

# Grafos Mínimos de Sistemas Dinâmicos Simbólicos

Daniel P. B. Chaves e Cecilio Pimentel

**Resumo**— Sequências que obedecem a determinadas restrições são usualmente empregadas em sistemas de gravação e transmissão de dados. Estas sequências são usualmente descritas por um grafo rotulado direcionado. A determinação do grafo com número mínimo de vértices envolve duas etapas, a determinação de um grafo inicial e a aplicação de um algoritmo de colapso de vértices. Este trabalho propõe um novo algoritmo por divisão de vértices para identificar os vértices do grafo mínimo sem precisar determinar um grafo inicial.

**Palavras-Chave**— Grafos rotulados, dinâmica simbólica, sequências com restrições.

**Abstract**— Constrained sequences have been used in many storage systems as well as in data transmission schemes. These sequences are usually described by a labeled directed graph. The labeled graph with the fewest vertices is obtained, in general, via a two-step procedure: The first step is to generate an initial graph, and the second one is to apply a vertex-minimization algorithm to identify classes of equivalent vertices. This work proposes a new algorithm based on vertex splitting to identify the vertices of the minimal graph without determining an initial graph.

**Keywords**— Labeled graphs, symbolic dynamics, constrained sequences.

## I. INTRODUÇÃO

Codificadores restritivos, também conhecidos como *codificadores moduladores* ou *codificadores de linha*, convertem sequências arbitrárias em sequências satisfazendo uma dada restrição na ocorrência de símbolos adjacentes. O objetivo de um código restritivo é melhorar o desempenho do sistema de comunicações adaptando a sequência transmitida às características do canal, contribuindo para a redução da interferência inter-simbólica e melhoria do sincronismo de símbolos.

Algumas aplicações de codificadores restritivos em sistemas práticos incluem: sequências de comprimento limitado (RLL, *RunLength-Limited*) são empregadas em drivers para CD e DVD [1]. A restrição anti-tom, que impõe um limite máximo para o comprimento de sequências periódicas, é aplicada para melhorar a temporização e algoritmos de controle de ganho em sistemas de gravação magnética [2]. Sequências com espectro limitado em  $f = 0$ , chamadas *dc-free* ou de conteúdo limitado, são empregadas em sistemas de gravação óptica para reduzir interações entre os dados gravados e o sistema de gravação [2]. A restrição *dc-free* também é usada em sistemas de comunicação onde sinais de baixa frequência são susceptíveis à distorção, o que inclui linhas de transmissão via cabo e fibra óptica. Códigos FM0 e de Miller são empregados para melhorar a eficiência espectral em sistemas de identificação via rádio frequência (RFID) [3], [4].

D. Chaves e C. Pimentel, Departamento de Eletrônica e Sistemas, Universidade Federal de Pernambuco, Recife-PE, Brasil, E-mails: daniel.chaves@ufpe.br, cecilio@ufpe.br. Este trabalho foi parcialmente financiado pelo CNPq e FACEPE.

A teoria de dinâmica simbólica, que explora as propriedades de conjuntos de sequências bi-infinitas e dos mapeamentos entre estes, vem sendo aplicada em diversos problemas abordados pela teoria de codificadores restritivos [5], bem como no projeto de sistema para codificação conjunta fonte-canal [6]. Nesta teoria destacam-se os sistemas dinâmicos simbólicos de memória finita (SFT, *shift of finite type*), caracterizado por possuir um conjunto de palavras proibidas  $\mathcal{F}$  finito. Um SFT pode ser especificado por conjuntos  $\mathcal{F}$  distintos, mas entre estes há um único conjunto  $\mathcal{O}$  com número mínimo de palavras proibidas [5, p.33]. Alternativamente, um SFT pode ser especificado por um grafo direcionado rotulado, denominado de apresentação do SFT. Essa apresentação é a entrada de vários métodos para geração de codificadores restritivos e seus respectivos decodificadores. Contudo, tais métodos têm complexidade computacional diretamente relacionada ao número de vértices da apresentação. Portanto, em casos práticos, algoritmos eficientes são necessários para geração de apresentações mínimas (com número mínimo de vértices) a partir de um conjunto de palavras proibidas.

As técnicas para geração de uma apresentação mínima são implementadas em duas etapas. Inicialmente é gerada uma apresentação determinística do SFT a partir do conjunto  $\mathcal{F}$  [5, Teorema 3.1.5], [7], [8]; seguida por um algoritmo para identificar as classes de vértices equivalentes [9] (que definirão os vértices da apresentação mínima) e as conexões entre os vértices. Existem duas classes de métodos para implementar a segunda etapa: Por divisão de vértices ou por colapso de vértices. As técnicas por divisão de vértices (e.g. [10]), de forma geral, possuem menor complexidade [9] e são baseadas em um processo sucessivo de refinamento de uma partição inicial do conjunto de vértices da apresentação.

Em [11], os autores deste trabalho propuseram um método por colapso de vértices para determinar os vértices da apresentação mínima diretamente do conjunto  $\mathcal{O}$  sem ter que construir uma apresentação inicial, como tipicamente é realizado na literatura [8], [12], [10]. Uma vantagem desta metodologia é a redução do número de operações para determinação da função de transição entre vértices, já que no método proposto em [11] esta função só é calculada para os vértices da apresentação mínima, enquanto nos métodos usuais esta é calculada para todos os vértices da apresentação inicial.

Neste trabalho apresentamos um algoritmo por divisão de vértices baseado nos conceitos propostos em [11], a saber, de identificar os vértices da apresentação mínima para depois calcular a função de transição. O algoritmo é baseado na proposta de Moore [9], com a vantagem de gerar a apresentação mínima sem precisar determinar uma apresentação inicial.

As próximas seções estão organizadas da seguinte forma. Na Seção II são estabelecidos os conceitos necessários da teoria de dinâmica simbólica. Em particular, a definição de conjuntos

de restrições e suas propriedades são revisitadas na Seção II-C. A Seção III introduz os conceitos de máscara de restrição e memória de restrição que é o mecanismo fundamental do processo de particionamento recursivo proposto na Seção IV. As conclusões deste trabalho são apresentadas na Seção V.

## II. PRELIMINARES

Nesta seção são abordados os conceitos necessários sobre teoria de dinâmica simbólica, partições e conjunto de restrição [5], [9], [11].

### A. Dinâmica Simbólica

O conjunto de todas as seqüências bi-infinita  $x = \dots x_{-2} x_{-1} x_0 x_1 \dots$  sobre um alfabeto finito  $\mathcal{A}$ , ou seja,  $x_i \in \mathcal{A}$  para  $-\infty < i < \infty$ , é denominado por  $\mathcal{A}^{\mathbb{Z}}$ . Por alfabeto finito entende-se que o conjunto  $\mathcal{A}$  é finito, com cardinalidade  $|\mathcal{A}|$ . Uma seqüência finita de símbolos consecutivos de  $\mathcal{A}$  forma uma *palavra*, comumente representada por  $\mathbf{w}$ . O conjunto  $\mathcal{A}^*$  é composto por todas as palavras finitas sobre o alfabeto  $\mathcal{A}$ , incluindo a palavra nula  $\varepsilon$ , que satisfaz  $\mathbf{w}\varepsilon = \varepsilon\mathbf{w} = \mathbf{w}$  para toda palavra  $\mathbf{w}$ . Uma palavra  $\mathbf{w}$  é um fator de um ponto  $x \in \mathcal{A}^{\mathbb{Z}}$  se há inteiros  $i \leq j$ , tal que  $\mathbf{w} = x_i x_{i+1} \dots x_j$ . Logo o *comprimento* de  $\mathbf{w}$  é igual a  $\mathbf{w} = j - i + 1$ . Uma dinâmica simbólica  $X$  é um subconjunto de  $\mathcal{A}^{\mathbb{Z}}$  formado por todos os pontos que não possuem fatores em um conjunto  $\mathcal{F}$  de palavras sobre  $\mathcal{A}$ , por isso  $\mathcal{F}$  recebe o nome de *conjunto proibido* de  $X$ . A *linguagem* de  $X$  é o conjunto  $L$  de todos os fatores dos pontos de  $X$ , incluindo a palavra vazia  $\varepsilon$ , esta última a única palavra em  $L$  de comprimento zero e que satisfaz  $\varepsilon\mathbf{w} = \mathbf{w}\varepsilon = \mathbf{w}$ . O *contexto à direita*, ou simplesmente contexto, de uma palavra  $\mathbf{w}$  é o conjunto  $F(\mathbf{w})$  de todas as palavras em  $L$  que podem seguir  $\mathbf{w}$ , i.e.,  $F(\mathbf{w}) = \{\mathbf{u} \in L : \mathbf{w}\mathbf{u} \in L\}$ . Duas palavras  $\mathbf{w}, \mathbf{u}$  com o mesmo contexto são ditas equivalentes, ou seja,  $F(\mathbf{w}) = F(\mathbf{u})$ . Uma palavra  $\mathbf{w} = w_1 \dots w_n$  é uma *proibição mínima* se  $\mathbf{w} \notin L$  ao passo que  $w_1 \dots w_{n-1}$  e  $w_2 \dots w_n$  pertencem a linguagem. O conjunto de todas as proibições mínimas de uma dinâmica simbólica é especificado por  $\mathcal{O}$ . Quando  $\mathcal{O}$  é finito, então  $X$  é de memória finita e é geralmente chamado de SFT. Se  $M + 1$  é o comprimento da palavra mais longa em  $\mathcal{O}$ , então  $X$  é de memória  $M$ .

Dados  $\mathcal{V}$  um conjunto de vértices,  $\mathcal{E}$  um conjunto de ramos e  $\mathcal{L} : \mathcal{E} \rightarrow \mathcal{A}$  uma função de rotulação, então  $G = (\mathcal{V}, \mathcal{E}, \mathcal{L})$  é um *grafo rotulado*. As funções  $i : \mathcal{E} \rightarrow \mathcal{V}$  e  $t : \mathcal{E} \rightarrow \mathcal{V}$  especificam o vértice inicial e o vértice final de um ramo, respectivamente. Um *caminho* em  $G$  é uma seqüência de ramos  $\pi = e_1 e_2 \dots e_n$ , tal que, o vértice terminal de  $e_i$  é o vértice inicial de  $e_{i+1}$ . O rótulo de  $\pi$  é a palavra  $\mathcal{L}(\pi) = \mathcal{L}(e_1)\mathcal{L}(e_2) \dots \mathcal{L}(e_n)$ . Um *percurso* em  $G$  é uma seqüência bi-infinita de ramos  $\xi = \dots e_{-1} e_0 e_1 \dots$  tal que  $t(e_i) = i(e_{i+1})$  para todo  $i$ . Uma palavra  $\mathbf{w} \in L$  é gerada por um caminho  $\pi$  em  $G$  se  $\mathbf{w} = \mathcal{L}(\pi)$ . Uma dinâmica simbólica  $X$  é representado por  $G$ , ou  $G$  é uma apresentação de  $X$ , se o rótulo de todo percurso em  $G$  é um elemento de  $X$ , com o contrário também verdadeiro. O conjunto de todas as palavras, de comprimento arbitrário, geradas a partir de caminhos em  $G$  que começam em  $I$  é o contexto de  $I$ , representado por  $F(I)$ . Dois vértices  $I, J$  são *equivalentes* se  $F(I) = F(J)$ . Um grafo  $G$  é *reduzido*

se para todo  $I, J \in \mathcal{V}$ , então  $F(I) = F(J)$  implica que  $I = J$ . Uma palavra  $\mathbf{w} \in L$  e de *sincronização* para  $G$  se todo caminho em  $G$  com rótulo  $\mathbf{w}$  termina no mesmo vértice. Uma apresentação  $G$  é *determinística* se para todo  $e_1, e_2 \in \mathcal{E}$  tal que  $i(e_1) = i(e_2)$  e  $\mathcal{L}(e_1) = \mathcal{L}(e_2)$ , então  $e_1 = e_2$ . Se para todo vértice de  $G$  pode ser associada uma palavra de sincronização, então  $G$  é sincronizável. Se uma apresentação  $G$  é determinística e sincronizável, então ela é um SDP de  $X$ . Caso  $G$  também seja reduzida, então ela é o m-SDP de  $X$ .

### B. Partições

Uma *partição*  $\mathcal{P}$  de um conjunto  $\mathcal{E}$  é uma família de subconjuntos não vazios e disjuntos de  $\mathcal{E}$ , portanto  $\mathcal{E} = \bigcup_{P \in \mathcal{P}} P$ . O que torna o conceito de partição importante é sua dualidade com o conceito de *relação de equivalência*. Toda partição define uma relação de equivalência, e o contrário também é verdadeiro.

Sejam  $\mathcal{Q}$  e  $\mathcal{P}$  duas partições do conjunto  $\mathcal{E}$ . Então  $\mathcal{Q}$  é um *refinamento* de  $\mathcal{P}$ , ou  $\mathcal{P}$  é um *adensamento* de  $\mathcal{Q}$ , se cada subconjunto de  $\mathcal{Q}$  está contido em algum subconjunto de  $\mathcal{P}$ , o que é representado por  $\mathcal{Q} \leq \mathcal{P}$ . A mais densa partição que refina  $\mathcal{P}$  e  $\mathcal{Q}$ , denominada de  $\mathcal{P} \wedge \mathcal{Q}$ , é gerada pelos conjuntos não nulos  $P \cap Q$ , para todo  $P \in \mathcal{P}$  e todo  $Q \in \mathcal{Q}$ . Esta notação é naturalmente estendida para uma coleção de partições, de forma que  $\mathcal{P} = \mathcal{P}_1 \wedge \dots \wedge \mathcal{P}_n$  determina o refinamento mais denso comum de  $\mathcal{P}_1, \dots, \mathcal{P}_n$ . Se  $n = 0$  então  $\mathcal{P}$  é o particionamento universal de  $\mathcal{E}$  e só contém este subconjunto.

### C. Conjunto de Restrições

Seguem conceitos adicionais da teoria de linguagens formais. O *conjunto de prefixos* de uma palavra  $\mathbf{w} = w_1 w_2 \dots w_n$  em  $\mathcal{A}^*$  é representado por  $\mathcal{P}(\mathbf{w}) = \{w_1 \dots w_j \mid 1 \leq j \leq n\} \cup \{\varepsilon\}$ . De forma similar, o *conjunto de sufixos* de  $\mathbf{w}$  é  $\mathcal{S}(\mathbf{w}) = \{w_j \dots w_n \mid 1 \leq j \leq n\} \cup \{\varepsilon\}$ . Assim, seja  $\mathbf{w} = abbc$ , então  $\mathcal{P}(\mathbf{w}) = \{\varepsilon, a, ab, abb, abbc\}$  e  $\mathcal{S}(\mathbf{w}) = \{\varepsilon, c, bc, bbc, abbc\}$ . A *extensão* de uma palavra  $\mathbf{w} \in \mathcal{A}^*$  por uma palavra  $\mathbf{v} \in \mathcal{A}^*$  é igual a  $\mathbf{w}\mathbf{v}$ . Dado  $B_1, B_2 \subseteq \mathcal{A}^*$ , tem-se que  $B_1 B_2^{-1} = \{w_1 \in \mathcal{A}^* \mid w_1 w \in B_1 \text{ para algum } w \in B_2\}$ , de forma similar  $B_1^{-1} B_2 = \{w_2 \in \mathcal{A}^* \mid w w_2 \in B_2 \text{ para algum } w \in B_1\}$ .

O contexto de uma palavra pode ser descrito de forma compacta através do conceito de *conjunto de restrição* [11].

**Definição 1.** [11, Definição 3] *Seja  $L$  uma linguagem sobre  $\mathcal{A}$  com conjunto de restrições mínimas  $\mathcal{O}$ . O conjunto de restrições  $\mathcal{C}(\mathbf{w})$  de uma palavra  $\mathbf{w} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  é formado pelos sufixos  $\mathbf{v} \in \mathcal{S}(\mathcal{A}^{-1}\mathcal{O})$ , tal que,  $\mathbf{w}\mathbf{v} \notin L$ , mas  $\mathbf{w}\mathbf{u} \in L$  para todo prefixo  $\mathbf{u} \in \mathcal{P}(\mathcal{v}\mathcal{A}^{-1})$ . Os elementos de  $\mathcal{C}(\mathbf{w})$  são denominados restrições de  $\mathbf{w}$ .*

Pela Definição 1, para todo  $\mathbf{w} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  e  $\mathbf{v} \in \mathcal{C}(\mathbf{w})$ , a palavra  $\mathbf{w}\mathbf{v}$  deve possuir um, e um único, fator em  $\mathcal{O}$ . Portanto, há  $\mathbf{v}_1 \in \mathcal{S}(\mathbf{w})$  e  $\mathbf{v}_2 \in \mathcal{P}(\mathbf{v})$ , tal que,  $(\mathbf{v}_1 \mathbf{v}_2) \in \mathcal{O}$ . Assim,  $\mathcal{C}(\mathbf{w})$  é formado por palavras  $\mathbf{u}$  que quando concatenadas com algum  $\mathbf{v} \in \mathcal{S}(\mathbf{w})$  geram uma palavra em  $\mathcal{O}$ , mas o mesmo não ocorre com os prefixos próprios de  $\mathbf{u}$ .

**Exemplo 1.** *Seja  $\mathcal{O} = \{aac, baa, babb, babc, bbb, bbc, ca, cbb, cbc\}$  o conjunto de proibições mínimas e  $\mathbf{w} = ba$  uma palavra*

em  $\mathcal{P}(\mathcal{O}A^{-1})$ . As palavras  $a, ac, baa, bb, bc$  são possíveis elementos de  $\mathcal{C}(\mathbf{w})$ , pois quando concatenadas com  $ba$  geram palavras com fatores  $aac, baa, babb, babc \in \mathcal{O}$ , respectivamente. Já que nenhum prefixo dos demais elementos em  $\mathcal{O}$  é sufixo de  $\mathbf{w}$ , o conjunto  $\mathcal{C}(\mathbf{w})$  é determinado eliminando-se a redundância do conjunto  $\{a, ac, baa, bb, bc\}$ . Uma vez que só  $ac$  possui um prefixo próprio neste conjunto, a palavra  $a$ , então  $\mathcal{C}(\mathbf{w}) = \{a, baa, bb, bc\}$ .

Como estabelece o próximo teorema, a coleção de conjuntos de restrições de uma linguagem  $L$  específica, de forma compacta, o conjunto de contextos possíveis na linguagem.

**Teorema 1.** [11, Teorema 5] *Sejam  $\mathbf{w}, \mathbf{w}' \in L$ , então  $F(\mathbf{w}) = F(\mathbf{w}')$  se, e somente se,  $\mathcal{C}(\mathbf{w}) = \mathcal{C}(\mathbf{w}')$ .*

Da proposição a seguir, a coleção de conjuntos de restrições em uma linguagem é determinado pelo conjunto  $\mathcal{P}(\mathcal{O}A^{-1})$ .

**Proposição 2.** [11, Lema 7] *Seja  $\mathbf{w} \in L$  e  $\mathbf{v}$  seu sufixo mais longo em  $\mathcal{P}(\mathcal{O}A^{-1})$ , então  $\mathcal{C}(\mathbf{w}) = \mathcal{C}(\mathbf{v})$ .*

Os resultados desta subseção permitem a construção de um grafo de contextos (um m-SDP de um SFT),  $G = (\mathcal{V}, \mathcal{E}, \mathcal{L})$ , para um sistema dinâmico simbólico. Em  $G$  o conjunto de vértices é associado aos possíveis conjuntos de contextos, ou seja,  $\mathcal{V} = \{F(\mathbf{w}) \mid \mathbf{w} \in L\}$ . Para finalizar a construção, há um ramo com rótulo  $a$  do vértice  $I$  para o vértice  $J$  se, e somente se,  $I = F(\mathbf{w})$  e  $J = F(\mathbf{wa})$ . Como conjunto de restrições e conjunto de contexto são conceitos duais, a construção de  $G$  pode ser realizada a partir do conjunto  $\{\mathcal{C}(\mathbf{w}) \mid \mathbf{w} \in L\}$  [11], com uma transição de rótulo  $a$  de  $\mathcal{C}(\mathbf{w})$  para  $\mathcal{C}(\mathbf{wa})$ .

### III. MEMÓRIA DE RESTRIÇÃO

O conjunto  $\mathcal{P}(\mathcal{O}A^{-1})$  contém a informação necessária para especificar o conjunto de restrições para qualquer palavra em  $L$  (vide Proposição 2). Ademais, os conjuntos de restrições nos permitem dizer quando palavras possuem ou não o mesmo contexto (vide Teorema 1). Nesta seção abordaremos inicialmente a determinação do máximo sufixo próprio em  $\mathcal{P}(\mathcal{O}A^{-1})$  de uma palavra  $\mathbf{w}$  em  $\mathcal{P}(\mathcal{O}A^{-1})$ . Em seguida, essa informação é empregada na Definição 2 e na Definição 3 para calcular a máscara de restrição e a memória de restrição.

A determinação dos máximos sufixos próprios pode ser realizada construindo-se uma árvore rotulada com as palavras em  $\mathcal{P}(\mathcal{O}A^{-1})$ . De cada nó desta árvore divergem ramos rotulados com elementos de  $\mathcal{A}$  distintos e o rótulo de um nó é o rótulo do caminho que parte deste nó e termina no nó raiz. Este árvore deve ter o menor número de ramos para que os elementos de  $\mathcal{P}(\mathcal{O}A^{-1})$  sejam rótulos de seus nós. Dessa forma, os rótulos dos nós da árvore correspondem aos possíveis sufixos das palavras em  $\mathcal{P}(\mathcal{O}A^{-1})$ . Os rótulos das folhas e de alguns nós intermediários correspondem aos elementos em  $\mathcal{P}(\mathcal{O}A^{-1})$  e esses nós devem ser marcados com '×', enquanto os demais são marcados com '•'. Essa árvore constitui uma forma algorítmica simples de determinar o máximo sufixo próprio de uma palavra em  $\mathcal{P}(\mathcal{O}A^{-1})$ . Para isto, basta armazenar o rótulo associado ao último nó intermediário com marca × encontrado ao percorrer na árvore o caminho associado a  $\mathbf{w}$ . Considerando o conjunto  $\mathcal{O}$  do Exemplo 1, a árvore obtida é

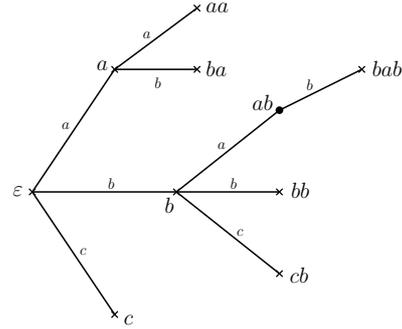


Fig. 1. Árvore para determinação dos máximos sufixos próprios.

apresentada na Fig. 1 a partir do conjunto de prefixos próprios  $\mathcal{P}(\mathcal{O}A^{-1}) = \{\varepsilon, a, b, c, aa, ba, bab, bb, cb\}$ . Para determinar o máximo sufixo próprio de  $bab$  observa-se nesta árvore que o nó precedente de rótulo  $ab$  está marcado com '•', porque a palavra  $ab$  não pertence a  $\mathcal{P}(\mathcal{O}A^{-1})$ . Por outro lado, o nó com rótulo  $b$  é o primeiro nó precedente com marcador ×, então  $b$  é o máximo sufixo próprio de  $bab$  em  $\mathcal{P}(\mathcal{O}A^{-1})$ .

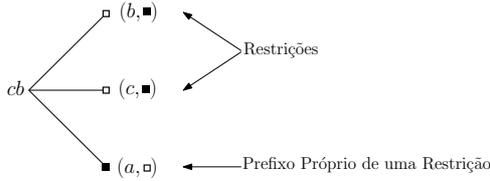
A máscara de restrição define as restrições ou prefixos próprios de restrições de uma palavra em  $\mathcal{P}(\mathcal{O}A^{-1})$ , ambos de comprimento um. Informação que permite identificar se duas palavras podem ser equivalentes. Para isso é necessário que suas máscaras de restrição sejam iguais, caso contrário, uma palavra possui uma restrição de comprimento um que a outra não possui ou uma restrição que possui um determinado símbolo como prefixo próprio e que a outra não possui. Na Definição 2 os símbolos ■ e □ são marcadores que identificam quando a extensão de uma palavra gera uma palavra proibida, ou o prefixo próprio de uma.

**Definição 2.** *Dada uma palavra  $\mathbf{w} \in \mathcal{P}(\mathcal{O}A^{-1})$ , sua máscara de restrição  $\mathcal{M}(\mathbf{w})$  é um subconjunto de  $\mathcal{A} \times \{\square, \blacksquare\}$ , construído sob os critérios:*

- Seja  $a \in \mathcal{A}$  uma restrição de um sufixo de  $\mathbf{w}$ , ou seja,  $\mathbf{wa} \notin L$ , então  $(a, \blacksquare) \in \mathcal{M}(\mathbf{w})$ ;*
- Seja  $a \in \mathcal{A}$  um prefixo próprio de uma restrição  $\mathbf{u}$  de  $\mathbf{w}$ , ou seja,  $a \in \mathcal{P}(\mathbf{u}) \setminus \{\mathbf{u}\}$  e  $\mathbf{wu} \notin L$ , contudo  $\mathbf{wa} \in L$ . Então  $(a, \square) \in \mathcal{M}(\mathbf{w})$ ;*
- Se não existe uma restrição  $\mathbf{u}$  de  $\mathbf{w}$ , tal que,  $a \in \mathcal{P}(\mathbf{u})$ , então  $a$  não possui informação relevante para determinação do conjunto de restrições de  $\mathbf{w}$ .*

A máscara de restrição de  $\mathbf{w} \in \mathcal{P}(\mathcal{O}A^{-1})$  é formada por dois grupos de contribuições: 1) O grupo 1 está associado aos prefixos de comprimento um das palavras em  $\mathbf{w}^{-1}\mathcal{O}$ ; 2) O grupo 2 de elementos da máscara de restrição do máximo sufixo próprio de  $\mathbf{w}$  em  $\mathcal{P}(\mathcal{O}A^{-1})$ . Para determinação de  $\mathcal{M}(\mathbf{w})$ , as contribuições do grupo 1 são calculadas primeiro, seguidas pelas do grupo 2. O grupo 2 é formado por pares cujos símbolos não aparecem nos pares do grupo 1. Assim, existe uma ordem para o cálculo da máscara de restrição, que começa com as palavras em  $\mathcal{P}(\mathcal{O}A^{-1})$  de menor comprimento para as de maior comprimento.

**Exemplo 2** (Continuação do Exemplo 1). *Considerando o prefixo  $\mathbf{w} = b$  em  $\mathcal{P}(\mathcal{O}A^{-1})$ . O conjunto  $\mathcal{M}(b) = \{(a, \square), (b, \square)\}$  só possui elementos do grupo 1, já que seu máximo sufixo em*


 Fig. 2. Representação da máscara de restrição de  $cb$ .

$\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  é  $\varepsilon$ , para o qual  $\mathcal{M}(\varepsilon) = \emptyset$ . Sabendo  $\mathcal{M}(b)$ , podemos calcular a máscara de restrição para qualquer palavra em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  que tenha-o como máximo sufixo. Escolhendo a palavra  $\mathbf{w} = cb$ , então  $\{(b, \blacksquare), (c, \blacksquare)\} \subseteq \mathcal{M}(cb)$ , decorrentes do grupo 1. Agora, considerando o grupo 2, o par  $(a, \square)$  será a única contribuição do desse grupo a  $\mathcal{M}(cb)$ , já que  $a$  é o único símbolo que aparece nos pares em  $\mathcal{M}(b)$ , mas não nos de  $\mathcal{M}(cb)$ . A Fig. 2 apresenta a representação gráfica de  $\mathcal{M}(cb)$ .

Para um conjunto  $B \subseteq \mathcal{A}^*$ , tal que, para todos  $\mathbf{u}, \mathbf{w} \in B$  verifica-se que  $\mathcal{M}(\mathbf{u}) = \mathcal{M}(\mathbf{w})$ , sem ambiguidade, podemos afirmar que  $\mathcal{M}(B) = \mathcal{M}(\mathbf{w})$ , como uma extensão do conceito de máscara de restrição.

**Definição 3.** Dada uma palavra  $\mathbf{w} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  e um símbolo  $a \in \mathcal{A}$ , se  $\mathbf{w}a \in L$ , a memória de restrição  $\mathcal{R}(\mathbf{w}a)$  da extensão de  $\mathbf{w}$  por  $a$  é o máximo sufixo de  $\mathbf{w}a$  em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ . Portanto,  $\mathcal{R}(\mathbf{w}a)$  especifica um elemento  $\mathbf{v} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  que satisfaz  $\mathcal{C}(\mathbf{w}a) = \mathcal{C}(\mathbf{v})$ .

A memória de restrição não é definida para  $(a, \blacksquare) \in \mathcal{M}(\mathbf{w})$ , pois  $\mathbf{w}a \notin L$  e neste caso,  $\mathcal{C}(\mathbf{w}a) \neq \mathcal{C}(\mathbf{v})$ , para todo  $\mathbf{v} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ . Para  $(a, \square) \in \mathcal{M}(\mathbf{w})$ , há três situações que podem ocorrer ao calcular a memória de restrição: 1) Quando  $\mathbf{w} = \varepsilon$ ,  $\mathcal{R}(\varepsilon a) = a$ ; 2)  $\mathcal{R}(\mathbf{w}a) = \mathbf{w}a$  quando  $\mathbf{w}a \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ ; 3)  $\mathcal{R}(\mathbf{w}a) = \mathbf{u}$  onde  $\mathbf{u}$  é o máximo sufixo próprio de  $\mathbf{w}a$  em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ , que é igual a memória de restrição do máximo sufixo próprio de  $\mathbf{w}$  estendido por  $a$  em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ . A memória de restrição do máximo sufixo próprio de  $\mathbf{w}$  estendido por  $a$  já terá sido calculada sempre que a memória de restrição for determinada das palavras de menor comprimento para as de maior comprimento. Com isso, a complexidade computacional para determinação da memória de restrição é limitada pela complexidade do caso 2, para o qual há algoritmos cuja complexidade é linear com o número de palavras em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ .

**Exemplo 3** (Continuação do Exemplo 2). Ao estender  $\mathbf{w} = cb$  por  $a$  obtém-se  $\mathcal{R}(cba) = ba$ . Como  $cba \notin \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ , o cálculo de  $\mathcal{R}(cba)$  refere-se ao terceiro caso. Pela Fig. 1, o máximo sufixo próprio de  $cb$  em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  é a palavra  $b$ . Contudo, se o valor da memória de restrição é determinado das palavras de menor para as de maior comprimento em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ , então  $\mathcal{R}(ba)$  já foi determinado em uma etapa anterior. De acordo com o segundo caso,  $\mathcal{R}(ba) = ba$  uma vez que  $ba \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ .

Na definição a seguir, a memória de restrição é empregada para refinar as partições na etapa recursiva do algoritmo que será apresentado na Seção IV.

 TABELA I  
ALGORITMO DE DIVISÃO PROPOSTO.

1. $\text{ALG}(\mathcal{P}(\mathcal{O}\mathcal{A}^{-1}))$	
2. $n \leftarrow 1$	▷ Início da Etapa 1
3. <b>choose</b> $\mathbf{w} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$	
4. $P_n \leftarrow \{\mathbf{w}\}$	
5. <b>for all</b> $\mathbf{w} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ <b>do</b>	
6. $i \leftarrow 1$	
7. <b>repeat</b>	
8. <b>if</b> $\mathcal{M}(\mathbf{w}) = \mathcal{M}(P_i)$ <b>do</b>	
9. $P_i \leftarrow P_i \cup \{\mathbf{w}\}$	
10. <b>ifelse</b>	
11. $i \leftarrow i + 1$	
12. <b>if</b> $i = n + 1$ <b>do</b>	
13. $n \leftarrow n + 1$	
14. $P_n \leftarrow \{\mathbf{w}\}$	
15. <b>until</b> $\mathcal{M}(\mathbf{w}) = \mathcal{M}(P_i)$	
16. $\mathcal{P} \leftarrow \{P_1, P_2, \dots, P_n\}$	▷ Partição inicial
17. <b>repeat</b>	▷ Início da Etapa 2
18. $\mathcal{P}' \leftarrow \mathcal{P}$	▷ $\mathcal{P}'$ : Partição precedente
19. <b>for all</b> $a \in \mathcal{A}$ <b>do</b>	
20. $\mathcal{P}_a \leftarrow \bigwedge_{P \in \mathcal{P}} (P, a)   \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$	
21. $\mathcal{P} \leftarrow \mathcal{P} \wedge \bigwedge_{a \in \mathcal{A}} \mathcal{P}_a$	▷ $\mathcal{P}$ : Partição atual
22. <b>until</b> $\mathcal{P} = \mathcal{P}'$	▷ Etapa 3

**Definição 4.** Seja  $P \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  e  $a \in \mathcal{A}$ , a secção de  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})a$  com relação a  $P$  é definido por:

$$P \setminus \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})a = \{\mathbf{w} \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1}) \mid \mathbf{w}a \in L \text{ e } \mathcal{R}(\mathbf{w}a) \in P\}.$$

A partição de  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  com relação a  $P$  e pela extensão por  $a$  é especificada por  $(P, a) | \mathcal{P}(\mathcal{O}\mathcal{A}^{-1}) = \{P_1, P_1^c\}$ , em que  $P_1 = P \setminus \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})a$  e  $P_1^c$  é o complemento de  $P_1$  com relação ao conjunto de prefixos  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ .

Como  $P_1$  e  $P_1^c$  formam uma partição de  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ , e além disso  $\varepsilon \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ , então a partição proposta na Definição 4 é bem definida.

#### IV. ALGORITMO DE DIVISÃO PARA IDENTIFICAÇÃO DE VÉRTICES DE UM m-SDP

Esta seção descreve um algoritmo por divisão para particionar  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ , tal que, palavras pertencem a uma mesma classe se, e somente se, possuem o mesmo contexto (ou, analogamente, possuem o mesmo conjunto de restrições). Para isso não é necessário listar esses conjuntos, basta saber a máscara de restrição e a memória de restrição das palavras em  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ . O algoritmo é dividido essencialmente em três etapas, que são descritas a seguir.

**Etapa 1. Particionamento por máscara de restrição:** Esta etapa determina a partição inicial  $\mathcal{P} = \{P_1, P_2, \dots, P_n\}$  que satisfaz a seguinte condição. Dadas duas palavras  $\mathbf{w}_i \in P_i$  e  $\mathbf{w}_j \in P_j$ , em que  $P_i$  e  $P_j$  são subconjuntos da partição inicial, então  $i = j \Leftrightarrow \mathcal{M}(\mathbf{w}_i) = \mathcal{M}(\mathbf{w}_j)$ . Essa etapa corresponde as linhas 2 a 16 do algoritmo da Tabela I.

**Etapa 2. Particionamento por memória de restrição:** Nesta etapa o particionamento proposto na Definição 4 é empregado para refinar a partição  $\mathcal{P}$  gerada na Etapa 1. Como discutido na Seção III, as palavras em  $P \in \mathcal{P}$  só são estendidas por um símbolo  $a \in \mathcal{A}$  se  $(a, \square) \in \mathcal{M}(P)$ . Na linha 20 do algoritmo

TABELA II  
MÁSCARAS DE RESTRIÇÃO.

$w \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$	Máscara de Restrição
$\varepsilon$	$\emptyset$
$a$	$\{(a, \square)\}$
$b$	$\{(a, \square), (b, \square)\}$
$c$	$\{(b, \square), (a, \blacksquare)\}$
$aa$	$\{(a, \square), (c, \blacksquare)\}$
$ba$	$\{(b, \square), (a, \blacksquare)\}$
$bb$	$\{(a, \square), (b, \blacksquare), (c, \blacksquare)\}$
$cb$	$\{(a, \square), (b, \blacksquare), (c, \blacksquare)\}$
$bab$	$\{(a, \square), (b, \blacksquare), (c, \blacksquare)\}$

 TABELA III  
PARTIÇÃO PELA ETAPA 1.

$a \in \mathcal{A}$	Partição
$a$	$\{\varepsilon\}, \{c, ba\}, \{b, a, aa, bb, cb, bab\}$
$b$	$\{\varepsilon\}, \{c, ba\}, \{a, aa\}, \{b\}, \{bb, cb, bab\}$
$c$	$\{\varepsilon\}, \{c, ba\}, \{a\}, \{aa\}, \{b\}, \{bb, cb, bab\}$

da Tabela I gera-se uma partição de  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  com relação a todo  $P \in \mathcal{P}$  pela extensão por um símbolo específico  $a \in \mathcal{A}$ , já na linha 21 a partição resultante desse processo refina  $\mathcal{P}$ .

*Etapa 3. Critério de parada:* O critério de parada do algoritmo é estabelecido quando a partição obtida em uma iteração da etapa recursiva (Etapa 2) não puder ser mais refinada, ou seja, quando uma partição for igual a sua subsequente. Essa etapa corresponde a linha 22 do algoritmo da Tabela I.

#### A. Exemplo

Seja  $\mathcal{O}$  o conjunto de restrições do Exemplo 1. Inicialmente, para todo  $w \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$  calcula-se o conjunto  $\mathcal{M}(w)$ . Para tornar esse etapa computacionalmente mais eficiente, esses conjuntos são calculados a partir dos prefixos de menor para os de maior comprimento. O resultado dessa etapa é apresentado na Tabela II.

Definidas as máscaras de restrição, segue o particionamento inicial descrito pela Etapa 1. Iniciando com  $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ , as partições são refinadas de forma sequencial para cada  $a \in \mathcal{A}$ , tal que ao final de cada refinamento, toda palavra  $w$  de um elemento da partição satisfaz-se apenas um dos três casos:  $(a, \square) \in \mathcal{M}(w)$ , ou  $(a, \blacksquare) \in \mathcal{M}(w)$ , ou nenhum dos casos anteriores. O resultado da partição sequencial para cada símbolo é apresentado na Tabela III, sendo a partição  $\mathcal{P} = \{\{\varepsilon\}, \{c, ba\}, \{a\}, \{aa\}, \{b\}, \{bb, cb, bab\}\}$  a entrada do processo iterativo do algoritmo, que corresponde a Etapa 2.

Como os singletons de  $\mathcal{P}$  não são considerados na Etapa 2, só precisamos analisar os subconjuntos  $P_2 = \{c, ba\}$  e  $P_6 = \{bb, cb, bab\}$ . Como devemos apenas considerar extensões associados ao marcador  $\square$ , deve-se analisar a extensão de  $P_2$  pelo símbolo  $b$  e a extensão de  $P_6$  por  $a$ . Por exemplo, a extensão de  $P_6$  pelo símbolo  $a$ , gera o conjunto  $\{bba, cba, baba\}$ . No entanto,  $\mathcal{R}(bba) = \mathcal{R}(cba) = \mathcal{R}(baba) = ba$ , o que é um elemento de  $P_2$ . Analogamente, as memórias de restrição das extensões dos elementos de  $P_2$  por  $b$  estão em  $P_6$ . Como não houve refinamento da partição, o critério de parada da Etapa 3 é estabelecido. Cada elemento desta partição é um vértice do grafo de contextos. A Figura 3 apresenta o grafo obtido ao conectar-se os vértices do grafo, contudo, eliminando aqueles que só são vértices terminais ou vértices iniciais, além dos

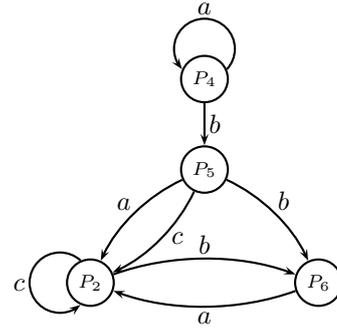


Fig. 3. Grafo de contextos construído da partição  $\mathcal{P}$  gerada pelo algoritmo da Tabela I. Cada vértice do grafo está associado aos seguintes elementos da partição,  $P_2 = \{c, ba\}$ ,  $P_4 = \{aa\}$ ,  $P_5 = \{b\}$ ,  $P_6 = \{bb, cb, bab\}$ .

respectivos ramos, já que esta operação não altera a linguagem do sistema dinâmico simbólico apresentado [5].

## V. CONCLUSÕES

O algoritmo apresentado estabelece um novo paradigma para aplicação dos conjuntos de restrição na determinação de apresentações mínimas. Neste trabalho foi apresentado uma proposta baseada no algoritmo de Moore [9], que possui complexidade  $\mathcal{O}(n^2)$ , em que  $n$  é o número de vértices da apresentação inicial. Uma proposta para a continuidade deste trabalho é apresentar um algoritmo baseado na técnica de Hopcroft [10] que alcançará a menor complexidade para algoritmos gerais de minimização  $\mathcal{O}(n \log n)$ , com a vantagem de não precisar de uma apresentação inicial, o que reduz o número de transições entre vértices a ser estabelecida. Conjectura-se que essa proposta terá a menor complexidade entre os algoritmos conhecidos.

## REFERÊNCIAS

- [1] K. Imminck, "Efimplus: the coding format of the multimedia compact disc," *Consumer Electronics, IEEE Transactions on*, vol. 41, no. 3, pp. 491–497, 1995.
- [2] B. H. Marcus, R. M. Roth, and P. H. Siegel, "Constrained systems and coding for recording channels," in *Handbook of Coding Theory* (V. S. Pless and W. Huffman, eds.), vol. 2, pp. 1635–1764, Eds. Amsterdam:Elsevier, 1999.
- [3] J. Park and T.-J. Lee, "Channel-aware line code decision in rfid," *Communications Letters, IEEE*, vol. 15, no. 12, pp. 1402–1404, 2011.
- [4] M. Simon and D. Divsalar, "Some interesting observations for certain line codes with application to rfid," *Communications, IEEE Transactions on*, vol. 54, no. 4, pp. 583–586, 2006.
- [5] D. Lind and B. H. Marcus, *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, 1995.
- [6] S. Ben-Jamaa, C. Weidmann, and M. Kieffer, "Analytical tools for optimizing the error correction performance of arithmetic codes," *Communications, IEEE Transactions on*, vol. 56, no. 9, pp. 1458–1468, 2008.
- [7] N. T. Sindhushayana, "Symbolic dynamics, automata theory and the theory of coding: A comparative study and applications," Master's thesis, Cornell University, 1992.
- [8] M. Crochemore, F. Mignosi, and A. Restivo, "Automata and forbidden words," *Information Processing Letters*, no. 67, pp. 111–117, 1998.
- [9] J. Berstel, L. Boasson, O. Carton, and I. Fagnot, "Minimization of automata," *CoRR*, vol. abs/1010.5318, 2010.
- [10] J. E. Hopcroft, "An  $(n \cdot \log n)$  algorithm for minimizing states in a finite automaton," Tech. Rep. STAN-CS-71-190, Stanford University, 1971.
- [11] D. P. B. Chaves and C. Pimentel, "A new algorithm for finding the shannon cover of shifts of finite type," in *Proceedings of the VI International Telecommunications Symposium*, (Fortaleza, CE, Brazil), pp. 694–699, 3-6 Sept. 2006.
- [12] M. V. Lawson, *Finite Automata*. Chapman & Hall/CRC, 2004.