas, os elementos da primeira linha definem as palavras de comprimento 1, 2, 3,...*, da linguagem de* **A** *. Calculando os tempos t_i, têm-se t_1 =2, t_2=4, t_3=4, t_4=4,... Substituindo esses valores e multiplicando essas matrizes por $\theta_1(\mathbf{A})$, encontra-se $L(\mathbf{A}) = \{2\varepsilon, 6\alpha, 8\alpha\beta, 12\alpha\beta\alpha, ...\}$. Observe que a palavra vazia ε está presente em $L(\mathbf{A})$, a qual tem um tempo de vida igual a* 2 *(tempo de atraso inicial do autômato), e que as palavras de comprimento i, encontram-se na linha* 1 *das matrizes $\mathbf{A}^i$. A linguagem marcada desta matriz é determinada pelas palavras da linha* 1 *de $\mathbf{A}^i$, multiplicadas por $\theta_1(\mathbf{A})$ e $\phi_k(\mathbf{A})$, em que seu valor é calculado em relação ao último tempo decorrido. Então, $L_m(\mathbf{A})$ é dada por $L_m(\mathbf{A}) = \{8\varepsilon, 16\alpha\beta, ...\}$.*

As propriedades dos autômatos como acessibilidade, coacessibilidade, e outras operações, como a composição síncrona, são definidas de forma similar para os ***ATV***s, o que pode ser visto em [17].

SÍNTESE DO SUPERVISOR

O formalismo para a síntese de supervisores de SEDs modelados por ***ATV***s utiliza a formulação de Costa [15]. Assim, com a definição da matriz de incidência, considera-se $\Sigma=\Sigma_c \cup \Sigma_{uc}$, de onde se constrói a matriz de incidência dos eventos não controláveis $\mathbf{A}_{uc}$, e definem-se a especificação de comportamento $\mathbf{E}$ e o supervisor $\mathbf{S}$ de forma similar (como matrizes de incidência). Além do mais, aqui também é considerado que o procedimento de síntese deve ser feito através da transformação das matrizes de incidência $\mathbf{A}$ em $\mathbf{A}^\#$ e $\mathbf{E}$ em $\mathbf{E}^\#$, tal que as condições $L(\mathbf{E})=L(\mathbf{E}^\#)$, $L(\mathbf{A})=L(\mathbf{A}^\#)$ e $L(\mathbf{E}^\#)\subseteq L(\mathbf{A}^\#)$. Este procedimento pode ser visto em [15].

Com as considerações citadas, definem-se os operadores:

$$ACES(\mathbf{A}) = \mathbf{B},\ b_{i,j} = \begin{cases} a_{i,j} & se\ i\ é\ acessível \\ \epsilon & caso\ contrário. \end{cases}$$

$$COACES(\mathbf{A}) = \mathbf{B},$$

$$b_{i,j} = \begin{cases} a_{i,j}\ se\ \exists s \,|\, s = a_{i,j_1} a_{j_1,j_2} \ldots a_{j_{n-1},j_n}, \\ \qquad a_{i,j_1}, a_{j_1,j_2}, \ldots, a_{j_{n-1},j_n} \neq \epsilon\ \ e\ \phi_{j_n}(\mathbf{A}) = e; \\ \epsilon \quad caso\ contrário. \end{cases}$$

$$TRIM(\mathbf{A}) = ACES(COACES(\mathbf{A})) = B$$

e $\trianglelefteq$: $\mathbf{A} \trianglelefteq \mathbf{B} \Rightarrow L(\mathbf{A}) \subseteq L(\mathbf{B})$; $\trianglerighteq$: $\mathbf{A} \trianglerighteq \mathbf{B} \Rightarrow L(\mathbf{A}) \supseteq L(\mathbf{B})$; $\triangleright$:$\mathbf{A} \triangleright \mathbf{B} \Rightarrow L(\mathbf{A}) \supset L(\mathbf{B})$ e $\triangleleft$: $\mathbf{A} \triangleleft \mathbf{B} \Rightarrow L(\mathbf{A}) \subset L(\mathbf{B})$.

Com estas operações, diz-se que uma especificação de comportamento é válida se $\mathbf{E} \neq [\epsilon]$ (matriz nula) e se

$$\forall e_{i,j},\ f(t_\sigma) \geq f(t_\sigma'),\ f(t_\sigma')\ função\ de\ tempo\ de\ vida\ de\ \sigma \subset a_{i,j}$$

$$\theta_1(\mathbf{E}) \geq \theta_1(\mathbf{A}) \qquad e$$

$$\phi_i(\mathbf{E}) \geq \phi_i(\mathbf{A}) \vee \phi_i(\mathbf{E}) = \epsilon,\ \forall \phi_i(\mathbf{A}) \neq \epsilon,\ \forall i = 1\ até\ N,$$

e a condição de controlabilidade é dada por

$$ACES(\mathbf{E} \oplus \mathbf{A}_{uc}) = \mathbf{E}. \qquad (7)$$

Com estas condições, a síntese do supervisor é formalizada como a seguir:

Lema 1 *Um supervisor* $\mathbf{S}$ *para uma matriz de incidência* $\mathbf{A}$ *é definido por* $\mathbf{E}$ *se e somente se* $ACES(\mathbf{E} \oplus \mathbf{A}_{uc}) = \mathbf{E}$.

Corolário 1 *Dada uma especificação de comportamento* $\mathbf{E}$ *válida e uma matriz de incidência temporizada trim* $\mathbf{A}$*,* $\mathbf{S} = TRIM(\mathbf{E})$ *se e somente* $ACES(\mathbf{E} \oplus \mathbf{A}_{uc}) = \mathbf{E}$.

Este corolário garante a construção de um supervisor para o caso em que a especificação de comportamento satisfaz a condição de controlabilidade. Quando esta condição não é satisfeita, isto é, $ACES(\mathbf{E}\oplus\mathbf{A}_{uc}) \triangleright \mathbf{E}$, é necessário encontrar a suprema sublinguagem controlável - *Sup*(*L*). Isto é formalizado pelo teorema a seguir.

Teorema 1 *Dada* $\mathbf{E}$ *válida e a matriz* $\mathbf{A}_{uc}$, *se* $ACES(\mathbf{E}\oplus\mathbf{A}_{uc})\triangleright\mathbf{E}$, *então a Sup*(*L*) *será determinada recursivamente por:*

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

$$\mathbf{B}^n_{uc} = \mathbf{E} \oslash (\mathbf{A}_{uc})^{n-1}$$

$$\mathbf{S}^n = \left[st^n_{ij}\right], st^n_{ij} = \begin{cases} st^{n-1}_{i,j} & \text{se } \sigma^1\sigma^2_{uc}...\sigma^n_{uc} \in \mathbf{B}^n_{uc} \wedge \sigma^n_{uc} \in \mathbf{E}; \\ \epsilon & \text{se } \sigma^n_{uc} \notin \mathbf{E} \wedge \sigma^1 \in \Sigma_c \end{cases}$$

$$\mathbf{S}^n = TRIM(\mathbf{S}^n)$$

onde σ^n_{uc} é o n-ésimo evento da seqüência de um termo de $\mathbf{B}^n_{uc}$, que pode ser $\sigma_{uc} \notin \mathbf{E}$.

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

Neste teorema, $\mathbf{B}^n_{uc} = \mathbf{E} \otimes (\mathbf{A}_{uc})^{n-1}$, que é a matriz de caminhos cujo primeiro elemento de cada seqüência é um evento de $\mathbf{E}$ e os demais são elementos de $\mathbf{A}_{uc}$, com o vetor de estados finais dado por

$$\phi_i(\mathbf{B}^n_{uc}) = \begin{cases} \phi_i(\mathbf{E}) \oplus \phi_i(\mathbf{A}), & se\ \phi_i(\mathbf{A}) \neq \epsilon \\ \epsilon & caso\ contrário, \end{cases}$$

e o de estado inicial por $\theta(\mathbf{B}^n_{uc}) = \theta(\mathbf{A}) = \theta(\mathbf{E})$.

Estas condições garantem a construção de um supervisor para um SED temporizado (com tempos de vida constantes ou variáveis) ou não temporizado, como visto em [17].

RESULTADOS E DISCUSSÃO

A seguir é apresentado um exemplo de síntese do supervisor para um SED modelado por *ATV*, onde é mostrado a eficácia da abordagem proposta.

Exemplo 5 *Considere o ATV apresentado na Fig. 3. Este autômato representa uma simples célula de manufatura, na qual podem-se processar peças gerando materiais diversos* (*palavras* αβ, ακββ, ακμββ, μμββ, μμβκμββ, μβκββ, *etc.*). *Neste sistema, quando se processam peças que precisam ser trabalhadas em uma determinada máquina* (*M*), *o que é definido pelo alcance do estado 4, pode ocorrer uma pane no sistema* (*evento não controlável* ν) *que pode paralisá-lo totalmente, ou ser reinicializado através do evento* η. *As funções de tempos de vida dos eventos são:* $f_0=2$; $f_1=1{,}1t$; $f_2=1{,}001t$; $f_3=t/3$; $f_4=0{,}1t$; $f_5=t+1$; $f_6=1{,}2t$; $f_7=2$; $f_8=f_9=t/4$ *e* $f_{10}=f_0$. *Esses tempos indicam o menor tempo para cada evento se tornar habilitado, podendo ocorrer e variam para cada tempo de chagada ao respectivo estado onde eles são definidos* (*ver o Exemplo 1*). *Assim*, tem-se neste sistema apenas o evento $\nu \in \Sigma_{uc}$.

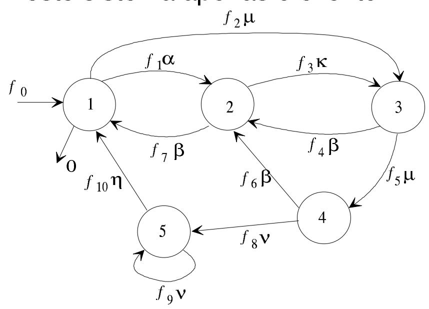

Fig. 3: Sistema modelado por um *ATV*.

Para este sistema, deseja-se construir peças que não necessitam do processamento da máquina M, e tenham para cada evento, um tempo de vida 5% atrasado em relação ao tempo de vida original do sistema. Esta especificação é vista na Fig. 4, em que $f_0=2{,}1$; $f_1=1{,}155t$; $f_2=1{,}05105t$; $f_3=0{,}01667t$; $f_4=0{,}105t$ *e* $f_7=2{,}1$. *Esta especificação é válida e satisfaz* $ACES(\mathbf{E}\oplus\mathbf{A}_{uc})=\mathbf{E}$. *Dessa forma,* $\mathbf{S}=TRIM(\mathbf{E})=\mathbf{E}$.

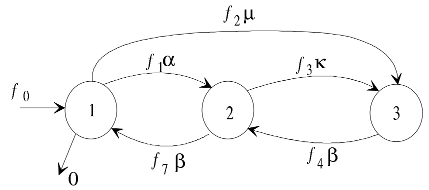

Fig. 4: Especificação controlável.

Por outro lado, se for considerado a construção de peças que utilizam a máquina M, sem que o sistema entre em pane, e que estas peças não sejam iniciadas através da ocorrência do evento μ, *e as funções* f_i *sejam as mesmas citadas anteriormente, porém com* $f_7=2t$, *tem-se a especificação dada pelo ATV da Fig. 5.*

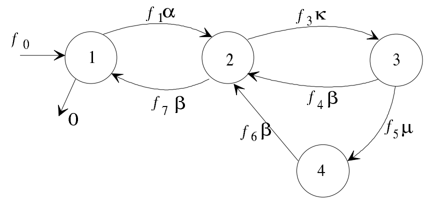

Fig. 5: Especificação não controlável.

Para esta especificação, $ACES(\mathbf{E}\oplus\mathbf{A}_{uc})\triangleright\mathbf{E}$. *Assim, calculando* $\mathbf{B}^n_{uc} = \mathbf{E} \otimes (\mathbf{A}_{uc})^{n-1}$ *para n=2, encontra-se o termo* (*i,j*)=(*3,1*) *dado pela palavra* μν, *que define a saída do estado 3 pelo evento* μ *para o estado 4 e, do estado 4 por* ν *para o estado 5, o que determina a pane no sistema. Como o evento* μ *é controlável, inibindo-o no estado 3, encontra-se o supervisor apresentado na Fig. 6, em que* $f_7=\max(2t;\ 2)$. *Observe que, como o valor de* f_7 *no sistema é um valor constante igual a 2, nos processamentos de* *peças em que o sistema gera palavras que passam pelo estado 3, o tempo para f_7 sempre se torna menor que 2. Isto implica em acelerar a ocorrência deste evento no sistema, o que não é permitido.*

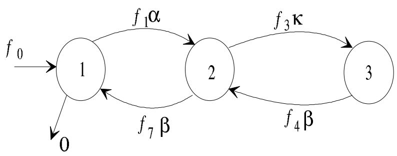

Fig. 6: Supervisor para a SupC(*L*).

CONSIDERAÇÕES GERAIS

A formulação para a síntese do supervisor para SEDs com as características apresentadas, mostra-se similar à aplicação da abordagem de Costa [15] para os autômatos (max,+). A complexidade computacional para esta abordagem é da mesma ordem do algoritmo da SupC(*L*) de Ramadge e Wonham [4].

A construção das linguagens dos *ATV*s apresenta uma maior complexidade pela necessidade da separação explícita dos tempos associados para cada estado alcançado. Com a formalização proposta, mostra-se que não é necessário o cálculo da matriz B^{n}_{uc} utilizando a temporização, para solucionar o problema de controle.

CONCLUSÃO

Aqui é mostrado o uso do *ATV* na síntese de supervisores de SEDTs. Com o *ATV* alguns modelos de sistemas que apresentam variação nos tempos de vida dos eventos, os quais são dependentes de ocorrências anteriores, podem ser construídos com a introdução das funções de tempo de vida. Este autômato se mostra determinístico e finito para os casos em que os autômatos (max,+) [12], geram modelos não determinísticos ou infinitos. O exemplo apresentado mostra sua eficácia na representação dessa classe de SEDs, bem como na síntese de supervisores, garantindo inclusive que especificações dadas, não introduzam redução nos tempos de vida dos eventos. Dessa forma, o *ATV* se apresenta como um importante paradigma na modelagem de SEDs, solucionando tanto o problema apresentado, como os problemas definidos em Costa [15], quando se consideram todas as funções de tempos de vida como constantes (caso temporizado) ou definidas por e (caso não temporizado). Este formalismo está sendo atualmente utilizado para a estruturação de supervisores com adaptação temporal para variações nos tempos de vida de especificações de comportamento definidas.

AGRADECIMENTOS

O autor agradece ao CNPq e à UFBA pelo suporte dado a esta pesquisa.

REFERÊNCIAS

[1] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, USA, 1979.

[2] P.J.G. Ramadge and W.M. Wonham. Supervision of discrete event processes. *Proceedings of 21st Conference on Decision and Control*, pages 1228–1229, 1982.

[3] A. Salomaa. *Formal Languages*. ACM Monograph Series. Academic Press, Inc. New York, 1973.

[4] P.J.G. Ramadge and W.M. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–98, 1989.

[5] R. Alur and D. Dill. Automata for modeling real-time systems. *Proc. 17th International Colloquium on Automata, Languages and Programming, Lectures Notes on Computer Science, New York: Springer Verlag*, 443:322–335, 1990.

[6] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, (126):183–235, 1994.

[7] R. Alur and T.A. Henzinger. Back to the future: Towards a theory of timed regular languages. *IEEE 0-8186-2900-2/92*, pages 177–186, 1992.

[8] P. Caspi E. Asarin and O. Maler. A kleene theorem for timed automata. *Proc. of LICS'97*, pages 160–171, 1997.

[9] J.S. Östroff and W.M. Wonham. A framework for real-time discrete event control. *IEEE Transactions on Automatic Control*, 35(4):386–397, April 1990.

[10] M.S. Lawford. *Model Reduction of Discrete Real-Time Systems*. PhD thesis, University of Toronto, 1997.

[11] B.A. Brandin and W.M. Wonham. Supervisory control of timed discrete-event systems. *IEEE Transactions on Automatic Control*, 39(2):329–342, 1994.

[12] S. Gaubert. Performance evaluation of (max,+) automata. *IEEE Transactions on Automatic Control*, 40(12):2014–2025, December 1995.

[13] A. Gill. *Introduction to the Theory of Finite-State Machines*. McGraw-Hill Electronic Sciences Series. McGRAW-HILL Book Company, 1962.

[14] E.M.M. Costa and A.M.N. Lima. Síntese de supervisores de sistemas a eventos discretos temporizados. *Anais do V Simpósio Brasileiro de Automação Inteligente* (V *SBAI*), 2001.

[15] E.M.M. Costa. *Síntese de Supervisores de Sistemas a Eventos Discretos Temporizados e Não Temporizados*. Tese de Doutorado. UFPB - Campus II, Novembro de 2001.

[16] E.M.M.Costa. *Autômatos com Temporização Variável: Um Novo Formalismo para Representação de sistemas*. Revista Ciência e Engenharia, Universidade Federal de Uberlândia – Julho/Dezembro de 2002.

[17] E.M.M.Costa. *Síntese de Supervisores de Sistemas a Eventos Discretos Temporizados Utilizando Autômatos com Temporização Variável*. Anais do XIV Congresso Brasileiro de Automática (CBA2002), Natal, RN. Setembro de 2002.

[18] H. Comon and Y. Jurski. *Timed automata and the theory of real numbers. Proc. 10th Int. Conf. Concurrency Theory*, 1664 of Lectures in Computer Science:242–257, 1999.