



**HAL**  
open science

# Avanços na teoria de controle para sistemas lineares na álgebra tropical

Mariano Vinicius, Carlos Andrey Gonçalves, Laurent Hardouin

► **To cite this version:**

Mariano Vinicius, Carlos Andrey Gonçalves, Laurent Hardouin. Avanços na teoria de controle para sistemas lineares na álgebra tropical. Congresso Brasileiro de Automatica, Sep 2014, Belo Horizonte, Brazil. pp.1704-1711. hal-02535614

**HAL Id: hal-02535614**

**<https://univ-angers.hal.science/hal-02535614>**

Submitted on 7 Apr 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## AVANÇOS NA TEORIA DE CONTROLE PARA SISTEMAS LINEARES NA ÁLGEBRA TROPICAL

VINICIUS MARIANO GONÇALVES, CARLOS ANDREY MAIA\*, LAURENT HARDOUIN†

\*Programa de Pós Graduação de Engenharia Elétrica, Universidade Federal de Minas Gerais, UFMG, Brasil

†Laboratoire Angevin d'Ingénierie des Systèmes (LARIS-ISTIA), Université d'Angers, França

Emails: mariano@cpdee.ufmg.br, maia@cpdee.ufmg.br, laurent.hardouin@univ-angers.fr

**Abstract**— Recently, attention has been given to theory of linear systems in the so-called “Tropical Algebra”, which is suitable to model certain kinds of discrete event systems. This text proposes a method (computationally simple, in comparison with previous approaches that solve similar problems) for the resolution of a problem of regulation in the tropical algebra setting, being the first to do so (by the authors knowledge) in the manner that will be posed. An example extracted from a study in the industry setting will be used to show the efficiency of the proposed method.

**Keywords**— Tropical Algebra, Linear System Theory

**Resumo**— Recentemente, tem sido dada atenção a teoria de sistemas lineares na chamada “Álgebra Tropical”, teoria essa que é adequada para a modelagem de certos tipos de eventos discretos. Este texto propõe um método (computacionalmente simples, em comparação com os métodos anteriores que resolvem problemas similares) para a resolução de um problema de regulação na álgebra tropical, sendo o primeiro trabalho (no conhecimento dos autores) a fazê-lo da forma como será colocada. Um exemplo retirado de um estudo de caso real na indústria será utilizado para demonstrar a eficácia do método proposto.

**Palavras-chave**— Álgebra Tropical, Teoria de Sistemas Lineares

### 1 Introdução

A teoria de controle para sistemas lineares invariantes no tempo (amostrados no tempo, nesse caso)

$$\mathbf{x}[k+1] = A\mathbf{x}[k] + B\mathbf{u}[k] \quad (1)$$

foi e é amplamente estudada. Seus resultados, baseado em conceitos fortes e elegantes da álgebra linear, são ubíquos em grades curriculares de engenheiros na área de controle e automação. Sua importância é irrefutável, seja como aplicação direta ou como base para teorias mais gerais (como a de sistemas não lineares).

Certos tipos de sistemas a eventos discretos, mais especificamente Grafos a Eventos Temporizados (uma subclasse de Redes de Petri, veja (Baccelli et al., 1992)), GETs doravante, admitem uma representação por espaço de estado curiosamente similar à de (1)

$$\mathbf{x}[k+1] = A\mathbf{x}[k] \oplus B\mathbf{u}[k] \quad (2)$$

em que  $x_i[k], u_i[k]$  representam o tempo do  $k$ -ésimo disparo da  $i$ -ésima transição de estado e de controle, respectivamente. Entretanto, no contexto de (2), os produtos e somas matriciais são interpretados em outra álgebra: a chamada *Álgebra Tropical*<sup>1</sup>.

<sup>1</sup>Também conhecida como *álgebra Max-Plus*. A álgebra foi assim nomeada em honra ao matemático húngaro-brasileiro Imre Simon, que introduziu o semi-anél para resolver um problema na teoria de linguagens formais (ver (Gaubert and M.Plus, 1997)).

Recentemente, uma atenção significativa foi colocada no propósito de desenvolver uma teoria sólida de sistemas lineares para (2). Devido a estrutura peculiar da álgebra tropical, poucos resultados da bem desenvolvida teoria para sistemas na álgebra convencional (equação (1)) podem ser facilmente transpostos para essa nova álgebra. Em particular, conceitos como controlabilidade e observabilidade ainda não são bem compreendidos (veja (Loreto et al., 2010)). Pode-se dizer portanto que a teoria ainda encontra-se em relativa infância.

Este texto trata de um problema cujo análogo mais próximo na teoria tradicional de sistemas lineares seria o problema de *regulação*. Nesse problema, em sua formulação mais geral, procura-se por um controlador  $\mathbf{u}[k]$  que garanta a convergência em regime permanente para um espaço vetorial  $\mathcal{V}$  e sua permanência nesse. Dessa forma, o problema análogo na álgebra tropical - assunto desse texto - procura um controlador  $\mathbf{u}[k]$  que garanta que  $\mathbf{x}[k]$  em (2) convirja em regime permanente a um *semimódulo*, que é o análogo de um espaço vetorial no contexto de semianéis (que é a classe algébrica da álgebra tropical), e permaneça nesse.

### 2 Estado da arte

Colocando o problema informalmente como

“Dado um sistema da forma (2) e um semimódulo  $\mathcal{S}$ , encontre um controlador realizável (isso é, causal e não decrescente) que garanta, para todas as condições iniciais  $\mathbf{x}[0]$ , que  $\mathbf{x}[k]$  eventualmente

convirja para  $\mathcal{S}$  em um número finito de passos”

tem-se que, do conhecimento dos autores, o mesmo nunca foi posto e tratado na literatura exatamente dessa forma no tocante da generalidade do conjunto  $\mathcal{S}$  e também da abrangência para a condição inicial.

Do conhecimento dos autores, o primeiro texto a abordar um problema similar ao proposto foi (Amari et al., 2004) (o que sugere que o problema é relativamente recente), embora o texto trate de um problema muito específico (não há genericidade).

Em seguida (Amari et al., 2005) considera o problema para todas as condições iniciais e apresenta uma condição suficiente para um tipo específico de semimódulo  $\mathcal{S}$ . (Katz, 2007), também considerando um tipo específico de semimódulo  $\mathcal{S}$  (embora a metodologia é facilmente extensível a um genérico), computa concomitantemente um controlador  $\mathbf{u}[k]$  e um subconjunto  $\mathcal{S}_{geo} \subseteq \mathcal{S}$  tal que, se a condição inicial reside em  $\mathcal{S}_{geo}$ , o controlador garante a permanência nesse. Nesse caso, as condições apresentadas são necessárias e suficientes, embora o método apresentado tenha uma complexidade elevada em termos computacionais. (Maia, Andrade and L.Hardouin, 2011), tratando um tipo especial de conjunto  $\mathcal{S}$ , considera que a condição inicial está dentro desse. Foi derivado então condições necessárias e suficientes. (Maia, Hardouin, Santos-Mendes and Loiseau, 2011) trata o problema de controle para um semimódulo genérico e sob a hipótese de que a condição inicial seja um autovetor da matriz do sistema.

Em (Amari et al., 2012), considera-se o problema para todas as condições iniciais, embora apenas uma classe específica de conjuntos  $\mathcal{S}$ . O texto deriva condições suficientes nesse caso. Finalmente, (Gonçalves et al., 2012) lida com um semimódulo  $\mathcal{S}$  genérico, mas assume a escolha de conjunto de condições iniciais  $\mathcal{X}$ . Dado esse conjunto, deriva-se condições suficientes. Ressalta-se também que o trabalho de (Gonçalves et al., 2012) introduz o conceito de sistemas equivalentes e generaliza os resultados de (Maia, Andrade and L.Hardouin, 2011; Maia, Hardouin, Santos-Mendes and Loiseau, 2011), com uma escolha adequada de condições iniciais.

Ressalta-se que, embora pouco estudada até então, o requerimento que a condição inicial seja arbitrária é de extrema importância. Ele induz uma característica (altamente desejável) de *robustez* no sistema: mesmo na ocorrência de perturbações no estado (eventualmente levando  $\mathbf{x}[k]$  para fora de  $\mathcal{S}$ ), o sistema convergirá novamente ao conjunto desejado. Isso ocorre pois, devido a invariância a eventos (as matrizes  $A, B$  não dependerem de  $k$ ), pode se considerar a evolução a partir do estado perturbado como uma nova evolução, no

mesmo sistema, em que a condição inicial é esse estado. Portanto, como a convergência ocorre para todas as condições iniciais, o estado eventualmente retornará ao conjunto desejado.

### 3 Álgebra Tropical

A *álgebra tropical* é um semi-anel idempotente <sup>2</sup> (veja (Baccelli et al., 1992) para uma introdução e (Maia, 2003) para uma referência em português)

$$\mathbb{T}_{max} = \{\mathbb{Z} \cup \{-\infty\}, \oplus, \otimes\} \quad (3)$$

em que  $\oplus$  é o máximo e  $\otimes$  é a soma tradicional. É usual, também, definir o elemento neutro da soma,  $-\infty$ , como  $\perp$ . Como na álgebra tradicional, é usual omitir o símbolo de  $\otimes$  para o produto. Assim,  $xy$  é lido, no contexto da álgebra tropical, como “ $x$  vezes  $y$ ”. Como o “vezes” é interpretado na álgebra tropical, seu significado na álgebra tradicional seria  $x + y$ .

Assume-se doravante que o leitor é familiar com os conceitos básicos da álgebra. A matriz identidade tropical (0 na diagonal e  $\perp$  nos outros elementos) de ordem apropriada é denotada por  $I$ . Para um número natural  $i$  e uma matriz quadrada  $M$ ,  $M^i$  é definido recursivamente como  $M^{i+1} = MM^i$  e  $M^0 = I$ . Se  $\lambda$  é um escalar,  $\lambda^{-1} = -\lambda$ .  $M^* = \bigoplus_{i=0}^{\infty} M^i$  é o *fecho de Kleene* da matriz  $M$ .  $\{M\}_{ij}$  ou  $M_{ij}$ , aquele que for mais conveniente, é o elemento  $ij$  da matriz  $M$ .  $M_{i\bullet}$  é a  $i$ -ésima linha da matriz  $M$ .  $Im\{M\}$  é a imagem da aplicação  $\mathbf{x} \mapsto M\mathbf{x}$ , ou seja, o semimódulo gerado por combinações lineares das colunas de  $M$ . Para uma matriz quadrada  $M$ ,  $\rho(M)$  é o raio espectral (tropical) de  $M$ : o maior autovalor. Todos os vetores estão escritos em negrito e devem ser considerados como vetores colunas.

### 4 Desenvolvimento

#### 4.1 Formulação do problema

Posto formalmente, considere um semimódulo  $\mathcal{S}$  e um sistema

$$\mathbf{x}[k+1] = A\mathbf{x}[k] \oplus B\mathbf{u}[k] \quad (4)$$

com  $A \in \mathbb{T}_{max}^{n \times n}$ ,  $B \in \mathbb{T}_{max}^{n \times m}$ .

Pede-se um controlador  $\mathbf{u}[k]$  *causal* (ou seja, que usa somente as informações disponíveis do estado para computar  $\mathbf{x}[k]$ ) e *não decrescente*:  $\mathbf{u}[k+1] \succeq \mathbf{u}[k]$  (caso contrário não seria implementável) que garante que, para todas as condições iniciais  $\mathbf{x}[0]$ , existe um  $l$  finito tal que  $\mathbf{x}[k] \in \mathcal{S} \forall k \succeq l$ .

Por conveniência, assume-se que  $\mathcal{S}$  é finitamente gerado <sup>3</sup>, ou seja, existe uma matriz  $M \in$

<sup>2</sup>Ou díóide.

<sup>3</sup>Todo espaço vetorial em  $\mathbb{R}^n$  é finitamente gerado com no máximo  $n$  vetores. Isso acontece devido a impossibili-

$\mathbb{T}_{max}^{n \times p}$ ,  $p$  finito, tal que  $\mathcal{S} = \text{Im}\{M\}$ . Na prática, é sempre o caso pois a descrição do semimódulo aparece naturalmente na formulação de um problema na forma a seguir

$$\mathcal{S} = \{\mathbf{x} \mid E\mathbf{x} = D\mathbf{x}\} \quad (5)$$

em que  $E, D \in \mathbb{T}_{max}^{r \times n}$ ,  $r$  finito (todo semimódulo na sua descrição explícita  $\mathcal{S} = \text{Im}\{M\}$  admite uma descrição implícita como em (5), veja “Duality Theorem” and “Finiteness Theorem” em (Gaubert and M.Plus, 1997)). Tal descrição sempre garante que o semimódulo é finito, já que o conjunto de todas as soluções de  $E\mathbf{x} = D\mathbf{x}$  pode ser escrito como  $\text{Im}\{M\}$  para uma matriz com um número finito de colunas (veja (Gaubert and M.Plus, 1997))).

#### 4.2 Condições de solubilidade

Mostra-se que, sobre certas condições, existe um  $F \in \mathbb{T}_{max}^{m \times n}$  tal que o controlador  $\mathbf{u}[k] = F\mathbf{x}[k]$  resolve o problema proposto, provido que todas as entradas da matriz  $F$  sejam não-negativas ou  $\perp$ .

**Lema 1** *O problema proposto tem solução para um controlador da forma  $\mathbf{u}[k] = F\mathbf{x}[k]$  se e somente se existe um natural  $l$  tal que*

$$E(A \oplus BF)^l = D(A \oplus BF)^l. \quad (6)$$

**Prova:** Com o controlador  $\mathbf{u}[k] = F\mathbf{x}[k]$ , tem-se que  $\mathbf{x}[k+1] = (A \oplus BF)\mathbf{x}[k]$  ou  $\mathbf{x}[k] = (A \oplus BF)^k \mathbf{x}[0]$ . Como requer-se que exista um  $l$  tal que  $\forall k \geq l \mathbf{x}[k] \in \mathcal{S}$ , ou,  $E\mathbf{x}[k] = D\mathbf{x}[k]$ , tem-se que deve existir um natural  $l$  tal que

$$E(A \oplus BF)^l \mathbf{x}[0] = D(A \oplus BF)^l \mathbf{x}[0]. \quad (7)$$

Como o mesmo deve ser válido para todo  $\mathbf{x}[0]$ , tem-se a necessidade de (6) (basta escolher  $\mathbf{x}[0]$  como as colunas da matriz identidade). A suficiência provém facilmente após pós-multiplicar ambos os lados de (6) por  $(A \oplus BF)^{k-l} \mathbf{x}[0]$ ,  $k \geq l$ . Assim, tem-se que  $\mathbf{x}[k] \in \mathcal{S} \forall k \geq l$ , como desejado.  $\square$

Em princípio, o problema se resumiria a resolver a equação (6) para as incógnitas  $l$  e  $F$ . Uma tentativa de proceder seria fixar o  $l$  em sucessivos valores e então tentar resolver a equação para  $F$ . Entretanto, resolver (6) para  $l > 1$  é, no geral, uma tarefa difícil: essa se torna uma equação não-linear na álgebra tropical.

A equação será resolvida de maneira indireta. Para isso, é necessária uma definição importante.

**Definição 4.1** *A equação característica associada a um problema é a seguinte equação para as*

*dade de haverem mais de  $n$  vetores linearmente independentes em  $\mathbb{R}^n$ . Na álgebra tropical, não é o caso. Ver (Gaubert and M.Plus, 1997).*

*incógnitas  $\chi \in \mathbb{T}_{max}^n$ ,  $\mu \in \mathbb{T}_{max}^m$ ,  $\lambda \in \mathbb{T}_{max}$*

$$\begin{aligned} (i) : \lambda\chi &= A\chi \oplus B\mu; \\ (ii) : E\chi &= D\chi. \end{aligned} \quad (8)$$

$\square$

A equação (8) é substancialmente mais fácil de ser resolvida do que (6). Por exemplo, fixado o  $\lambda$ , a equação resultante é tropical linear. Nesse caso, existem vários algoritmos eficientes que podem ser utilizados (ver (Gonçalves et al., 2014) para um exemplo de algoritmo e também citações a outros). Dessa forma, uma maneira de proceder é fazer uma pesquisa em linha para  $\lambda \geq \rho(A)$ . Outra alternativa é utilizar o método de (Gaubert and Sergeev, 2013) para resolver *problemas espectrais bilaterais tropicais*, isso é, equações do tipo  $\lambda U\mathbf{y} = V\mathbf{y}$ , notando que (8) pode ser reescrita de maneira equivalente (para  $\lambda \neq \perp$ ) como

$$\lambda \begin{pmatrix} I & \perp \\ E & \perp \end{pmatrix} \begin{pmatrix} \chi \\ \mu \end{pmatrix} = \begin{pmatrix} A & B \\ DA & DB \end{pmatrix} \begin{pmatrix} \chi \\ \mu \end{pmatrix} \quad (9)$$

(ou, simetricamente, a equação obtida trocando  $E$  por  $D$ 's, *vice-e-versa*) que é um problema espectral bilateral tropical. O algoritmo proposto em (Gaubert and Sergeev, 2013) tem complexidade pseudopolinomial e é bastante eficiente em casos médios. De fato, pode-se mostrar que resolver esse problema pode ser reduzido a um problema de otimização com um parâmetro,  $\lambda$ , para uma função não convexa (no geral) e afim por partes.

Ressalta-se que, ao resolver a equação (8), nem sempre  $\lambda$  é um número inteiro ou  $\perp$ , portanto fora do conjunto  $\mathbb{T}_{max}$ . Entretanto, se uma solução  $\neq \perp$  existir, será sempre um número racional da forma  $u/v$ ,  $u$  e  $v$  inteiros<sup>4</sup>. Dessa forma, se as unidades das matrizes  $A$  e  $B$  forem redefinidas por multiplicação (na álgebra tradicional) por  $v$  - uma mera questão de escolha de novas unidades - o novo  $\lambda$  encontrado será  $u$  e portanto inteiro. Assim, podemos sempre assumir  $\lambda \in \mathbb{T}_{max}$ . Ainda, ressalta-se que pela equação (8),  $\lambda\chi \succeq A\chi$ , o que implica que para  $\chi \neq \perp$  necessariamente há de ser  $\lambda \geq \rho(A)$ .

Outra definição importante faz-se necessária.

**Definição 4.2** *O número de convergência  $\kappa(C, D)$  de matrizes  $C \in \mathbb{T}_{max}^{n \times n}$  e  $D \in \mathbb{T}_{max}^{n \times n}$ ,  $\rho(C) < 0$  é o número de passos  $k$  que a sequência*

$$M[k+1] = CM[k] \oplus D \quad (10)$$

*com  $M[0] = I$  requer para convergir para seu valor final,  $M[k] = C^*D$ .  $\square$*

<sup>4</sup>Ver a metodologia para resolver problemas espectrais bilaterais apresentada em (Gaubert and Sergeev, 2013): as únicas operações utilizadas são a soma, subtração, produto e divisão tradicionais. Como o conjunto dos racionais é fechado sob essas operações, conclui-se que se as entradas são inteiras (portanto racionais) então  $\lambda$  é racional.

**Lema 2**<sup>5</sup> Se  $C^*D$  não tem elementos  $\perp$ , o número de convergência é finito.

**Prova:** Claramente, iterando (10)

$$M[k+1] = C^{k+1} \oplus \bigoplus_{i=0}^k C^i D. \quad (11)$$

É sabido que, se  $\rho(C) < 0$  e  $C \in \mathbb{T}_{max}^{n \times n}$ ,  $\bigoplus_{i=0}^k C^i = C^* \forall k \geq n$  (veja (Baccelli et al., 1992)). Assim, assumindo  $k \geq n$

$$M[k] = C^k \oplus C^*D \quad \forall k \geq n. \quad (12)$$

Como  $\rho(C) < 0$ , os elementos de  $C^k$  aproximam de  $\perp$  quando  $k \rightarrow \infty$  (veja (Baccelli et al., 1992)). Assim, se  $C^*D$  não tem nenhum elemento  $\perp$ , existe um  $l$  finito,  $l \geq n$ , tal que para todo  $k \geq l$  tem-se  $C^k \oplus C^*D = C^*D$ . Isso claramente implica que o número de convergência é finito.  $\square$

Outra definição conveniente é

**Definição 4.3** Para um  $\lambda \in \mathbb{T}_{max} \neq \perp$  e uma matriz  $M$ , define-se

$$M_{[\lambda]} \equiv \lambda^{-1}M. \quad (13)$$

$\square$

Então

**Lema 3** Seja  $\{\chi, \mu, \lambda\}$  uma solução para a equação característica do problema tal que  $\lambda > \rho(A)$  e  $\chi$  não tenha entradas  $\perp$ . Seja  $\zeta^T = (-\chi)^T(A_{[\lambda]})^*$ . Então

$$\begin{aligned} (i) : \quad & \zeta^T B \mu = \lambda; \\ (ii) : \quad & \zeta^T A \preceq \lambda \zeta^T. \end{aligned} \quad (14)$$

**Prova:** Note que se  $\lambda > \rho(A)$ , a equação (8)-(i) se reduz<sup>6</sup> a  $\chi = (A_{[\lambda]})^* B_{[\lambda]} \mu$ . Assim, podemos provar (i) pois

$$\zeta^T B \mu = (-\chi)^T ((A_{[\lambda]})^* B \mu) = \lambda (-\chi)^T \chi = \lambda. \quad (15)$$

A afirmação (ii) vem direto da definição de  $\zeta^T$ , já que para toda matriz  $M$ ,  $M^*M \preceq M^*$  e portanto para todo vetor  $\mathbf{n}$ ,  $\mathbf{n}^T M^*M \preceq \mathbf{n}^T M^*$ . Utilizando  $M = A_{[\lambda]}$  e  $\mathbf{n} = -\chi$ , temos o resultado desejado.  $\square$

<sup>5</sup>Embora seja, aparentemente, um resultado simples e bem conhecido na álgebra tropical, os autores não foram capazes de encontrar uma referência com o exato resultado desejado. Portanto, por completude, decidiu-se apresentá-lo.

<sup>6</sup>Um teorema básico da álgebra tropical diz que a equação  $\mathbf{g} = C\mathbf{g} \oplus \mathbf{d}$  tem solução única  $\mathbf{g} = C^* \mathbf{d}$  provido que  $\rho(C) < 0$ . Veja (Baccelli et al., 1992).

**Corolário 4**  $\zeta^T = (-\chi)^T(A_{[\lambda]})^*$  é um autovetor a esquerda de  $A \oplus B\mu\zeta^T$  com autovalor  $\lambda$ :

$$\zeta^T(A \oplus B\mu\zeta^T) = \lambda\zeta^T. \quad (16)$$

**Prova:** Aplicação direta dos resultados (14).  $\square$

Finalmente, o resultado principal desse texto.

**Teorema 5** Seja  $\{\chi, \mu, \lambda\}$  uma solução para a equação característica do problema tal que  $\lambda > \rho(A)$  e  $\chi$  não tenha entradas  $\perp$ . Seja  $\zeta^T = (-\chi)^T(A_{[\lambda]})^*$ . Então  $l = \kappa(A_{[\lambda]}, B_{[\lambda]}\mu\zeta^T)$  e  $F = \mu\zeta^T$  resolvem a equação (6).

**Prova:** Seja  $M[k] = (A \oplus B\mu\zeta^T)^k$ . Pelo Corolário 4, conclui-se que  $\zeta^T M[k] = \lambda^k \zeta^T$ . Assim

$$\begin{aligned} M[k+1] &= (A \oplus B\mu\zeta^T)M[k] = \\ AM[k] \oplus B\mu(\zeta^T M[k]) &= AM[k] \oplus \lambda^k B\mu\zeta^T. \end{aligned} \quad (17)$$

Multiplicando ambos os membros de (17) por  $\lambda^{-(k+1)}$ , é possível trabalhar com a matriz normalizada  $M_{[\lambda]}[k] = \lambda^{-k}M[k]$ :

$$M_{[\lambda]}[k+1] = A_{[\lambda]}M_{[\lambda]}[k] \oplus B_{[\lambda]}\mu\zeta^T. \quad (18)$$

A equação (18) é uma equação recursiva da forma (10). Como  $\lambda > \rho(A)$ ,  $\rho(A_{[\lambda]}) = \rho(A) - \lambda < 0$  (e assim  $\rho(C) < 0$  na equação (10), como requerido). Dessa forma

$$M_{[\lambda]}[l] = ((A_{[\lambda]})^* B_{[\lambda]}\mu)\zeta^T = \chi\zeta^T \quad (19)$$

e portanto  $M[l] = \lambda^l \chi\zeta^T$ . Nota-se que  $l$  é finito pelo Lema 2:  $((A_{[\lambda]})^* B_{[\lambda]}\mu) = \chi$  não tem elementos  $\perp$ , por hipótese. Tampouco é o caso para  $\zeta^T = (-\chi)^T(A_{[\lambda]})^*$ , já que  $-\chi$  não tem elementos  $\perp$  e  $(A_{[\lambda]})^* \succeq I$ , donde  $\zeta^T \succeq -\chi$  e portanto  $\zeta$  não tem elementos  $\perp$ . Assim,  $\chi\zeta^T$  não tem elementos  $\perp$ .

Pela definição de  $\chi$ ,  $E\chi = D\chi$  e consequentemente  $E(\lambda^l \chi\zeta^T) = D(\lambda^l \chi\zeta^T)$ . E a proposição está provada.  $\square$

### 4.3 Causalidade

Após o cômputo de  $F$ , deve-se verificar se  $F$  é uma matriz *causal*: todos os elementos são  $\geq 0$  ou  $\perp$ . Caso contrário, uma lei de controle do tipo  $\mathbf{u}[k] = F\mathbf{x}[k]$  pode exigir predições. Por exemplo,  $u_1[k] = (-2)x_1[k]$  gera a seguinte regra: dispare  $u_1$  2 unidades de tempo *antes* de  $x_1$  disparar.

Entretanto, a obtenção de uma matriz  $F$  não-causal não implica que a mesma não possa ser utilizada para resolver o problema, apenas que nesse caso ações adicionais devem ser tomadas. Se houver um algoritmo que permita prever o estado,  $\hat{\mathbf{x}}[k]$ , então o controlador pode ser utilizado como  $\mathbf{u}[k] = F\hat{\mathbf{x}}[k]$ . Mesmo que o algoritmo de predição cometa um erro eventual - e portanto induza eventuais erros na ação de controle - a robustez do

controlador garantirá a convergência para o estado desejado. Obviamente, se o modelo (2) não for representativo do sistema (as temporizações variam muito com relação os valores nominais utilizados, por exemplo), o preditor cometerá erros com uma frequência elevada. Tal fato ocasionará erros também frequentes na ação de controle. Nessa situação, é possível que o controlador, mesmo sendo robusto, não consiga estabilizar o sistema. O êxito da estratégia controlador robusto+preditor está condicionado, portanto, pela relação entre o índice de robustez do controlador (atrelado ao parâmetro  $l = \kappa(A_{[\lambda]}, B_{[\lambda]}\mu\zeta^T)$ ) e a fidelidade do modelo ao sistema. Uma análise formal dessa situação vai além dos objetivos do presente texto, e portanto a metodologia apresentada vale-se da hipótese informal que o sistema é suficientemente regular (no sentido de desvios com relação ao modelo) para que o controlador proposto seja capaz de rejeitar as perturbações provocadas por eventuais erros na ação de controle.

Propõe-se aqui uma estratégia simples de predição. O algoritmo deverá ser executado a cada instante de tempo  $t$  e assume um controlador da forma  $\mathbf{u}[k] = F\mathbf{x}[k]$  e um sistema da forma (2). Os *estados* do preditor são

- A variável booleana  $N[t]$  ( $V$  ou  $F$ ), que diz se houve informação nova no tempo  $t$ ;
- $k_+[t]$ , o disparo mais recente de *alguma* transição de *controle* até o tempo  $t$ :

$$k_+[t] \equiv \{\max k \mid \exists j u_j[k] \leq t\}. \quad (20)$$

Se a ação de controle mais avançada, digamos  $j$ , teve seu  $k_+[t]$  disparo em um tempo  $\leq t$ , pela lei de controle  $\mathbf{u}[k] = F\mathbf{x}[k]$  isso implica que até o instante  $t$  só há a necessidade de ter predito (ou medido) os estados  $\mathbf{x}[k]$  até  $k_+[t] + 1$  (que é o próximo disparo da ação de controle  $j$ ). É necessário inicializar  $k_+[-1] = -1$ ;

- $k_-[t]$ , o disparo mais recente de *todas* transições de *controle e estado* até o tempo  $t$ :

$$k_-[t] \equiv \{\max k \mid \forall i x_i[k] \leq t, \forall j u_j[k] \leq t\}. \quad (21)$$

Se no instante  $t$  todas as transições  $k_-[t]$  foram medidas, isso implica que não há a necessidade de predizer nenhum evento  $k \leq k_-[t]$ ;

- Os eventos preditos e medidas no tempo  $t$ ,  $\hat{\mathbf{x}}[k, t]$  ou  $\hat{\mathbf{u}}[k, t]$ . A cada evento está associado um *status* que pode ser  $P$  de *predito*,  $M$  de *medido* ou  $I$  de *inicializado*, acessado pela sintaxe  $(\cdot).St$ . Assim, por exemplo,  $\hat{x}_4[3, 15].St = P$  implica que no tempo  $t = 15$  a transição de estado  $x_4$  teve seu disparo  $k = 3$  *predito* pelo algoritmo. Se fosse o caso de  $\hat{x}_4[3, 15].St = M$ , o disparo teria sido *medido* e portanto teria seu valor real  $x_4[3]$ . Como ficará claro pelo algoritmo, no instante de tempo  $t$  só haverá necessidade de guardar na memória de dados os valores de  $k = k_-[t] + 1$  até  $k = k_+[t] + 1$ . É *necessário* uma inicialização dos valores  $\hat{\mathbf{x}}[-1, t]$  e

$\hat{\mathbf{u}}[-1, t]$ , esses podem ser feitos de maneira arbitrária (por exemplo, considerando o tempo inicial  $t = 0$ , um vetor de 0's).

O algoritmo assume uma função  $No(e, t)$ , cuja implementação depende da arquitetura de dados do sistema em questão, que dado um evento  $e$  (estado ou controle) e um tempo  $t$ , informa qual o evento  $k \geq 0$  de  $e$  foi medido mais recentemente até o tempo  $t$ . Assim,  $No(x_3, 18) = 3$  implica que, até o tempo  $t = 18$ , mediu-se  $x_3[0], x_3[1], x_3[2]$  e  $x_3[3]$ . Retornar o valor  $-1$  significa que não foi medido nada, até então.

A função principal,  $Preditor()$ , deve ser executada antes de implementar a ação de controle e depois da dinâmica do sistema ter ocorrido. Assume-se então que no intervalo de tempo aberto  $(t, t+1)$  não ocorrerá nenhum evento relativo aos estados, mas sim somente nas transições de controle (advindos das ações do controlador).

---

```

função PREDITOR( )
  Atualizakmais();
  Atualizakmenos();
   $N[t] \leftarrow F$ ;
  se  $k_+[t] > k_+[t-1]$ 
    Inicializa();
  fim se
  Atualizamedida();
  se  $N[t]$ 
    PredizInfo();
  fim se
fim função

```

---



---

```

função ATUALIZAKMAIS( )
   $k_+[t] \leftarrow 0$ ;
  para  $j \leftarrow 1$  até  $m$  faça
     $k_+[t] \leftarrow \max(k_+[t], No(u_j, t))$ ;
  fim para
fim função

```

---



---

```

função ATUALIZAKMENOS( )
   $k_-[t] \leftarrow \infty$ 
  para  $i \leftarrow 1$  até  $n$  faça
     $k_-[t] \leftarrow \min(k_-[t], No(x_i, t))$ ;
  fim para
  para  $j \leftarrow 1$  até  $m$  faça
     $k_-[t] \leftarrow \min(k_-[t], No(u_j, t))$ ;
  fim para
fim função

```

---

## 5 Aplicação: “High-throughput screening systems”

Um *high-throughput screening system* (HTS) é um conjunto de equipamentos totalmente automatizados que testam, rapidamente, compostos químicos (ver figura 1, retirada de (Brunsch, 2014)).

Ele é utilizado para testar combinações de diferentes reagentes na expectativa de encontrar aquelas que sejam promissoras para o desenvolvimento de um determinado remédio. Um teste, ou *ensaio*, é composto de um determinado número de *bateladas*. Cada *batelada* é um conjunto de *atividades* sequenciais, nos diversos equipamentos disponíveis, que um conjunto de reagentes deve passar. Várias *bateladas* podem ser processadas em paralelo, uma dada *batelada* pode passar por um dado equipamento mais de uma vez ao longo do tempo e também pode ocupar mais de um equipamento ao mesmo tempo (Brunsch, 2014).

---

**função INICIALIZA( )**

▷ Ocorreu algum disparo  
▷ Inicializa alguns valores  
▷ Valor 0 escolhido arbitrariamente  
 $N[t] \leftarrow V$ ;  
**para**  $k \leftarrow k_-[t] + 2$  **até**  $k_+[t] + 1$  **faça**  
  **para**  $i \leftarrow 1$  **até**  $n$  **faça**  
     $\hat{x}_i[t, k] \leftarrow 0$ ;  
     $\hat{x}_i[t, k].St \leftarrow I$ ;  
  **fim para**  
  **para**  $i \leftarrow 1$  **até**  $m$  **faça**  
     $\hat{u}_j[t, k] \leftarrow 0$ ;  
     $\hat{u}_j[t, k].St \leftarrow I$ ;  
  **fim para**  
**fim para**  
**fim função**

---



---

**função ATUALIZAMEDIDA( )**

▷ Atualiza valores  
**para**  $k \leftarrow k_-[t] + 1$  **até**  $k_+[t]$  **faça**  
  **para**  $i \leftarrow 1$  **até**  $n$  **faça**  
    ▷ Atualiza para estados  
    **se**  $No(x_i, t) \geq k$  **E**  $\hat{x}_i[k, t].St \neq M$   
      ▷ Informação foi medida  
      ▷ mas não atualizada  
       $\hat{x}_i[k, t] \leftarrow \hat{x}_i[k]$ ;  
       $\hat{x}_i[k, t].St \leftarrow M$ ;  
       $N[t] \leftarrow V$ ;  
    **fim se**  
  **fim para**  
  **para**  $j \leftarrow 1$  **até**  $m$  **faça**  
    ▷ Atualiza para controle  
    **se**  $No(u_j, t) \geq k$  **E**  $\hat{u}_j[k, t].St \neq M$   
      ▷ Informação foi medida  
      ▷ mas não atualizada  
       $\hat{u}_j[k, t] \leftarrow \hat{u}_j[k]$ ;  
       $\hat{u}_j[k, t].St \leftarrow M$ ;  
       $N[t] \leftarrow V$ ;  
    **fim se**  
  **fim para**  
**fim para**  
**fim função**

---

O sistema em questão está sujeito a fenômenos de sincronização. Dado uma *batelada* em que

---

**função PREDIZINFO( )**

**para**  $k \leftarrow k_-[t] + 1$  **até**  $k_+[t] + 1$  **faça**  
  ▷ Prediz, primeiramente, o estado  
  **para**  $i \leftarrow 1$  **até**  $n$  **faça**  
    **se**  $No(x_i, t) < k$   
      ▷ Necessário predizer  
       $\hat{x}_i[k, t] \leftarrow A_i \bullet \hat{x}[k - 1, t]$ ;  
       $\hat{x}_i[k, t] \leftarrow \hat{x}_i[k, t] \oplus B_i \bullet \hat{u}[k - 1, t]$ ;  
       $\hat{x}_i[k, t].St \leftarrow P$ ;  
    **fim se**  
  **fim para**  
  ▷ Agora prediz o controle  
  **para**  $j \leftarrow 1$  **até**  $m$  **faça**  
    **se**  $No(u_j, t) < k$   
      ▷ Necessário predizer  
       $\hat{u}_j[k, t] \leftarrow F_j \bullet \hat{x}[k, t]$ ;  
       $\hat{u}_j[k, t].St \leftarrow P$ ;  
    **fim se**  
  **fim para**  
**fim função**

---



Figura 1: Maquinário para um sistema de high-throughput screening (CyBio-AG)

uma atividade está sendo processada, pode ser necessário que a mesma eventualmente espere, antes de seu término, até que outra atividade seja iniciada. O modelo em GET pode ser observado na figura 2, que demonstra sincronização entre duas atividades ocorrendo em equipamentos distintos (por exemplo, a transição  $x_9$  representa uma sincronização entre *Peak Carousel* e *Robot 2*).

Algumas *restrições* devem ser respeitadas. Por exemplo, que um determinado tempo de incubação seja sempre entre 10 e 12 minutos, sob pena da perda da *batelada*. Se  $x_1[k]$  e  $x_2[k]$  representam o início e o término, respectivamente, da  $k$ -ésima incubação, tal restrição pode ser escrita como  $10 \leq x_2[k] - x_1[k] \leq 12$  ou

$$\begin{aligned} 12x_2[k] &= 12x_2[k] \oplus x_1[k]; \\ x_1[k] &= 10x_2[k] \oplus x_1[k]. \end{aligned} \quad (22)$$

Tais restrições podem ser garantidas com um *controle* judicioso do início das atividades em cada

máquina. O início das atividades também está sujeito a fatores não diretamente controláveis, como a disponibilidade da máquina. Entretanto, dado que ela está pronta para iniciar, fica a cargo da ação de controle decidir quando fazê-lo.

Dito isso, o sistema pode ser modelado como um GET, cuja dinâmica da temporização dos eventos  $\mathbf{x}[k]$  (início/término das atividades) dado as ações de controle de início de atividades  $\mathbf{u}[k]$  é descrita por (2). Tratando restrições como aquelas na equação (22) - que podem ser escritas da forma  $E\mathbf{x}[k] = D\mathbf{x}[k]$  e portanto caracterizam um semimódulo finitamente gerado  $\mathcal{S}$  - temos um problema de regulação na álgebra tropical, tal qual como o proposto na Subseção 4.1. Dessa forma, a metodologia desenvolvida pode ser aplicada.

Será utilizado como exemplo o modelo de um exemplo real de sequenciamento para HTS apresentado em (Brunsch, 2014), provido pela empresa CyBio-AG <sup>7</sup>. O modelo original presente em (Brunsch, 2014) contempla 140 eventos. Entretanto, muitos desses eventos são redundantes e podem ser eliminados para sintetizar um modelo equivalente com 69 eventos. 5 restrições diferentes são impostas nos estados. Veja (Brunsch, 2014) para detalhes. Convém ressaltar que (Brunsch, 2014) propõe um método para resolver esse problema, mas ele se diferencia do proposto neste texto em (i) não há garantia de robustez para todas as condições iniciais, (ii) a topologia do controlador gerado é muito mais complexa que a utilizada aqui (iii) o método é mais oneroso em termos computacionais (no geral).

O sistema original tem  $\rho(A) = 324$  minutos. Com  $\lambda = 325 \text{ min} > \rho(A)$ , a equação (8) pode ser resolvida em poucos segundos utilizando o Método Dual discutido em (Gonçalves et al., 2014) (veja o texto em questão para detalhes históricos do algoritmo). O controlador  $F$  resultante é não causal, então o método apresentado na Subseção 4.3 foi utilizado. O número de convergência para esse sistema,  $\kappa(A_{[\lambda]}, B_{[\lambda]}\mu\zeta^T)$ , é 3. Então, na ausência de perturbações, em no máximo 3 passos o sistema retorna ao regime de controle.

A tabela 5 mostra, para cada uma das cinco restrições, a medida de erro  $e_i[k] = |E_{i\bullet}\mathbf{x}[k] - D_{i\bullet}\mathbf{x}[k]|$ . A condição inicial utilizada foi aquela referente a política “dispare o quão mais cedo puder a partir do tempo  $t = 0$ ” <sup>8</sup>. Nota-se que a convergência ocorre em três passos, como esperado.

Ressalta-se que a topologia do GET, oriunda de um problema de otimização, pode mudar entre um ensaio e outro (Brunsch, 2014). Nesse caso, é necessário o recômputo do controlador. Dados que novos sistemas com mais de 1000

Tabela 1: Evolução do erro

$k$	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$
0	39	43	5	190	58
1	1011	0	0	1097	1030
2	1151	0	0	0	827
3	0	0	0	0	0

eventos estão sendo implementados atualmente (Brunsch, 2014), é profícuo uma metodologia que o faça rapidamente mesmo para sistemas grandes. A técnica aqui apresentada se encaixa nesse requisito, dada que a etapa crítica (resolução da equação (8)) pode ser solucionada por algoritmos pseudopolinômias.

## 6 Conclusão

Neste texto, foi apresentado um problema de controle que (a forma exata) até então não foi encontrado na literatura. Condições suficientes e computacionalmente simples foram propostas. Questões importantes, como a implementação do controlador em casos de não-causalidade, foram discutidas. Espera-se que os resultados obtidos possam contribuir na implementação prática de controladores para GETs em ambientes industriais.

Trabalhos futuros incluem analisar a restritibilidade da condição suficiente proposta. Embora os autores saibam que ela não é necessária, resultados teóricos e experimentais indicam que a condição proposta é *necessária e suficiente* para um classe importante, e caracterizável, de problemas.

## 7 Agradecimentos

Agradecemos a FAPEMIG, CAPES e CNPq pelo financiamento do projeto de pesquisa.

## Referências

- Amari, S., Demongodin, I. and Loiseau, J. (2005). Control of linear min-plus systems under temporal constraints, *44th IEEE Conference on Decision and Control and European Control Conference*.
- Amari, S., Demongodin, I. and Loiseau, J.-J. (2004). Sizing and cycle time of an industrial plant using dioid algebra, *Chapter 6 in Supply Chain Optimization, A. Dolgui, J. Soldek and O. Zaikin (Eds.), Series Applied Optimization, Springer-Verlag*, pp. 71–85.
- Amari, S., Demongodin, I., Loiseau, J., Jean-Jacques, J. and Martinez, C. (2012). Max-plus control design for temporal constraints meeting in timed event graphs, *IEEE Transactions on Automatic Control* **57**(2): 462 – 467.

<sup>7</sup><http://www.cybio-ag.com>

<sup>8</sup>Ao escrever o modelo do GET na forma  $\mathbf{x}[k+1] = A[0]\mathbf{x}[k+1] \oplus A[1]\mathbf{x}[k] \oplus A[2]\mathbf{x}[k-1]$ , essa condição inicial é  $\mathbf{x}[0] = A[0]^*\mathbf{0}$ , em que  $\mathbf{0}$  é um vetor de 0's.



- Baccelli, F., Cohen, G., Olsder, G. and Quadrat, J. (1992). *Synchronization and Linearity*, Wiley, New York.
- Brunsch, T. (2014). *Modeling and Control of Complex Systems in a Dioid Framework*, PhD thesis, TU Berlin, Berlin, Germany.
- Gaubert, S. and M.Plus (1997). Methods and applications of  $(\max,+)$  linear algebra, *Proceedings of the 14th Annual Symposium on Theoretical Aspects of Computer Science* pp. 261–282.
- Gaubert, S. and Sergeev, S. (2013). The level set method for the two-sided eigenproblem, *Discrete Event Dynamic Systems* **23**(2): 105–134.
- Gonçalves, V. M., Maia, A. and Hardouin, L. (2012). On the solution of max-plus linear equations with application on the control of timed event graphs, *Workshop on Discrete Event Systems (WODES 2012)* pp. 91–97.
- Gonçalves, V. M., Maia, A. and Hardouin, L. (2014). Solving tropical linear equations with weak dual residuations, *Linear Algebra and its Applications* pp. 69–84.
- Katz, R. D. (2007). Max-plus  $(A; B)$ -invariant spaces and control of timed discrete-event systems, *IEEE Transactions on Automatic Control* **52**(2): 229 – 241.
- Loreto, M. D., Gaubert, S., Katz, R. D. and Loiseau, J.-J. (2010). Duality between invariant spaces for max-plus linear discrete event systems, *SIAM J. Control Optim* **48**(8): 5606–5628.
- Maia, C. A. (2003). *Identificação e Controle de Sistemas a Eventos Discretos na Álgebra  $(\max,+)$* , PhD thesis, Unicamp, São Paulo, Brasil.
- Maia, C. A., Andrade, C. R. and L.Hardouin (2011). On the control of maxplus linear system subject to state restriction, *Automatica* **47**(5): 988–992.
- Maia, C. A., Hardouin, L., Santos-Mendes, R. and Loiseau, J. J. (2011). A super-eigenvector approach to control constrained max-plus linear systems, *IEEE CDC-ECC, Florida* .

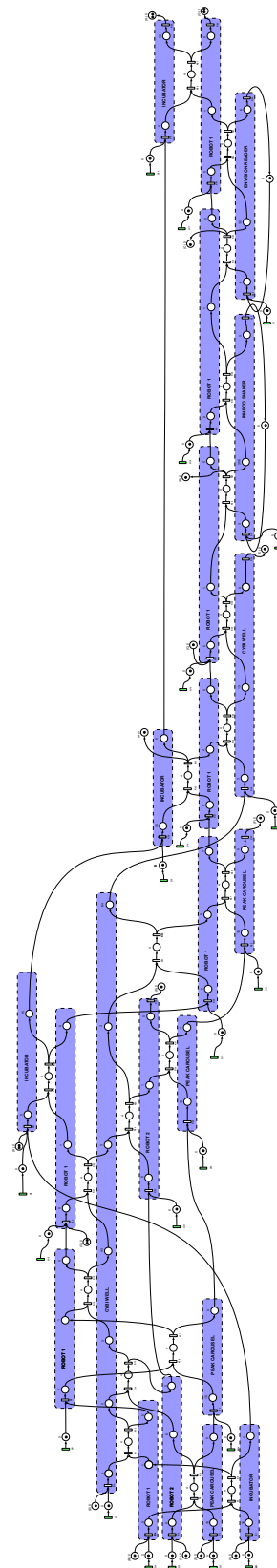


Figura 2: Grafo a evento temporizado para o sistema. Note que existem alguns lugares que não apontam e nem são apontados por uma transição. Esses lugares contêm letras (como A,B,C,...) que o remetem a outro lugar na mesma situação: são o mesmo lugar, desenhados duas vezes por simplicidade de ilustração. Assim, os dois lugares com a indicação “(A)” são na verdade o mesmo lugar.