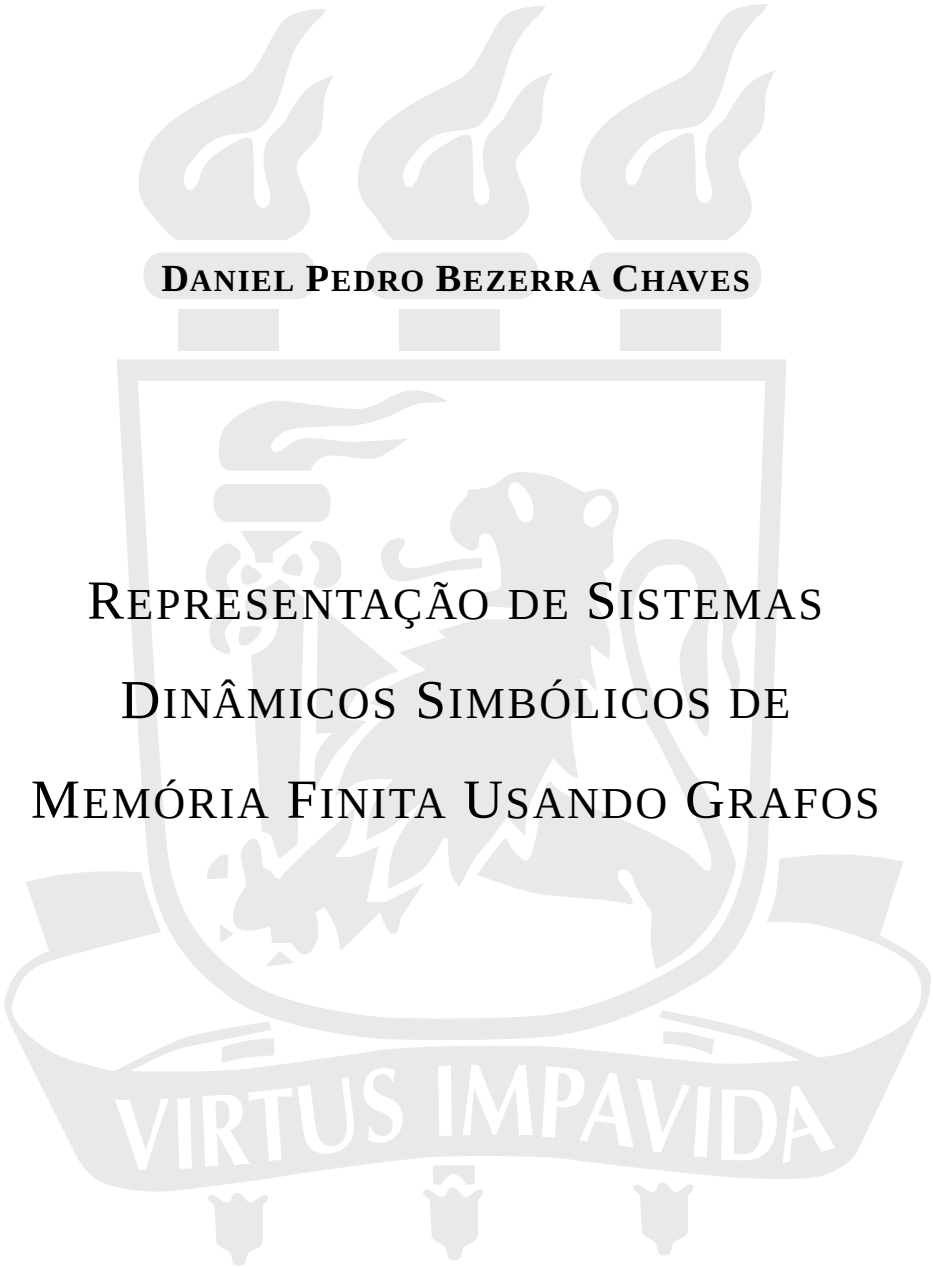
UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE TECNOLOGIA E GEOCIÊNCIAS

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



DANIEL PEDRO BEZERRA CHAVES

REPRESENTAÇÃO DE SISTEMAS DINÂMICOS SIMBÓLICOS DE MEMÓRIA FINITA USANDO GRAFOS

RECIFE, SETEMBRO DE 2006.

DANIEL PEDRO BEZERRA CHAVES

REPRESENTAÇÃO DE SISTEMAS DINÂMICOS SIMBÓLICOS DE MEMÓRIA FINITA USANDO GRAFOS

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco como parte dos requisitos para obtenção do grau de **Mestre em Engenharia Elétrica**

ORIENTADOR: PROF. CECILIO JOSÉ LINS PIMENTEL, PH.D.

Recife, Setembro de 2006.

C512r **Chaves, Daniel Pedro Bezerra.**

Representação de sistemas dinâmicos simbólicos de memória finita usando grafos. – Recife: O Autor, 2006.

74 folhas. : il. ; fig., tabs.

Dissertação (Mestrado) – Universidade Federal de Pernambuco. CTG. Engenharia Elétrica, 2006.

Inclui Bibliografia.

1. Engenharia elétrica. 2. Comunicações – Teoria da codificação. 3. Códigos de linha. 4. Grafos direcionados. I. Título.

UFPE

621.3CDD (22.ed.) **BCTG/2006-111**

Universidade Federal de Pernambuco

Pós-Graduação em Engenharia Elétrica

PARECER DA COMISSÃO EXAMINADORA DE DEFESA DE
TESE DE MESTRADO ACADÊMICO DE

DANIEL PEDRO BEZERRA CHAVES

TÍTULO

"REPRESENTAÇÃO DE SISTEMAS DINÂMICOS SIMBÓLICOS DE MEMÓRIA FINITA USANDO GRAFOS"

A comissão examinadora composta pelos professores: CECÍLIO JOSÉ LINS PIMENTEL, DES/UFPE, RICARDO MENEZES CAMPELLO DE SOUZA, DES/UFPE e REGINALDO PALAZZO JÚNIOR, DT/UNICAMP, sob a presidência do primeiro, consideram o candidato **DANIEL PEDRO BEZERRA CHAVES** **APROVADO**.

Recife, 18 de setembro de 2006.

JOAQUIM FERREIRA MARTINS FILHO
Coordenador do PPGEE

CECÍLIO JOSÉ LINS PIMENTEL
Orientador e Membro Titular Interno

REGINALDO PALAZZO JÚNIOR
Membro Titular Externo

RICARDO MENEZES CAMPELLO DE SOUZA
Membro Titular Interno

Aos meus pais e irmãs

DEDICO

AGRADECIMENTOS

A Deus pela oportunidade de aprender.

Aos meus pais pelo apoio constante e incondicional.

Ao Prof. Dr. Cecilio Pimentel pela convivência enriquecedora, confiança no trabalho realizado e orientação.

Aos Professores do DES-UFPE, principalmente aos do CODEC, pela disposição de dividirem seus conhecimentos conosco.

À comissão julgadora pela atenção despendida ao trabalho.

Aos amigos com os quais convivi durante o Programa de Mestrado em Engenharia Elétrica.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo apoio financeiro (Processo 134157/2004-4) concedido durante o período de outubro de 2004 a outubro de 2006, sem o qual não seria possível a realização do Programa de Mestrado em Engenharia Elétrica.

DANIEL PEDRO BEZERRA CHAVES

Universidade Federal de Pernambuco

18 de Setembro de 2006

Resumo da Dissertação apresentada à UFPE como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica

REPRESENTAÇÃO DE SISTEMAS DINÂMICOS SIMBÓLICOS DE MEMÓRIA FINITA USANDO GRAFOS

Daniel Pedro Bezerra Chaves

Setembro/2006

Orientador: Prof. Cecilio José Lins Pimentel, Ph.D.
Área de Concentração: Comunicações
Palavras-chaves: dinâmica simbólica, grafos direcionados, autômatos, códigos de linha, sistemas com restrição
Número de páginas: 74

Nesta dissertação empregamos a teoria de dinâmica simbólica como ferramenta matemática para abordar o problema da representação de seqüências de símbolos que podem ser modeladas por sistemas dinâmicos simbólicos de memória finita. Utilizando teoria de autômatos, apresentamos novos algoritmos para gerar grafos determinísticos com um número mínimo de vértices que apresentam a linguagem de um sistema dinâmico simbólico de memória finita. Para isto, definimos um novo método, empregando fundamentos da teoria algébrica de linguagem, para determinar as classes da relação de equivalência ρ de *Myhill-Nerode* sobre a linguagem do sistema dinâmico simbólico de memória finita. A linguagem de um sistema dinâmico simbólico de memória finita é regular e, portando, o conjunto das classes de equivalência de ρ é finito. Estas classes são interpretadas como os vértices do grafo determinístico com um número mínimo de vértices que apresenta a linguagem.

O método apresentado é estendido para sistemas dinâmicos simbólicos de memória finita periódicos, que formam a classe (na teoria de dinâmica simbólica) utilizada para modelar conjuntos de seqüências com restrição empregadas tanto para correção de erros quanto para codificação de linha.

Abstract of Dissertation presented to UFPE as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering

REPRESENTATIONS OF SHIFTS OF FINITE TYPE USING GRAPHS

Daniel Pedro Bezerra Chaves

September/2006

Supervisor: Prof. Cecilio José Lins Pimentel, Ph.D.
Area of Concentration: Communication
Keywords: symbolic dynamics, directed graphs, automata, line codes, constrained systems
Number of pages: 74

Symbolic dynamics is commonly applied as a mathematical tool for studying the presentation of sequences that form a shift of finite type. A new algorithm is proposed to generate the minimal synchronizing and deterministic presentation of a shift of finite type, based on algebraic language theory. The main idea is to determine the elements in a subset of the language of a shift of finite type that belong to the same equivalence classes of the *Myhill-Nerode* relation ρ. Since the language of a shift of finite type is regular, then ρ has a finite number of equivalence classes. These classes are the vertices of the proposed graph construction.

The proposed method is extended for periodic shifts of finite type, which is the symbolic dynamical system applied to study sets of sequences that have both error control and line coding properties.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

CAPÍTULO 1

INTRODUÇÃO

Em um sistema de comunicações, a informação é transmitida entre pontos do espaço através de um canal de comunicações. Exemplos de áreas de aplicação da engenharia de telecomunicações nas quais esta visão de canal é realçada são os sistemas de comunicação espacial e sistemas telefônicos fixos e móveis. No entanto, sistemas de armazenamento de informação, como DVD e *pen-drive*, também são interpretados como canais. A principal diferença entre canais de transmissão de informação e canais de armazenamento de informação é que o conceito de espaço é substituído pelo de tempo. Em sistemas de armazenamento, a informação é gravada em um ponto do tempo e acessada em um ponto posterior.

Aplicações atuais de sistemas de armazenamento requerem boa imunidade contra erro [1]. Uma das dificuldades para obtenção deste objetivo é a crescente demanda por maiores capacidades de armazenamento de informação, que propicia o desenvolvimento de dispositivos hábeis a armazenar mais dados por unidade de área, tornando a ação de leitura e escrita dos dados menos confiável. Esta complicação deve-se aos efeitos de interferência inter-simbólica, imprecisão do relógio e ruído.

Codificadores restritivos, também conhecidos como *codificadores moduladores* ou *codificadores de linha*, convertem seqüências arbitrárias em seqüências satisfazendo uma dada restrição. De forma geral, o objetivo de um código restritivo é melhorar o desempenho do sistema de comunicação adaptando a seqüência de dados às características do canal.

Além dos códigos restritivos, também são empregados os códigos corretores de erros (CCE). Um dos principais critérios de projeto de um CCE consiste na maximização da distância mínima do código, permitindo que a palavra-código transmitida possa ser recuperada a partir de sua versão corrompida pelo ruído do canal. Assim, como diferença entre CCE e códigos restritivos, podemos

dizer que enquanto no primeiro procura-se aumentar a distância mínima do código (característica do conjunto de palavras-código), no segundo a suscetibilidade das palavras-código de serem corrompidas ao passarem pelo canal deve ser minimizada (característica de cada palavra-código). Esta diferença é relativa, já que o conjunto de palavras código de um CCE não pode ser completamente arbitrário e também apresenta restrições. Isto aumenta o interesse em códigos restritivos que também possuam propriedades de correção de erros, o que vem ampliando a região de interseção entre estas duas classes, sem no entanto eliminar as peculiaridades concernentes a cada tipo de código, no que se refere aos fundamentos e problemas fundamentais de cada um deles.

Figura 1.1: Diagrama em blocos de um sistema de comunicação incluindo os blocos dos códigos CCE e restritivo.

A Figura 1.1 mostra o diagrama em blocos da parte de um sistema de comunicações relevante para este trabalho. A mensagem m passa pelo codificador CCE e em seguida pelo codificador restritivo. Na recepção, a mensagem recebida é processada por dois blocos decodificadores, gerando a mensagem decodificada m'. Se a ordem de codificação fosse invertida, o codificador CCE poderia destruir a restrição gerada pelo codificador restritivo. Mas o código restritivo pode alterar as características para as quais o CCE foi projetado, reduzindo sua eficácia. Tentando contornar este dilema, há técnicas que propõem um diagrama em blocos com um codificador CCE sistemático posposto ao codificador restritivo, associado a um procedimento para inserção dos dígitos de paridade na seqüência dos dígitos de informação sem que a restrição seja violada, assunto que será tratado no Capítulo 4. Atualmente, alguns trabalhos apresentam técnicas de codificação que permitem tanto a correção de erros como a geração de seqüências restritas de forma integrada [2], [3], [4].

1.1 Algumas Aplicações

Nesta seção são apresentadas algumas seqüências restritas empregadas em dispositivos comerciais de forma isolada ou combinadas.

Restrição RLL

Seqüências que satisfazem a restrição (d, k)-RLL ((d, k)-*runlength-limited*), com d e k inteiros tais que $0 \leq d \leq k$, não possuem seqüências de zeros consecutivos de comprimento maior que k (restrição-k) e seqüências de zeros consecutivos entre uns possuem comprimento pelo menos d (restrição-d). Nas aplicações, a restrição-d reduz o efeito da interferência inter-simbólica e a restrição-k melhora o controle da sincronização de relógio. Entre as seqüências (d, k)-RLL comerciais estão a $(1, 3)$-RLL, empregada em drivers para discos magnéticos flexíveis, e a $(2, 10)$-RLL, empregada em drivers para CD e DVD [5].

Todas as seqüências satisfazendo uma restrição (d, k)-RLL podem ser descritas pela leitura dos rótulos de caminhos em um grafo direcionado, rotulado e finito como apresentado na Figura 1.2.

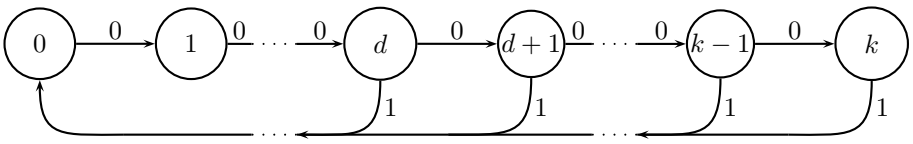


Figura 1.2: Grafo apresentando a restrição (d, k)-RLL.

Como descrição alternativa, poderíamos especificar um conjunto de seqüências proibidas $\mathcal{F}$, tal que, para qualquer seqüência $a_1 a_2 \ldots a_n$ se $a_i \ldots a_j \notin \mathcal{F}$, $1 \leq i \leq j \leq n$, então $a_1 a_2 \ldots a_n$ satisfaz a restrição (d, k)-RLL. Para $d > 0$ e $k < \infty$ o conjunto proibido é:

$$\mathcal{F} = \{11, 101, \ldots, 1\underbrace{0 \ldots 0}_{(d-1)}1, \underbrace{00 \ldots 0}_{(k+1)}\}$$

Restrição anti-tom

Há aplicações que impõem um limite máximo para o comprimento de seqüências periódicas. Da analogia com um sinal periódico decorre o nome dado a esta restrição.

Como exemplo, enquanto na restrição RLL o parâmetro k limita o comprimento máximo da seqüência de período um $000\ldots$, poderia ser de justificativa prática limitar o comprimento máximo de seqüências de período dois, como $101010\ldots$ ou $010101\ldots$.

A restrição de seqüências com unidade periódica 10 e 01 têm sido empregada em algumas versões de canais sem fio por infravermelho para troca de dados entre dispositivos móveis, como computadores portáteis. A seqüência de bits é convertida em uma seqüência de pulsos, na qual o bit "1" é

representado pela presença de pulso e o bit "0" pela ausência de pulso. Quanto maior o ciclo de trabalho de uma seqüência (número de ocorrências de uns dividido pelo comprimento da seqüência) maior o número de pulsos transmitidos por unidade de tempo. Assim, para limitar a potência transmitida, seqüências com alto ciclo de trabalho são restringidas. Na Figura 1.3 apresentamos um grafo que satisfaz simultaneamente a restrição $(1, 3)$-RLL e limita a dois o número máximo de ocorrências de "10" ou "01", sendo o conjunto proibido $\mathcal{F} = \{11, 0000, 10101\}$.

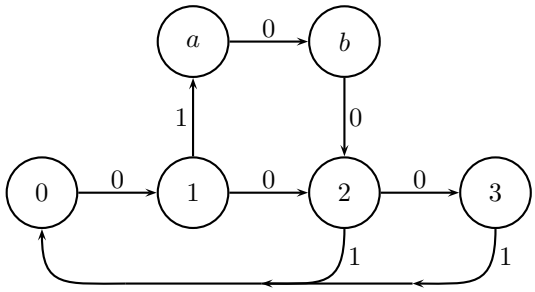


Figura 1.3: Restrição anti-tom: satisfaz a restrição $(1, 3)$-RLL e limita o número máximo de repetições de "10" ou "01".

A restrição (d, k)-RLL é imposta para reduzir a interferência inter-simbólica e melhorar o sincronismo. Se esta restrição for $(1, k)$-RLL, então ciclos de trabalho máximo (aproximadamente 50%) só são alcançados pelas seqüências $0101\ldots$ e $1010\ldots$. Assim, limitar tais seqüências pode ser útil em algumas aplicações, *e.g.*, o padrão *Very Fast Infrared* adotado pela *Infrared Data Association* (IrDA-VFIr) que inclui como parte do formato uma restrição que simultaneamente satisfaz a $(1, 13)$-RLL e limita o número máximo de repetições de "10" ou "01" a cinco [6]. A restrição anti-tom também tem sido aplicada para melhorar a temporização e algoritmos de controle de ganho em sistemas de gravação magnética [4].

Restrição para sincronização

Há aplicações onde a restrição é definida por um conjunto de palavras proibidas sobre um alfabeto, comum em sistemas que empregam marcadores. Tais restrições podem ser úteis para sincronização, procedimento comum em sistemas de armazenamento, onde os dados são agrupados em setores. Para acessar os dados em um setor, a cabeça de leitura deve ser posicionada no início deste. No entanto, tanto o posicionamento preciso no primeiro bit de um setor, como a possível perda de sincronismo na execução do posicionamento, são dificuldades técnicas a serem contornadas. Para redução destas dificuldades, os setores são antecedidos por seqüências reservadas (não ocorrem dentro dos setores), que auxiliam na rápida sincronização do relógio, permitindo a determinação do

início dos dados codificados. A este conjunto de palavras reservadas dar-se-á o nome de marca de sincronização ou *sync mark* (ver Figura 1.4).

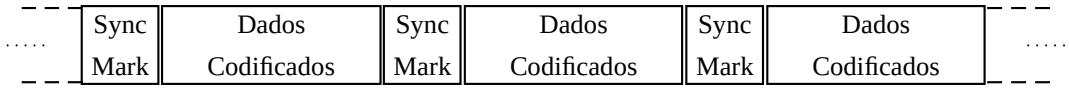


Figura 1.4: Organização da seqüência de dígitos gravada em um dispositivo de armazenamento de dados.

O posicionamento das marcas de sincronização é determinado pela correlação das seqüências de entrada com as marcas de sincronização. Caso uma marca de sincronização pudesse ocorrer na seqüência de dados, a leitura de uma falsa marca poderia levar a uma decodificação errada. Assim, a codificação dos dados deve gerar seqüências em que não ocorram marcas de sincronização, como também as seqüências mais prováveis geradas pela ação do ruído sobre a marca de sincronização.

Como exemplo, consideremos o conjunto de quatro palavras de comprimento 24 cada: $010w_0$ $01w_1$, $010w_0 10w_1$, $001w_0 01w_1$ e $001w_0 10w_1$, onde $w_0 = 000000$ e $w_1 = 0101010101010$. Utilizadas em um sistema magnético [4], a primeira destas seqüências foi utilizada como marca de sincronização e as outras são as seqüências mais prováveis de ocorrer pela ação do ruído sobre a primeira. Nestes sistemas as restrições $(1, 7)$-RLL e anti-tom também foram empregadas.

Restrição RLL multi-espaçada

Esta restrição constitui uma variante da (d, k)-RLL, caracterizada pelos parâmetros (d, k, s), onde d e k determinam o número mínimo e máximo de zeros entre uns consecutivos, respectivamente. O parâmetro s indica que o número de zeros entre uns consecutivos deve ser um múltiplo de s, *e.g.*, se $s = 2$ toda seqüência de zeros tem comprimento par. Na Figura 1.5 temos o grafo que apresenta a restrição $(2, 18, 2)$-RLL.

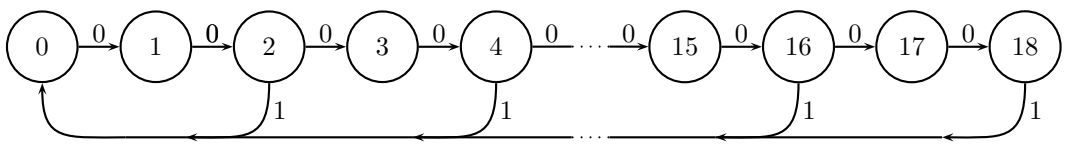


Figura 1.5: Grafo apresentando a restrição $(2, 18, 2)$-RLL.

As restrições $(d, k, 2)$-RLL foram inicialmente investigadas no contexto de gravação magnética

[7]. Recentemente foi mostrado que as restrições $(d, k, 2)$-RLL desempenham um papel natural em sistemas de gravação óptico-magnético [4], tendo a IBM desenvolvido um protótipo deste sistema que emprega a restrição $(2, 18, 2)$-RLL.

Restrição espectro limitado

Algumas restrições são definidas sobre um alfabeto bipolar $\{+1, -1\}$. Este alfabeto é diretamente associado a polaridade do sinal gravado ou transmitido, o que é enfatizado pela denominação *seqüência de polaridades*, empregada para enfatizar essa associação.

Um sistema restrito de seqüências bipolares tem *espectro limitado* em uma dada freqüência (normalizada) $f = m/n$, se existe uma constante B, tal que, para toda seqüência $w = w_0 w_1 \ldots w_{\ell-1}$ que satisfaz a restrição do sistema e qualquer $0 \leq i \leq i' < \ell$, tem-se

$$\left| \sum_{s=i}^{i'} w_s e^{-\jmath\, 2\pi s \frac{m}{n}} \right| \leq B, \tag{1.1}$$

onde $\jmath = \sqrt{-1}$ [8] (a freqüência real é dada por f/T, sendo T o período de um bit).

Seqüências com espectro limitado em $f = 0$, chamadas *dc-free* ou de conteúdo limitado, vêm sendo empregadas em sistemas de gravação magnética. Restrições relacionadas são impostas a sistemas de gravação óptica para reduzir interações entre os dados gravados e o sistema de gravação, como também para filtragem de ruídos de baixa freqüência resultantes de marcas digitais [4]. A restrição *dc-free* também é usada em sistemas de comunicação onde sinais de baixa freqüência são susceptíveis à distorção, o que inclui linhas de transmissão via cabo e fibra óptica.

Para $f = 0$, o valor máximo do módulo do somatório da Equação 1.1 é chamado de DSV (do inglês *digital sum variation*) da seqüência. O conjunto de todas as seqüências com DSV limitado a B é um sistema restrito apresentado pelo grafo da Figura 1.6. Para uma restrição *dc-free*, quanto maior o valor de B menor será a redução causada nas componentes espectrais próximas a $f = 0$. Esta restrição é um exemplo em que o conjunto proibido não é finito.

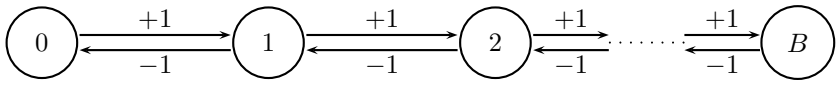


Figura 1.6: Grafo da restrição espectro limitado *dc-free*.

1.2 Dinâmica Simbólica e Teoria de Autômatos: Aspectos Gerais

Nesta seção enfocamos a motivação que levou ao desenvolvimento da teoria de dinâmica simbólica e de autômatos e o contexto da aplicação destas à teoria de códigos.

Dinâmica Simbólica

Dinâmica Simbólica faz parte da teoria de sistemas dinâmicos. Este campo de estudos tem como origem os trabalhos em topologia de Marston Morse no início do século vinte [9], que o descreveu como “estudo algébrico e geométrico da recorrência e transitividade". A idéia é dividir a superfície associada a um sistema dinâmico em um número finito de regiões e associa-las a símbolos. A seqüência de símbolos gerados pela passagem de uma trajetória do sistema dinâmico (associada a um ponto inicial) pelas regiões correspondentes, gera um sistema dinâmico “simbólico", que reflete comportamentos quantitativos e qualitativos do sistema original, como por exemplo o tempo médio em que o sistema permanece em uma região, ou se uma região torna-se periódica.

A teoria de dinâmica simbólica teve seu escopo de aplicações expandido ao ser usada como ferramenta matemática nas teorias de código e informação, a partir do emprego de ferramentas para análise de propriedades entre conjuntos de seqüências infinitas e dos mapeamentos entre estes. Como por exemplo, a teoria de dinâmica simbólica tem sido usada no estudo de codificadores de linha e decodificadores de janela deslizante [10], [11].

Autômatos

Com as tentativas de esclarecer questões sobre a natureza das demonstrações matemáticas, foi estabelecida a definição de algoritmo. Como meio de avaliar os algoritmos, dois métodos de classificação foram desenvolvidos. O primeiro os classifica pelo tempo de execução, sendo conhecido como teoria da complexidade. O segundo os classifica de acordo com a quantidade de memória necessária para implementação, sendo conhecido como teoria da linguagem. De acordo com esta última, os algoritmos mais simples são aqueles que podem ser implementados por autômatos finitos, sendo estes objetos de nosso interesse.

A teoria de autômatos, tema largamente explorado na teoria de linguagens formais, é utilizada no estudo de seqüências de símbolos gerados por uma máquina de estados finitos. Devido ao repertório de algoritmos originados na teoria de autômatos e com aplicação em teoria de grafos, avanços na teoria de dinâmica simbólica têm sido obtidos pela aplicação de resultados da teoria de autômatos. Como exemplo, a teoria de autômatos tem sido usada na simplificação de conjuntos finitos de se-

qüências e de grafos usados para representar sistemas simbólicos.

1.3 Objetivos da Dissertação

Como objetivos deste trabalho, inicialmente, construiremos algoritmos de complexidade linear que gerem o grafo determinístico com o menor número de vértices (dadas certas restrições) e que representem a classe dos sistemas dinâmicos simbólicos de memória finita ou SFT (do inglês *shift of finite type*). Posteriormente, estenderemos o método apresentado para sistemas dinâmicos simbólicos de memória finita periódicos ou PFT (do inglês *periodic shift of finite type*). Para este caso, iremos nos concentrar nos sistemas dinâmicos simbólicos PFT irredutíveis, por serem estes de maior interesse prático.

1.4 Organização da Dissertação

Esta dissertação está organizada em quatro capítulos. O Capítulo 2 introduz conceitos e definições da teoria de autômatos e dinâmica simbólica. O Capítulo 3 apresenta um novo algoritmo para gerar o grafo determinístico com o menor número de vértices que apresenta um sistema dinâmico simbólico de memória finita. No Capítulo 4 estendemos o algoritmo do Capítulo 3 para o caso de sistemas dinâmicos simbólicos de memória finita periódicos.

CAPÍTULO 2

AUTÔMATO, DINÂMICA SIMBÓLICA E GRAFO: CONCEITOS E RELAÇÕES

As relações entre a teoria de autômatos, multi-grafos direcionados e dinâmica simbólica têm sido usadas por vários autores para obtenção de novos resultados e novas aplicações para estas teorias [12],[13], [14]. Neste trabalho também usamos estas relações para obtenção de novos resultados.

Neste capítulo, apresentaremos os conceitos básicos de cada teoria, definiremos a nomenclatura usada nesta dissertação e destacaremos algumas relações conhecidas entre estas teorias. Para uma abordagem mais detalhada sobre dinâmica simbólica e teoria de grafo, o livro de Douglas Lind e Brian Marcus [10] é o material mais indicado. Para uma abordagem introdutória nestas áreas com extensa revisão bibliográfica e que explora as correlações com a teoria de autômatos, indicamos o tutorial de Marie-Pierre Béal e Dominique Perrin [12]. Para a teoria de autômatos sugerimos [15], [16] e [17], com ênfase para o último.

2.1 Teoria de Autômatos: alfabeto, palavras e linguagem

Informação é usualmente representada como uma seqüência de símbolos obtidos de um conjunto arbitrário. Chamaremos de *alfabeto* qualquer conjunto de símbolos usado com esta conotação, representando-o por $\mathcal{A}$. Os elementos de $\mathcal{A}$ são chamados de *letras* ou *símbolos*. Para $n \geq 0$ um inteiro, a lista de símbolos $(a_1, a_2, \ldots, a_n)$, $a_i \in \mathcal{A}$, é chamada de *palavra* ou *seqüência* de comprimento n, sendo representada por $a_1 a_2 \ldots a_n$. A palavra de comprimento zero $()$, $n = 0$, é chamada de palavra nula e representada por ε. Se para uma seqüência $b_1 \ldots b_m$ de símbolos em $\mathcal{A}$ verifica-se que $b_1 \ldots b_m = a_i \ldots a_j$, $1 \leq i \leq j \leq n$, então $b_1 \ldots b_m$ é uma *sub-palavra* de $a_1 \ldots a_n$. Repre-

sentamos por $|w|$ o comprimento de uma palavra w em $\mathcal{A}$. Assim, para $\{a, b, c\} \in \mathcal{A}$ e $w = aabcb$ temos que $|w| = 5$. Particularmente, $|\varepsilon| = 0$.

Dado duas palavras u e v com símbolos em um alfabeto $\mathcal{A}$, podemos formar uma nova palavra $u \cdot v$, chamada a *concatenação* de u e v, pela justaposição dos símbolos de u com os símbolos de v. Exemplificando, se $u = a_1a_2 \dots a_n$ e $v = b_1b_2 \dots b_m$, então $u \cdot v = a_1a_2 \dots a_nb_1b_2 \dots b_m$, ou seja, a concatenação gerou uma nova palavra com símbolos em $\mathcal{A}$ onde os n primeiros símbolos coincidem com u e os m últimos símbolos coincidem com v. Usualmente, denotaremos a concatenação de u e v por uv, em substituição a $u \cdot v$. Decorre da operação de concatenar palavras que $|uv| = |u| + |v|$, assim $|u\varepsilon| = |u|+|\varepsilon| = |u|$. Concluímos que ε é o elemento identidade da operação de concatenação, ou seja, $u\varepsilon = \varepsilon u = u$.

Se $\mathcal{A}$ e $\mathcal{B}$ são conjuntos de palavras com símbolos em $\mathcal{A}$, denota-se pela concatenação de $\mathcal{A}$ e $\mathcal{B}$ o conjunto $\mathcal{AB} = \{uv | u \in \mathcal{A} \text{ e } v \in \mathcal{B}\}$. Decorre da operação de concatenação entre conjuntos que $\mathcal{A}^+ = \bigcup_{i=1}^{\infty} \mathcal{A}^i$, onde $\mathcal{A}^1 = \mathcal{A}$ e para $n \geq 2$, $\mathcal{A}^n = \mathcal{A}^{n-1}\mathcal{A}$. O conjunto $\mathcal{A}^* = \varepsilon \cup \mathcal{A}^+$ recebe a denominação especial de *fechamento de Kleene*, denominação justificada na teoria algébrica da linguagem, sendo o sobrescrito " $*$ " denominado *estrela de Kleene*. Qualquer subconjunto de $\mathcal{A}^*$ é denominado uma *linguagem formal* (por simplicidade, iremos chamá-lo de linguagem). Assim, $\mathcal{A} \subseteq \mathcal{A}^*$ é uma linguagem. O complemento de $\mathcal{B}$ com relação a $\mathcal{A}$ é definido como $\mathcal{B}' = \{v |\ v \in \mathcal{A} \text{ e } v \notin \mathcal{B}\}$. O conjunto de prefixos de $\mathcal{A}$, representado por $\mathcal{P}(\mathcal{A})$, é definido como $\mathcal{P}(\mathcal{A}) = \{v |\ \exists\, u \in \mathcal{A}^*,\ vu \in \mathcal{A}\}$, e o conjunto de sufixos de $\mathcal{A}$, representado por $\mathcal{S}(\mathcal{A})$, é definido como $\mathcal{S}(\mathcal{A}) = \{v |\ \exists\, u \in \mathcal{A}^*,\ uv \in \mathcal{A}\}$. Para $\mathcal{A} = \{110\}$, segue o cálculo de $\mathcal{P}(\mathcal{A})$ e $\mathcal{S}(\mathcal{A})$.

$$\mathcal{P}(\{110\}) = \{\varepsilon, 1, 11, 110\} \quad \text{e} \quad \mathcal{S}(\{110\}) = \{\varepsilon, 0, 10, 110\}$$

Definimos os conjuntos $\mathcal{AB}^{-1} = \{v |\ \exists\, u \in \mathcal{B},\ vu \in \mathcal{A}\}$ e $\mathcal{B}^{-1}\mathcal{A} = \{v |\ \exists\, u \in \mathcal{B},\ uv \in \mathcal{A}\}$, como exemplos:

$$\{111, 110, 000\}\{1\}^{-1} = \{11\} \quad \text{e} \quad \{1\}^{-1}\{111, 110, 000\} = \{11, 10\}$$

2.2 Dinâmica Simbólica

Apresentamos nesta seção uma introdução aos conceitos de Dinâmica Simbólica [10]. Primeiramente, denotamos por $\mathcal{A}^{\mathbb{Z}}$ o conjunto de todas as seqüências bi-infinitas de símbolos em um alfabeto

$\mathcal{A}$ e chamamos estas seqüências de *pontos* em $\mathcal{A}^{\mathbb{Z}}$. Assim, um ponto em $\mathcal{A}^{\mathbb{Z}}$ é uma seqüência

$$\begin{aligned} x &= (x_i)_{i\in\mathbb{Z}} \\ &= \ldots x_{-1}x_0x_1\ldots \end{aligned}$$

onde $x_i \in \mathcal{A}$ é a i-ésima *coordenada* de x. Definimos a *função deslocamento* como $\sigma : \mathcal{A}^{\mathbb{Z}} \to \mathcal{A}^{\mathbb{Z}}$, tal que, $(\sigma(x))_i = x_{i+1}$ para qualquer $i \in \mathbb{Z}$. A função σ é inversível e $\{\sigma^n |\ n \in \mathbb{Z}\}$ constitui uma ação do grupo infinito $(\mathbb{Z}, +)$ sobre o conjunto $\mathcal{A}^{\mathbb{Z}}$.

O conjunto $\mathcal{A}^{\mathbb{Z}}$ pode ser interpretado como um espaço topológico. Considerando a métrica discreta associada ao conjunto $\mathcal{A}$,

$$\rho(\alpha,\beta) = \begin{cases} 0 & \text{se } \alpha = \beta, \\ 1 & \text{se } \alpha \neq \beta. \end{cases}$$

define-se a métrica do espaço discreto associado a i-ésima coordenada do espaço de dimensão infinita $\mathcal{A}^{\mathbb{Z}}$:

$$\rho_i(\alpha,\beta) = \frac{\rho(\alpha,\beta)}{2^{|i|-1}}.$$

A métrica do espaço produto $\mathcal{A}^{\mathbb{Z}}$ formado pelos espaços discretos associados as suas coordenadas é dada por:

$$d(x,y) = \max_{-\infty<i<\infty}\{\rho_i(x_i,y_i)\}.$$

Portanto, a métrica associada pode ser definida por $d(x,y) = 2^{-e(x,y)}$, onde $e(x,y) = \max\{n \geq 0 |\ x_i = y_i,\ -n \leq i \leq n\}$ e por convenção, $e(x,y) = \infty$ se $x = y$ e $e(x,y) = -1$ se $x_0 \neq y_0$. Lembramos que para qualquer $x \in \mathcal{A}^{\mathbb{Z}}$ e $r > 0$, a *bola aberta* de raio r em torno de x é o conjunto $B_r(x) = \{y \in \mathcal{A}^{\mathbb{Z}} |\ d(x,y) < r\}$. Um subconjunto $X \subseteq \mathcal{A}^{\mathbb{Z}}$ é *aberto*, se para todo $x \in X$ há um $r > 0$, tal que $B_r(x) \subseteq X$. Um subconjunto $Y \subseteq \mathcal{A}^{\mathbb{Z}}$ é fechado se seu complemento $\mathcal{A}^{\mathbb{Z}}\backslash Y$ é aberto.

Um *sistema dinâmico simbólico fechado* ou simplesmente *sistema simbólico fechado*, ou ainda *espaço invariante por deslocamento* (do inglês *shift space*) é um subconjunto X de $\mathcal{A}^{\mathbb{Z}}$, tal que:

- X é fechado;
- X é invariante por deslocamento, i.e. $\sigma(X) = X$.

Equivalentemente, um sistema simbólico fechado X tem como propriedade característica que para qualquer $x \in \mathcal{A}^{\mathbb{Z}}$, se toda sub-palavra de x pertence a algum ponto de X, então x pertence a X. Assim, um sistema simbólico fechado é completamente caracterizado pela coleção de palavras

presentes em seus pontos, podendo ser representado por X ou (X, σ) para enfatizar o papel da função deslocamento σ.

Lembramos que um espaço métrico M é *compacto* se, e somente se, todo seqüência em M possui uma subseqüência convergente [18, Teorema 43.5]. Como $\mathcal{A}^{\mathbb{Z}}$ é compacto e X é fechado [10, Exemplo 6.1.17], X também é compacto [18, Teorema 43.8] (entre outras propriedades herdadas de $\mathcal{A}^{\mathbb{Z}}$), o que aumenta o interesse nestes como objeto de estudo em dinâmica simbólica.

Exemplo 2.1. Seja $\mathcal{A} = \{0, 1\}$ e $S = \{x \in \mathcal{A}^{\mathbb{Z}} |\ \exists\, n \in \mathbb{Z},\ x_i = 0 \text{ para } i \leq n \text{ e } x_i = 1 \text{ para } i > n\}$ um sistema simbólico, pela definição de S, este é invariante por deslocamento, contudo, não é fechado. Para verificarmos esta afirmação, observemos que a seqüência formada só por zeros $y = \ldots 000 \ldots$ pertence a $\mathcal{A}^{\mathbb{Z}} \backslash S$, no entanto, para qualquer $0 < r \leq 1$ ou $r = 2$ (contradomínio da função $d(x, y)$, onde $x, y \in \mathcal{A}^{\mathbb{Z}}$) existe uma seqüência bi-infinita $x = (x_i)_{i \in \mathbb{Z}}$, $x_i = 0$ para $i \leq \lceil -\log_2 r + 1 \rceil$ e $x_i = 1$ para $i > \lceil -\log_2 r + 1 \rceil$, tal que, $x \in B_r(y)$ e $x \notin \mathcal{A}^{\mathbb{Z}} \backslash S$, ou seja, $B_r(y) \nsubseteq \mathcal{A}^{\mathbb{Z}} \backslash S$. Portanto, o complemento de S com relação ao conjunto $\mathcal{A}^{\mathbb{Z}}$ não é aberto, logo, S não é um sistema simbólico fechado. A existência de um seqüência em $\mathcal{A}^{\mathbb{Z}}$ que não pertence a S, mas com todos os seus fatores sendo fatores de seqüências de S, implica que S não é fechado.

No contexto de dinâmica simbólica, uma palavra sobre um alfabeto $\mathcal{A}$ também é chamada de *bloco*, sendo ε chamado de bloco vazio. Utilizamos $\mathcal{A}^k$ para nos referirmos ao conjunto de todos os blocos de comprimento k em $\mathcal{A}$. Se $w \in \mathcal{A}^k$, w é chamado de um k-bloco. Um *fator* ou *sub-bloco* de um bloco $u = a_1 a_2 \ldots a_k$ é um bloco $a_i a_{i+1} \ldots a_j$, onde $1 \leq i \leq j \leq k$. Por convenção, ε é um fator de todo bloco. Para todo $x \in \mathcal{A}^{\mathbb{Z}}$ e $i \leq j$, denota-se o bloco em x, da coordenada i a j por $x_{[i,j]} = x_i x_{i+1} \ldots x_j$. Se $i > j$, define-se $x_{[i,j]} = \varepsilon$. Como simples extensão, define-se $x_{[i,j)} = x_i x_{i+1} \ldots x_{j-1}$ e assim, $x_{[i,\infty)} = x_i x_{i+1} \ldots$ é denominada seqüência infinita à direita e $x_{(\infty,i]} = \ldots x_{i-1} x_i$ seqüência infinita à esquerda, ressaltando que as duas últimas não são blocos, já que não possuem comprimento finito.

Como conseqüência das propriedades de um sistema simbólico fechado $X \subseteq \mathcal{A}^{\mathbb{Z}}$, podemos determiná-lo a partir de um subconjunto de $\mathcal{A}^*$. Inicialmente, vamos definir o *conjunto cilindro*, $C_k^X(u)$, como sendo o conjunto dos pontos $x \in X$ para os quais u é um fator iniciando na coordenada k, i.e., $C_k^X(u) = \{x \in X \mid u = x_{[k,k+|u|-1]}\}$. Como X é um conjunto fechado, então podemos afirmar que $\mathcal{A}^{\mathbb{Z}} \backslash X$ é aberto. Assim, para todo $y \in \mathcal{A}^{\mathbb{Z}} \backslash X$ existe $k = k(y)$, tal que para $u_y = y_{[-k,k]}$, então $C_{-k}^{\mathcal{A}^{\mathbb{Z}}}(u_y) \subseteq \mathcal{A}^{\mathbb{Z}} \backslash X$. O conjunto $\mathcal{F} = \{u_y |\ y \in \mathcal{A}^{\mathbb{Z}} \backslash X\}$ é suficiente para definir X. Para verificarmos esta afirmação, suponha que para $u_y \in \mathcal{F}$ existe $x \in X$ tal que $u_y = x_{[i,j]}$, então $\sigma^{(i+k)}(x) \in C_{-k}^{\mathcal{A}^{\mathbb{Z}}}(u_y)$, o que é uma contradição já que X é invariante por deslocamento.

Dado um subconjunto $\mathcal{F} \subseteq \mathcal{A}^*$, se X é o conjunto de pontos em $\mathcal{A}^{\mathbb{Z}}$ que não possuem fatores em $\mathcal{F}$, então $\mathcal{A}^{\mathbb{Z}} \backslash X$ é aberto, consequentemente X é fechado. Para qualquer $x \in X$ temos que $\sigma(x) \in X$ (se x não possui fatores em $\mathcal{F}$, $\sigma(x)$ também não possui). Concluímos que X é um subconjunto de $\mathcal{A}^{\mathbb{Z}}$ fechado e invariante por deslocamento, ou seja, um sistema simbólico fechado. Isto nos permite definir um sistema simbólico fechado, X, a partir de um conjunto $\mathcal{F} \subseteq \mathcal{A}^*$ como: X é o conjunto de pontos em $\mathcal{A}^{\mathbb{Z}}$ que não possuem blocos em $\mathcal{F}$. A relação entre X e $\mathcal{F}$ é representada pela equação $X = \mathsf{X}_{\mathcal{F}}$, onde a notação X diz respeito à operação de formar o conjunto X e este é o conjunto resultante da operação. Da propriedade associada ao conjunto $\mathcal{F}$, este é chamado de *conjunto proibido*.

Exemplo 2.2. Seja $\mathcal{A} = \{e, f, g\}$ e X o conjunto de seqüências de $\mathcal{A}^{\mathbb{Z}}$ para as quais o e só pode ser seguido por e ou f, f por g, g por e ou f. Assim, $\mathcal{F} = \{eg, ff, fe, gg\}$.

Exemplo 2.3. Seja $X \subset \mathcal{A}^{\mathbb{Z}}$, $\mathcal{A} = \{a, b, c\}$, tal que, as palavras da forma $ab^m c^k a$ ocorrem em seqüências de X se, e somente se, $m = k$. X é conhecido como o sistema simbólico fechado livre de contexto e possui conjunto proibido $\mathcal{F} = \{\, ab^m c^k a \,|\, m \neq k,\, m, k \geq 0\}$.

A definição do conjunto $\mathcal{F}$ realça a observação feita acima sobre a determinação de um sistema simbólico fechado X pelo conjunto de todos os fatores em pontos de X. Em alguns casos é mais fácil especificar X por este conjunto que por $\mathcal{F}$, o que justifica a próxima definição. Seja X um sistema simbólico fechado e $\mathcal{B}_n(X)$ o conjunto de todos os n-blocos que ocorrem em pontos de X. A *linguagem* de X é a coleção

$$\mathcal{B}(X) = \bigcup_{n=0}^{\infty} \mathcal{B}_n(X).$$

Apesar de todo conjunto proibido $\mathcal{F} \subseteq \mathcal{A}^*$ determinar um sistema simbólico fechado em $\mathcal{A}$, nem toda linguagem em $\mathcal{A}^*$ determina um sistema simbólico fechado. Um subconjunto em $\mathcal{A}^*$ deve satisfazer as propriedades listadas na Propriedade 2.1 para ser a linguagem de um sistema simbólico fechado [10, Proposição 1.3.4].

Propriedade 2.1.

(a) Seja X um sistema simbólico fechado e $\mathcal{B}(X)$ sua linguagem. Se $w \in \mathcal{B}(X)$, então:

- ▷ Todo sub-bloco de w pertence a $\mathcal{B}(X)$, portanto $\mathcal{B}(X)$ é uma *linguagem fatorial*;
- ▷ Há blocos não vazios u e v em $\mathcal{B}(X)$ tal que $uwv \in \mathcal{B}(X)$, portanto $\mathcal{B}(X)$ é uma *linguagem prolongável*.

(b) A linguagem de um sistema simbólico fechado é caracterizada por (a), ou seja, se $L \subseteq \mathcal{A}^*$, então L é a linguagem de algum sistema simbólico fechado X se, e somente se, L satisfaz a condição (a).

(c) A linguagem de um sistema simbólico fechado X o caracteriza, pois $X = \mathsf{X}_{\mathcal{B}(X)'}$. Assim, dois sistemas simbólicos fechados são iguais se, e somente se, eles têm a mesma linguagem.

Na prática os sistemas simbólicos fechados de maior interesse são os *irredutíveis*, sendo aqueles que para quaisquer $u, v \in \mathcal{B}(X)$, existe um w, tal que, $uwv \in \mathcal{B}(X)$. Neste caso $\mathcal{B}(X)$ é dita uma *linguagem transitiva* (formalmente, uma linguagem formal $L \subseteq \mathcal{A}^*$ é transitiva se $\forall u, v \in L,\ \exists\, w \in \mathcal{A}^*$ tal que $uwv \in L$). Quando $\mathcal{B}(X)$ não é transitiva dizemos que X é um sistema simbólico fechado *redutível*.

Exemplo 2.4. Seja S um subconjunto dos inteiros não negativos. Se S é finito, define-se $X = \mathsf{X}(S)$ como o conjunto de todas as seqüências binárias bi-infinitas para as quais o 1 ocorre infinitas vezes em cada direção e, tal que, o número de ocorrências do 0 entre uns consecutivos é um inteiro em S. Assim, $x = \ldots 10^{n_{-1}}10^{n_0}10^{n_1}1\ldots$, onde $n_i \in S$, é uma seqüência típica do conjunto $\mathsf{X}(S)$. Caso consideremos o conjunto S infinito, então $0^n \in \mathcal{B}(\mathsf{X}(S))$ para todo $n \geq 0$, no entanto, a seqüência $y = \ldots 000\ldots$, composta só por zeros, não pertence a $\mathsf{X}(S)$. Portanto, $\mathsf{X}(S)$ não é um sistema simbólico fechado se S é infinito. Seja S finito e $m = \max\{\, n \,|\, n \in S\}$, um possível conjunto pribido para $\mathsf{X}(S)$ é $\mathcal{F} = \{10^n1 \mid n \geq 0 \text{ e } n \notin S\} \cup \{0^{m+1}\}$.

Um importante conceito em dinâmica simbólica é o de *sistema dinâmico*, representado por (M, ϕ), onde M é um espaço métrico compacto e $\phi : M \to M$ é uma função contínua. Se ϕ é um *homeomorfismo* (ϕ é contínua, injetiva, sobrejetiva e seu inverso é contínuo), (M, ϕ) é chamado um sistema dinâmico inversível. Um homeomorfismo importante, já apresentado neste texto, é a função σ, sendo (X, σ_X) (o subscrito indica a restrição de σ ao subconjunto X de $\mathcal{A}^{\mathbb{Z}}$) interpretado como o mapeamento subjacente que provê a dinâmica, tal que, se (X, ϕ) é um sistema dinâmico, então $\sigma_X \circ \phi = \phi \circ \sigma_X$, ou seja, o comportamento do sistema dinâmico é invariante no “tempo".

Com o conceito de sistema dinâmico surge a questão: Como dois sistemas dinâmicos estão relacionados? Para abordá-la, partimos da definição de um *homomorfismo* $\theta : (M, \phi) \to (N, \psi)$ entre sistemas dinâmicos como sendo a função contínua $\theta : M \to N$ satisfazendo a propriedade comutativa $\psi \circ \theta = \theta \circ \phi$, representada pelo diagrama

$$\begin{array}{ccc} M & \xrightarrow{\phi} & M \\ {\scriptstyle\theta}\downarrow & & \downarrow{\scriptstyle\theta} \\ N & \xrightarrow[\psi]{} & N \end{array}$$

Se θ é injetiva e sobrejetiva, como M é compacto, então o mapeamento inverso θ^{-1} também é contínuo. Neste caso, θ é chamado de *conjugado topológico* e é escrito como $\theta : (M, \psi) \cong (N, \phi)$. Quando há um conjugado topológico entre dois sistemas dinâmicos, estes são chamados de *topologicamente conjugados.*

Consideremos o importante caso quando ψ e ϕ são funções deslocamento. Dado os sistemas dinâmicos fechados (X, σ_X) e (Y, σ_Y) sobre $\mathcal{A}^{\mathbb{Z}}$ e $\mathcal{B}^{\mathbb{Z}}$, respectivamente, seja $\theta : (X, \sigma_X) \to (Y, \sigma_Y)$ um homomorfismo. Como $\mathcal{A}^{\mathbb{Z}}$ é compacto e X fechado, então θ é uniformemente contínua. Consequentemente, há um inteiro k tal que para todo ponto $x \in X$, $(\theta(x))_0$ é determinado pelo bloco $x_{[-k,k]}$. Como θ comuta com a função deslocamento, então qualquer símbolo de $\theta(x)$ é determinado por um bloco de comprimento $(2k + 1)$.

A importância deste resultado reside na determinação parcial da relação entre homomorfismos e *códigos de bloco deslizante* (SBC, do inglês *sliding block codes*). Um SBC mapeia seqüências $\ldots x_{-1}x_0x_1 \ldots$ de um sistema dinâmico fechado X sobre $\mathcal{A}$ em seqüências $\ldots y_{-1}y_0y_1 \ldots$ sobre um alfabeto $\mathcal{B}$. Uma apresentação mais precisa requer a especificação de um mapeamento de bloco, que é uma função $\Phi : \mathcal{B}_{m+n+1}(X) \to \mathcal{B}$, ou seja, de blocos em $\mathcal{B}_{m+n+1}(X)$ para símbolos de um alfabeto $\mathcal{B}$. Assim, a função $\phi : X \to \mathcal{B}^{\mathbb{Z}}$ definida por $y = \phi(x)$, onde $y_i = \Phi(x_{i-m} \ldots x_{i+n}) = \Phi(x_{[i-m,i+n]})$, é um SBC de *memória* m e *antecipação* n induzido por Φ, o que é representado por $\phi = \Phi_{\infty}^{[-m,n]}$, ou simplesmente $\phi = \Phi_{\infty}$, se a memória e antecipação são conhecidos. Quando $Y \subseteq \mathcal{B}^{\mathbb{Z}}$ é um sistema simbólico fechado e $\phi(X) \subseteq Y$, então escrevemos $\phi : X \to Y$.

Seja $\phi : X \to Y$ um SBC entre os sistemas simbólicos fechados X e Y. Sejam $x, x' \in X$ tal que $e(x, x') \geq m + n + 1$, então $e(\phi(x), \phi(x')) \geq e(x, x') - (m + n)$, logo dada uma seqüência de pontos em X, $\{x_n\}_{n=1}^{\infty}$, se $\lim_{n\to\infty} d(x, x_n) = 0$ (isto é, se $\{x_n\}_{n=1}^{\infty}$ converge para x) então $\lim_{n\to\infty} d(\phi(x), \phi(x_n)) = \lim_{n\to\infty} d(x, x_n) \cdot 2^{m+n} = 0$. Assim, um SBC é uma função contínua. Seja $y = \phi(x)$, observemos que $\phi(\sigma_X(x))_i = \Phi(x_{[i+1-m,i+1+n]}) = y_{i+1} = \sigma_Y(\phi(x))_i$, portanto $\phi \circ \sigma_X = \sigma_Y \circ \phi$, logo um SBC comuta com a função deslocamento. Concluímos que uma função $\theta : (X, \sigma_X) \to (Y, \sigma_Y)$, onde (X, σ_X) e (Y, σ_Y) são sistemas simbólicos fechados, é um SBC se, e somente se, θ é um homomorfismo.

2.3 Autômato: Linguagem Regular

Um autômato finito é definido pela quíntupla $A = (\mathcal{V}, \mathcal{E}, \lambda, \mathcal{I}, \mathcal{T})$, onde: $\mathcal{V}$ é um conjunto finito de *vértices* (ou *estados*); $\mathcal{E}$ é um conjunto finito de *ramos*, ao qual definem-se as funções $i : \mathcal{E} \to \mathcal{V}$ e $t : \mathcal{E} \to \mathcal{V}$, que determinam o estado inicial e o terminal de um ramo, respectivamente; $\lambda : \mathcal{E} \to \mathcal{A}$ é a *função de rotulação*; $\mathcal{I} \subseteq \mathcal{V}$ é o conjunto de *estados iniciais*; por fim, $\mathcal{T} \subseteq \mathcal{V}$ é o conjunto de *estados terminais*. Temos particular interesse em autômatos determinísticos, isto é, para quaisquer $e_1, e_2 \in \mathcal{E}$, se $i(e_1) = i(e_2)$ e $\lambda(e_1) = \lambda(e_2)$ então $e_1 = e_2$. A Figura 2.1 mostra um autômato finito, onde $\mathcal{V} = \{I, J, K\}$, $\mathcal{I} = \{I\}$ (indicado pela seta apontando para o círculo associado ao vértice), $\mathcal{T} = \{K\}$ (indicado pelo círculo interno associado ao vértice).

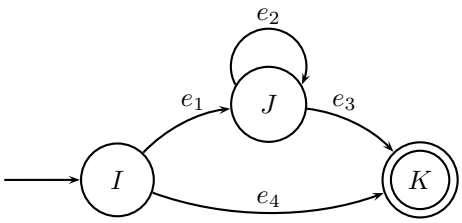


Figura 2.1: Autômato finito.

Quando o autômato é determinístico, também é comum defini-lo pela quíntupla $A = (\mathcal{V}, \mathcal{A}, \delta, \mathcal{I}, \mathcal{T})$, onde $\mathcal{A}$ é o alfabeto e $\delta : \mathcal{B} \to \mathcal{V}$, $\mathcal{B} \subseteq \mathcal{V} \times \mathcal{A}$, é a *função de transição*, que determina os ramos e seus rótulos, *e.g.*, seja $\delta(I, a) = J$ temos I como estado inicial, J como estado final e a como rótulo. Estas representações de um autômato são equivalentes, podendo-se gerar uma representação a partir da outra. Assim, para todo $(I, a) \in \mathcal{B}$ incluímos e em $\mathcal{E}$, tal que $i(e) = I$, $t(e) = \delta(I, a)$ e $\lambda(e) = a$. Ao contrário, para todo $e \in \mathcal{E}$ incluímos $(i(e), \lambda(e))$ em $\mathcal{B}$ e definimos $\delta(I, a) = t(e)$. Um autômato $A' = (\mathcal{V}', \mathcal{E}', \lambda', \mathcal{I}', \mathcal{T}')$ é um sub-autômato de $A = (\mathcal{V}, \mathcal{E}, \lambda, \mathcal{I}, \mathcal{T})$, se $\mathcal{V}' \subseteq \mathcal{V}$, $\mathcal{I}' \subseteq \mathcal{I}$, $\mathcal{T}' \subseteq \mathcal{T}$, todo $e \in \mathcal{E}$ para o qual $i(e), t(e) \in \mathcal{V}'$ pertence a $\mathcal{E}'$ e $\lambda' = \lambda|\mathcal{E}'$.

Uma seqüência $\pi = e_1 \dots e_n \in \mathcal{E}^*$ é um *caminho* em um autômato A se $t(e_i) = i(e_{i+1}),\ 1 \leq i < n$. Seja $P(A) = \{\pi|\ \pi$ é um caminho em $A\}$, podemos estender a função rotulação $\lambda : P(A) \to \mathcal{A}^*$, tal que, se $\pi = e_1 \dots e_n$ então $\lambda(\pi) = \lambda(e_1) \dots \lambda(e_n)$. As funções i e t também podem ser estendidas para $i, t : P(A) \to \mathcal{V}$, tal que $i(\pi) = i(e_1)$ e $t(\pi) = t(e_n)$. Agora dispomos de ferramentas suficientes para introduzirmos o conceito de uma linguagem apresentada por um autômato. Seja $L \subseteq \mathcal{A}^*$, L é apresentada por um autômato A se $L = \{\lambda(\pi)|\ \pi \in P(A), i(\pi) \in \mathcal{I}$ e $t(\pi) \in \mathcal{T}\}$. Denota-se por $L(A)$ a linguagem apresentada por um autômato A.

Uma questão natural neste ponto é: dada uma linguagem L, quando há um autômato finito A tal que $L = L(A)$? A resposta foi dada por Kleene, sendo um dos teoremas fundamentais da teoria da computação. Para expormos a idéia envolvida, considere $\mathcal{C}$ uma coleção de linguagens formais sobre um alfabeto $\mathcal{A}$. A coleção $\mathcal{C}$ é dita fechada sob união ($+$), concatenação ($\cdot$) e fechamento de Kleene ($*$), chamados de *operadores regulares*, se dadas duas linguagens quaisquer L e M em $\mathcal{C}$ então $L + M$, $L \cdot M$, L^* e M^* também estão em $\mathcal{C}$. Seja $\bar{\mathcal{C}}$ o fechamento de $\mathcal{C}$ sob os operadores regulares, conhecido como *fechamento regular* de $\mathcal{C}$ e descrito como: se $\mathcal{C}^{(n)}$ são todas as linguagens obtidas de $\mathcal{C}$ por n aplicações dos operadores regulares, então $\bar{\mathcal{C}} = \bigcup_{i=0}^{\infty} \mathcal{C}^{(n)}$, onde $\mathcal{C}^{(0)} = \mathcal{C}$. Observe que, há uma equação em $\bar{\mathcal{C}}$ com n ocorrências de operadores regulares para qualquer $n \geq 0$, mas que dada uma equação em $\bar{\mathcal{C}}$ há um número finito de operadores regulares. A importância do conjunto $\bar{\mathcal{C}}$ decorre da seguinte propriedade.

Propriedade 2.2.

(1) Linguagens finitas são apresentadas por autômatos [17, Proposição 2.2.4].

(2) Se L é a união de duas linguagens apresentadas por autômatos, então L é apresentada por um autômato [17, Proposição 2.5.6].

(3) Se L é a concatenação de duas linguagens apresentadas por autômatos, então L é apresentada por um autômato [17, Proposição 3.3.4].

(4) Se L é apresentada por um autômato, então L^* também é apresentada por um autômato [17, Proposição 3.3.6].

Dado um alfabeto $\mathcal{A}$, estamos interessados nos casos em que $\mathcal{C} = \{\varepsilon\} \cup \{\emptyset\} \cup \mathcal{A}$. Como $\{\varepsilon\}$, $\{\emptyset\}$ e $\mathcal{A}$ são apresentadas por autômatos, conforme ilustram as Figuras 2.2-2.4, concluímos da Propriedade 2.2 que as linguagens no fechamento regular $\bar{\mathcal{C}}$ de $\mathcal{C}$, chamadas de *linguagens regulares*, são apresentadas por autômatos (a prova deste resultado é realizada por indução do número de operadores regulares, partindo-se do conjunto $\emptyset \cup \{\varepsilon\} \cup \mathcal{A}$ [17, Teorema 5.2.1]). Como os elementos em $\bar{\mathcal{C}}$ são formados por expressões (compostas por um número finito de elementos em $\mathcal{C}$ e operadores regulares), estes também são chamados de *expressões regulares*, ou seja, uma expressão regular é uma forma de representar uma linguagem regular e a toda linguagem regular está associada uma

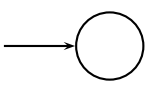


Figura 2.2: Linguagem $\emptyset$.

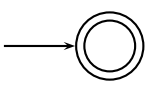


Figura 2.3: Linguagem $\{\varepsilon\}$.

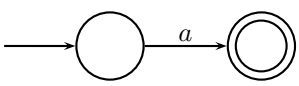


Figura 2.4: Linguagem $\{a\}$.

expressão regular.

Os autômatos nas Figuras 2.2-2.4 possuem no máximo um ramo e todos possuem expressões regulares associadas. Este é o ponto de partida para uma prova por indução (no número de ramos de um autômato finito) que a todo autômato finito está associado uma expressão regular, ou seja, dado um autômato A, $L(A)$ é uma linguagem regular. A prova algébrica que uma linguagem é apresentada por um autômato se, e somente se, esta é regular é apresentada em [17, Teorema 5.2.1]. Uma prova construtiva, baseada em algoritmos que geram um autômato dada uma expressão regular e uma expressão regular dado um autômato, é apresentada em [17, Seção 5.3]. Esse problema também é abordado em [16, Capítulo 2].

Em alguns algoritmos o contra domínio da função de rotulação é estendido $\lambda : \mathcal{E} \to \mathcal{A} \cup \{\varepsilon\}$. Quando no autômato há ramos com rótulo ε, este é dito com transição ε. Esta modificação não impõe restrições teóricas e algorítmicas, já que dado um autômato com transições ε é possível transformá-lo em um autômato não determinístico, bem como transformar este último em um determinístico. Em suma, os diferentes tipos de autômatos empregados para apresentar uma linguagem regular são equivalentes. Os autômatos determinísticos desempenham um papel importante no estudo de linguagens regulares, a saber, na teoria algébrica de linguagens é um autômato determinístico que está relacionado com o monóide sintático associado à linguagem. O algoritmo *accessible subset construction* utilizado para gerar autômatos determinísticos a partir de autômatos não determinísticos é apresentado em [17, Algoritmo 3.2.5] e [15, Capítulo 2].

Dado um vértice I para o qual não há caminhos originados em $\mathcal{I}$ que o alcançam, ou do qual não há caminhos para vértices em $\mathcal{T}$; este pode ser removido juntamente com os ramos $e \in \mathcal{E}$, $t(e) = I$ ou $i(e) = I$. Um autômato que não permite tal eliminação de vértices é dito *essencial*. Como estamos interessados em autômatos determinísticos e com o menor número de vértices, sempre iremos considerá-los essenciais.

Mínima apresentação determinística

Dado $A = (\mathcal{V}, \mathcal{E}, \lambda, \mathcal{I}, \mathcal{T})$ um autômato sem transições ε, seja $I \in \mathcal{V}$. O contexto à direita e à esquerda de I em A são, respectivamente, os conjuntos:

$$\mathsf{F}_A(I) = \{\lambda(\pi) |\ \pi \in P(A),\, i(\pi) = I \text{ e } t(\pi) \in \mathcal{T}\},$$
$$\mathsf{B}_A(I) = \{\lambda(\pi) |\ \pi \in P(A),\, i(\pi) \in \mathcal{I} \text{ e } t(\pi) = I\}.$$

Há conceitos similares para uma dada linguagem L e as palavras $w \in \mathcal{A}^*$. O contexto à direita e à esquerda de $w \in \mathcal{A}^*$ com respeito a L são, respectivamente, os conjuntos:

$$\mathsf{F}_L(w) = \{u|\ u \in \mathcal{A}^* \text{ e } wu \in L\},$$
$$\mathsf{B}_L(w) = \{u|\ u \in \mathcal{A}^* \text{ e } uw \in L\}.$$

Os conjuntos F_A e F_L são definidos para conjuntos de elementos distintos, enquanto F_A é definido para o conjunto de vértices do autômato A, F_L é definido para as palavras em $\mathcal{A}^*$. Assim, podemos escrever $\mathsf{F}_A(I)$ como $\mathsf{F}(I)$ e $\mathsf{F}_L(w)$ como $\mathsf{F}(w)$, pois que os argumentos I e w deixam claro que estamos nos referindo a F_A e F_L, respectivamente.

Dado $A = (\mathcal{V}, \mathcal{E}, \lambda, \mathcal{I}, \mathcal{T})$ obtemos $A_\sim = (\mathcal{V}/_\sim, \mathcal{E}_\sim, \lambda_\sim, \mathcal{I}_\sim, \mathcal{T}_\sim)$ pela aplicação da relação de equivalência $\sim$ sobre o conjunto $\mathcal{V}$ de A, a saber: $I \sim J$ se, e somente se, $\mathsf{F}(I) = \mathsf{F}(J)$; sendo esta relação chamada de redução sobre A. Os elementos de $\mathcal{V}/_\sim$ são as classes de equivalência obtidas pela aplicação de $\sim$ sobre $\mathcal{V}$ e formam o conjunto de vértices de $A_\sim$. Dado $I \in \mathcal{V}$ é comum denotar-se a classe de equivalência (ou vértice associado em $\mathcal{V}_\sim$) a qual I pertence por $[I]$. Dito isto, há um ramo $e_\sim \in \mathcal{E}_\sim$ de $[I]$ para $[J]$ com rótulo a se, e somente se, há um ramo $e \in \mathcal{E}$ com $i(e) \in [I]$, $t(e) \in [J]$ e $\lambda(e) = a$. Um vértice $[I]$ será inicial se, e somente se, existe $K \in [I]$ tal que $K \in \mathcal{I}$. Similarmente, $[J] \in \mathcal{T}_\sim$ se, e somente se, existe $K \in [J]$, $K \in \mathcal{T}$.

Um autômato é dito *reduzido* se as classes de equivalência de $\sim$ possuem um único elemento, assim $A_\sim$ é reduzido. Pelo processo de construção de $A_\sim$, este tem um número menor ou igual de estados que A. Como $L(A_\sim) = L(A)$ [17],[16], dado a apresentação A de uma linguagem L, podemos obter uma outra com menor número de vértices pela construção de $A_\sim$. A importância de se obter um autômato com menor número de vértices apresentando a mesma linguagem deve-se, por exemplo, a sua aplicação em algoritmos cuja complexidade cresce com seu número de vértices.

Pode-se mostrar que sendo A determinístico então $A_\sim$ é determinístico, portanto um autômato reduzido e determinístico pode ser obtido aplicando-se um algoritmo de determinização seguido por um de redução. Se um autômato finito reduzido e determinístico A tem um único estado inicial, então este é dito *o mínimo autômato determinístico* que apresenta $L(A)$, ou seja, é a apresentação determinística com menor número de vértices da linguagem $L(A)$. Para uma abordagem mais completa sobre redução de autômatos determinísticos, sugerimos [17, Capítulo 7] e [16, Seção 3.4]. Uma abordagem algébrica é apresentada em [15, Capítulo 3]. Estes algoritmos possuem complexidade assintótica n^2, onde n é o número de vértices. Para um algoritmo de complexidade assintótica $(n \cdot \log n)$ consultar [19].

2.3.1 Grafos

Um grafo (formalmente, um multi-grafo direcionado) pode ser interpretado como um autômato onde $\mathcal{V} = \mathcal{I} = \mathcal{T}$. Devido a redundância na especificação dos conjuntos $\mathcal{I}$ e $\mathcal{T}$, passamos a representá-lo como $G = (\mathcal{V}, \mathcal{E}, \lambda)$ (a troca do A pelo G é indicativo desta peculiaridade). Não havendo ambigüidade com relação aos conjuntos $\mathcal{V}$, $\mathcal{E}$ e a função λ, especificaremos o grafo por G.

Nosso interesse em grafos decorre da possibilidade de podermos apresentar alguns sistemas dinâmicos por meio destes. Inicialmente, se considerarmos $\mathcal{A} = \mathcal{E}$, então um caminho bi-infinito em G pode ser interpretado como um ponto em $\mathcal{E}^{\mathbb{Z}}$, sendo X_G o conjunto destes pontos (conhecido na literatura como *edge shift*). Como todos os vértices do grafo são iniciais e terminais, a coordenada 0 de um ponto é estabelecida arbitrariamente, o que implica em $\sigma(\mathsf{X}_G) = \mathsf{X}_G$. Se $x \in \mathcal{E}^{\mathbb{Z}}$ possui todos os seus fatores em $P(G)$, então existe um caminho bi-infinito em G que apresenta x, ou seja, um ponto não pertence a X_G se, e somente se, tiver algum fator em $\mathcal{E}^* \backslash P(G)$. Isto implica que $\mathcal{E}^{\mathbb{Z}} \backslash \mathsf{X}_G$ é aberto, assim X_G é fechado. Concluímos que o sistema simbólico gerado pelos caminhos bi-infinitos em um grafo é fechado. Pode-se mostrar que $\mathcal{B}(\mathsf{X}_G) = P(G)$.

Considerando G como uma forma de representar um sistema simbólico fechado, não há sentido mantermos vértices que não sejam visitados por um caminho bi-infinito em G. Assim, pela eliminação sucessiva de estados que não são terminais ou iniciais de algum ramo em $\mathcal{E}$, obtemos a componente essencial do grafo G. Tal procedimento é finito já que $\mathcal{V}$ é um conjunto finito. A componente essencial obtida por este processo é única, se a designarmos por H temos que $\mathsf{X}_G = \mathsf{X}_H$ [10, Proposição 2.2.10]

Voltando ao caso de uma função $\lambda : \mathcal{E} \to \mathcal{A}$ arbitrária. A função $\Phi = \lambda$ induz um SBC $\phi : \mathsf{X}_G \to \mathcal{A}^{\mathbb{Z}}$ com memória e antecipação iguais a zero. A imagem de um sistema simbólico fechado sob um SBC é um sistema simbólico fechado [10, Teorema 1.5.13], portanto, para um grafo G, $\phi(\mathsf{X}_G) = \lambda_\infty(\mathsf{X}_G)$ é um sistema simbólico fechado. Por este motivo, chamaremos $\phi(\mathsf{X}_G)$ um sistema simbólico regular, abreviado por SSR (conhecido na literatura como *sofic shift*). Note que um SSR é fechado, apesar de não estar explicito na sigla.

Um grafo G é *irredutível* se para todo par ordenado de vértices $(I, J) \in \mathcal{V} \times \mathcal{V}$ há um caminho $\pi \in P(G)$, $i(\pi) = I$ e $t(\pi) = J$. Assim, dado um SSR apresentado por um grafo G, onde $\lambda(e) = e$ (função identidade), se X_G é irredutível então G é irredutível [10, Proposição 2.2.14]. Quando G não é irredutível o chamamos de redutível, neste caso X_G é redutível, mas o mesmo não pode ser dito de $\lambda_\infty(\mathsf{X}_G)$ para uma função λ arbitrária.

Exemplo 2.5. Seja $X \subset \mathcal{A}^{\mathbb{Z}}$, $\mathcal{A} = \{a, b\}$, tal que, entre ocorrências consecutivas de b o número

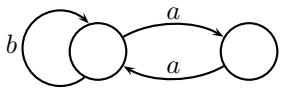


Figura 2.5: Apresentação do SSR par.

de ocorrências de a é par. X é chamado um sistema simbólico fechado par. O conjunto proibido associado à X é $\mathcal{F} = ba(aa)^*b$, portanto, a linguagem complementar à linguagem de X é $\mathcal{A}^*\mathcal{F}\mathcal{A}^* = (a+b)^*ba(aa)^*b(a+b)^*$. Como $\mathcal{A}^*\mathcal{F}\mathcal{A}^*$ é uma expressão regular, podemos representá-la por um autômato e, assim, à sua linguagem complementar com relação a $\mathcal{A}^*$, ou seja, a linguagem de X. Portanto, X é um SSR. Um grafo representando X é apresentado na Figura 2.5.

Um grafo G é chamado *de sincronização* se para qualquer vértice $I \in \mathcal{V}$ pode ser associado um bloco $u \in \mathcal{B}(X_G)$, tal que qualquer caminho $\pi \in P(G)$, $\lambda(\pi) = u$, tem-se que $t(\pi) = I$. A palavra u é chamada uma *palavra de sincronização de I em G*, sendo dito que u *sincroniza* para I. Grafos de sincronização desempenham um importante papel na determinação de apresentações com número mínimo de vértices, o que ficará claro à frente.

Definição 2.1. Sejam $G_i = (\mathcal{V}_i, \mathcal{E}_i, \lambda_i)$, $i = 1, 2$, dois grafos essenciais. Um homomorfismo Φ de G_1 para G_2 é definido pelo par de funções $\Phi_{\mathcal{V}} : \mathcal{V}_1 \to \mathcal{V}_2$ e $\Phi_{\mathcal{E}} : \mathcal{E}_1 \to \mathcal{E}_2$, tal que

$$\forall e_1 \in \mathcal{E}_1,\ \ i(\Phi_{\mathcal{E}}(e_1)) = \Phi_{\mathcal{V}}(i(e_1)),\ \ t(\Phi_{\mathcal{E}}(e_1)) = \Phi_{\mathcal{V}}(t(e_1)) \text{ e } \lambda_1(e_1) = \lambda_2(\Phi_{\mathcal{E}}(e_1)).$$

Dizemos que Φ é um isomorfismo quando $\Phi_{\mathcal{V}}$ e $\Phi_{\mathcal{E}}$ são sobrejetivas e injetivas.

Observe que, para o caso determinístico, dados $I = i(e_1)$ e $a = \lambda_1(e_1)$ podemos escrever a primeira equação como $i(\Phi_{\mathcal{E}}(e_1)) = \Phi_{\mathcal{V}}(I)$, o que aplicado na segunda equação resulta em $t(\Phi_{\mathcal{E}}(e_1)) = \delta(\Phi_{\mathcal{V}}(I), a)$ e $\Phi_{\mathcal{V}}(t(e_1)) = \Phi_{\mathcal{V}}(\delta(I, a))$. Assim, o homomorfismo passa a ser definido por uma única função $\Phi : \mathcal{V}_1 \to \mathcal{V}_2$ satisfazendo $\delta(\Phi(I), a) = \Phi(\delta(I, a))$. Neste caso, Φ é um isomorfismo quando é injetiva e sobrejetiva.

Mínima apresentação determinística de sincronização (m-SDP)

Alguns algoritmos que empregam grafos são tão mais complexos quanto maior a cardinalidade do conjunto $\mathcal{V}$, para alguns esta relação é exponencial. Isso justifica as recorrentes publicações de algoritmos que se propõem a gerar apresentações com número mínimo de vértices. Uma questão que antecede a proposição de tais algoritmos é: Dada uma apresentação, como saber se ela possui o menor número de vértices possível? Esta questão vem sendo respondida por partes. Inclusive com restrições

de maior interesse às apresentações de um SSR. Particularmente, procura-se responder esta questão para o caso de apresentações determinísticas. Aqui iremos abordá-la para o caso de apresentações determinísticas de sincronização, para o qual as apresentações com menor número de vértices são isomorfas, chamadas de *mínima apresentação determinística de sincronização* e abreviado por m-SDP (do inglês *minimal synchronizing deterministic presentation*).

Seja G um grafo determinístico, essencial e não reduzido apresentando um SSR irredutível $X = \lambda_\infty(\mathsf{X}_G)$. Há um máximo sub-grafo irredutível em G que apresenta $\mathcal{B}(X)$ [13]. Para obtenção do m-SDP, as classes de equivalência obtidas do conjunto de vértices deste sub-grafo são determinadas (relação de equivalência dada por $I \sim J$ se, e somente se, $\mathsf{F}(I) = \mathsf{F}(J)$). Por fim, o grafo reduzido é construído, no qual as classes de equivalência que foram determinadas são interpretadas como os vértices desta apresentação. O grafo reduzido obtido é o Shannon cover ou m-SDP do SSR irredutível, que é definido como uma apresentação reduzida, irredutível e determinística do SSR, sendo isomorfas quaisquer apresentações que satisfaçam tais condições [10, Proposição 3.3.17].

Para o caso de uma apresentação redutível G, como podemos determinar se esta tem o menor número de vértices? A resposta para esta questão foi dada em [13, Seção 5] para o caso de uma apresentação reduzida, determinística de sincronização. É demonstrado que qualquer tal apresentação é isomorfa a um sub-autômato terminal do mínimo autômato que apresenta o SSR (um sub-autômato $A' = (\mathcal{V}', \mathcal{E}', \lambda', \mathcal{I}', \mathcal{T}')$ de um autômato $A = (\mathcal{V}, \mathcal{E}, \lambda, \mathcal{I}, \mathcal{T})$ é terminal se para $I' \in \mathcal{V}'$ e $e \in \mathcal{E}$, $i(e) = I'$, então $t(e) \in \mathcal{V}'$). Aqui chegaremos ao mesmo resultado mostrando que duas apresentações SDP e reduzidas de um SSR são isomorfas.

Definição 2.2. Seja $L \in \mathcal{A}^*$ uma linguagem. Uma palavra $w \in \mathcal{A}^*$ é uma *constante* para L se, e somente se, para quaisquer $u_1, u_2, u_3, u_4 \in \mathcal{A}^*$, $u_1wu_2, u_3wu_4 \in L$, então $u_1wu_4 \in L$. Usaremos $C(L)$ para denotar o conjunto de constantes para L que pertencem a L.

Lema 2.1. *Seja $G = (\mathcal{V}, \mathcal{E}, \lambda)$ um SDP reduzido de um SSR $X = \lambda_\infty(\mathsf{X}_G)$. Uma palavra w é de sincronização para $I \in \mathcal{V}$ se, e somente se, $w \in C(\mathcal{B}(X))$.*

Demonstração: Seja $u_1wu_2 \in \mathcal{B}(X)$, então existe $\tau\pi\upsilon \in P(G)$ tal que $\lambda(\tau\pi\upsilon) = \lambda(\tau)\lambda(\pi)\lambda(\upsilon) = u_1wu_2$. Como w é de sincronização para I, $i(\upsilon) = t(\pi) = I$. Assim, para qualquer $u_3wu_4 \in \mathcal{B}(X)$, se $\lambda(\tau'\pi'\upsilon') = u_3wu_4$ então $i(\upsilon') = t(\pi') = I$, onde $\tau'\pi'\upsilon' \in P(G)$. Isso implica que $\tau\pi\upsilon' \in P(G)$ e assim que $\lambda(\tau\pi\upsilon') = u_1wu_4 \in \mathcal{B}(X)$.

Sejam $w \in C(\mathcal{B}(X))$ e $\pi_1, \pi_2 \in P(G)$ tais que $\lambda(\pi_1) = \lambda(\pi_2) = w$, $t(\pi_1) = I_1$, $i(\pi_1) = J_1$, $t(\pi_2) = I_2$ e $i(\pi_2) = J_2$. Como G é de sincronização, existem palavras w_1 e w_2 de sincronização para J_1 e J_2, respectivamente. Assim, para todo $v \in \mathsf{F}(I_1)$, w_1wv, $w_2w \in \mathcal{B}(X)$, $w_2wv \in \mathcal{B}(X)$.

Isto implica que $\mathsf{F}(I_1) \subseteq \mathsf{F}(I_2)$. Similarmente, $\mathsf{F}(I_2) \subseteq \mathsf{F}(I_1)$. Concluímos que $\mathsf{F}(I_1) = \mathsf{F}(I_2)$. Como G é reduzido, temos que $I_1 = I_2$. ■

Proposição 2.2. *Se $G = (\mathcal{V}, \mathcal{E}, \lambda)$ e $G' = (\mathcal{V}', \mathcal{E}', \lambda')$ são apresentações SDP e reduzidas de um SSR X, então estas são isomorfas.*

Demonstração: Seja $w \in C(\mathcal{B}(X))$ uma palavra de sincronização para $I \in \mathcal{V}$. Definimos $\Phi : \mathcal{V} \to \mathcal{V}'$, tal que $\Phi(I) = I'$ se há um caminho $\pi' \in P(G')$ para o qual $\lambda(\pi') = w$ e $t(\pi') = I'$. Inicialmente demonstraremos que Φ é bem definida, para isso consideremos $w, u \in C(\mathcal{B}(X))$. Do Lema 2.1, w e u são de sincronização em G, considerando que w e u sincronizam para o mesmo vértice $I \in \mathcal{V}$, então $\mathsf{F}(w) = \mathsf{F}(u)$. Como w e u também são palavras de sincronização em G' (novamente do Lema 2.1) então elas sincronizam para o mesmo vértice em $\mathcal{V}'$. Ou seja, Φ é uma função.

Agora provaremos que Φ é injetiva e sobrejetiva, respectivamente. Seja $\Phi(I_1) = \Phi(I_2) = I'$, então há palavras de sincronização $w_1, w_2 \in C(\mathcal{B}(X))$ que sincronizam para I_1 e I_2, respectivamente. Assim, w_1 e w_2 sincronizam para I' em G', isso implica que $\mathsf{F}(w_1) = \mathsf{F}(w_2)$. Como G é reduzido, concluímos que $I_1 = I_2$. Seguindo com a demonstração que Φ é sobrejetiva, seja w uma palavra de sincronização para $I' \in \mathcal{V}'$. Do Lema 2.1, $w \in C(\mathcal{B}(X))$ e, assim, também é de sincronização em G. Seja $I \in \mathcal{V}$ o vértice para o qual w sincroniza, então $\Phi(I) = I'$.

Para provarmos que Φ é um homomorfismo, consideremos que $a \in \mathcal{A}$ e w sincroniza para $I \in \mathcal{V}$. Se $\Phi(\delta(I, a)) = I'$ então wa sincroniza para I'. Como w sincroniza para a imagem de I em $\mathcal{V}'$, $\delta(\Phi(I), a) = I'$, ou seja, $\Phi(\delta(I, a)) = \delta(\Phi(I), a)$. Concluímos que G é isomorfo a G'. ■

2.4 Sistemas Simbólicos Fechados de Memória Finita (SFT)

Um sistema simbólico fechado de memória finita X (do inglês *shift of finite type*, abreviado por SFT) é um sistema simbólico fechado para o qual existe um conjunto proibido finito $\mathcal{F}$, ou seja $X = \mathsf{X}_{\mathcal{F}}$ para um $\mathcal{F}$ finito. Como propriedade característica de um SFT, sempre é possível determinar um inteiro não negativo M, tal que para todo $v \in \mathcal{B}(X)$, $|v| \geq M$, se $uv, vw \in \mathcal{B}(X)$ então $uvw \in \mathcal{B}(X)$. O menor M para o qual esta propriedade pode ser verificada é chamado *a memória* do SFT. Alguns autores definem a memória do SFT como o máximo comprimento de um bloco em $\mathcal{F}$. Estas denominações são idênticas quando o conjunto $\mathcal{F}$ é mínimo, o que será discutido em capítulos posteriores. Como propriedade complementar de um SFT X, se $\phi : X \to Y$ é um

SBC conjugado (o termo conjugado informa que ϕ é injetivo e sobrejetivo e, portanto, inversível [10, Teoema 1.5.14]) então Y também é um SFT [10, Teorema 2.1.10].

Seja $X \subseteq \mathcal{A}^{\mathbb{Z}}$ um SSR apresentado por $G = (\mathcal{V}, \mathcal{E}, \lambda)$. Dado qualquer bloco $e_1 e_2 \dots e_n \in \mathcal{E}^*$, então $e_1 e_2 \dots e_n \in \mathcal{B}(\mathsf{X}_G)$ se, e somente se, $t(e_i) = i(e_{i+1})$, $1 \leq i < n$. Ou seja, X_G é um SFT com $M = 1$. Isto implica que todo SSR é imagem de um SFT pela aplicação de um SBC sobrejetivo, neste caso induzido por $\Phi : \mathcal{B}_2(\mathsf{X}_G) \to \mathcal{A}$, onde $\Phi(e_1 e_2) = \lambda(e_2)$.

Adicionalmente, seja $Y \subseteq \mathcal{A}^{\mathbb{Z}}$ um SFT, podemos construir uma apresentação G para Y definida por: $G = (\mathcal{B}_M(Y), \mathcal{B}_{M+1}(Y), \lambda)$, tal que para todo $a_1 a_2 \dots a_{M+1} \in \mathcal{E}$ temos que $\lambda(a_1 a_2 \dots a_{M+1}) = a_{M+1}$, $i(a_1 a_2 \dots a_{M+1}) = a_1 a_2 \dots a_M$ e $t(a_1 a_2 \dots a_{M+1}) = a_2 \dots a_{M+1}$. Nestas apresentações só há um ramo de $a_1 a_2 \dots a_M \in \mathcal{V}$ para $b_1 b_2 \dots b_M \in \mathcal{V}$ se, e somente se, $a_1 a_2 \dots a_M b_M \in \mathcal{B}(Y)$. A função λ induz um SBC conjugado $\phi = \lambda_\infty$ de X_G para Y [10, Proposição 3.1.6]. Mais informações sobre esta construção de um SFT, partindo de um conjunto finito $\mathcal{F}$ arbitrário, é apresentada em [20]. Em [21] é apresentado um algoritmo matricial que determina o Shannon cover a partir desta apresentação reescrita em forma de matriz.

CAPÍTULO 3

ALGORITMO PARA DETERMINAÇÃO DO CONTEXTO À DIREITA: APLICAÇÃO NA CONSTRUÇÃO DA m-SDP

No Capítulo 2 definimos um SFT como um sistema simbólico fechado especificado por um conjunto proibido finito $\mathcal{F}$. Mostramos que todo SFT é um SSR (ver também [10, Teorema 3.1.5]), portanto podendo ser apresentado por um grafo $G = (\mathcal{V}, \mathcal{E}, \lambda)$ direcionado. A rotulação de um caminho bi-infinito em G gera uma seqüência que não contém fatores em $\mathcal{F}$. Sistemas práticos para armazenamento e transmissão de informação digital requerem a especificação de um conjunto $\mathcal{F}$ [22],[23]. Um SFT pode ser especificado por conjuntos $\mathcal{F}$ distintos, mas entre estes há um único conjunto $\mathcal{O}$ com número um mínimo de palavras proibidas, chamado a coleção mínima de palavras proibidas [10, p.33].

Os métodos propostos para obtenção do Shannon cover (apresentação reduzida, determinística e irredutível) de um SFT costumam ser divididos em duas etapas: inicialmente é gerada uma apresentação determinística do SFT a partir do conjunto $\mathcal{F}$ [10, Teorema 3.1.5], [14], [24]; seguida por um algoritmo para identificar as classes de vértices com mesmo contexto à direita e as conexões entre classes (vértices da nova apresentação) por ramos.

O método descrito em [10] é o apresentado na Seção 2.4 do Capítulo 2. A função de transição definida por este é simples de se calcular, mas o número de vértices do grafo inicial cresce exponencialmente com o máximo comprimento das palavras em $\mathcal{F}$, portanto isto o torna de pouco interesse prático. Há algoritmos com complexidade linear que são desenvolvidos aplicando-se conceitos da teoria de autômatos [14],[24]. Particularmente, no algoritmo descrito em [24], o conjunto de todos

os prefixos próprios das palavras do conjunto $\mathcal{O}$ formam os identificadores dos vértices da apresentação inicial.

O algoritmo descrito em [14] constrói inicialmente um autômato não determinístico e incompleto (o conjunto dos rótulos dos ramos que partem de um vértice é um subconjunto do alfabeto $\mathcal{A}$), tal que todas as palavras apresentadas por este possuem fatores em $\mathcal{F}$. Um autômato determinístico é obtido do não determinístico, fazendo-se dos vértices terminais os não terminais e vice-versa, obtendo-se um autômato determinístico que apresenta a linguagem do SFT [17]. Removendo-se os vértices não terminais deste autômato (com os ramos partindo deles e chegando a eles), obtém-se um grafo apresentando o SFT. A etapa intermediária deste algoritmo envolve a construção de um grafo determinístico a partir de um não determinístico (ver *accessible subset construction algorithm* [17, p.65]), procedimento que pode aumentar o número de estados em relação ao autômato inicial.

Neste capítulo apresentaremos um algoritmo para gerar o m-SDP de um SFT (o Shannon cover para um SFT irredutível). A entrada do algoritmo é a coleção mínima de palavras proibidas $\mathcal{O}$, a partir desta são determinadas classes de palavras com o mesmo contexto à direita, entre aquelas que pertencem ao conjunto $\mathcal{W}$ de todos os prefixos próprios de palavras em $\mathcal{O}$. Estas classes de palavras são empregadas como identificadores de vértices. A determinação destas classes é baseada na manipulação de palavras em $\mathcal{W}$ e $\mathcal{O}$, o que se diferencia da usual determinação de vértices com mesmo contexto à direita a partir de uma apresentação inicial [17], [19]. O algoritmo proposto associa a cada palavra $w \in \mathcal{W}$ o conjunto de sufixos próprios das palavras em $\mathcal{O}$ que quando concatenados com w geram uma palavra com um fator em $\mathcal{O}$. Palavras em $\mathcal{A}^*$ com o mesmo contexto à direita são determinadas pela igualdade de versões reduzidas destes conjuntos, assim podemos identificar as palavras em $\mathcal{W}$ que possuem o mesmo contexto à direita, separando-as em classes de equivalência. Definidas as classes de equivalência (ou os vértices da apresentação mínima), usamos uma função de transição, baseada em uma definida em [24], para construirmos o Shannon cover. Mostraremos que esta apresentação é isomorfa à componente essencial do grafo de contextos do SFT (Seção 3.3).

3.1 Alguns Conceitos e Definições

Iniciamos esta seção apresentando formalmente a coleção mínima de palavras proibidas. Uma palavra $w = w_1 \dots w_n$ é dita uma *proibição mínima* se $w \notin L$, mas $w_1 \dots w_{n-1} \in L$ e $w_2 \dots w_n \in L$. A coleção de todas as proibições mínimas é única [10, p.12], de forma que se $\mathcal{F} \subseteq \mathcal{O}$ e $X_{\mathcal{F}} = X_{\mathcal{O}}$ então $\mathcal{F} = \mathcal{O}$. Um SFT com coleção mínima de palavras proibidas $\mathcal{O}$ tem memória M se o máximo comprimento das palavras em $\mathcal{O}$ é $M + 1$. Na Seção 3.2, utilizamos a unicidade do conjunto $\mathcal{O}$ para

propor um procedimento que identifique palavras com o mesmo contexto à direita.

Seja $\{\mathcal{B}_i\}_{i=1}^N$ uma coleção de subconjuntos de $\mathcal{A}$ e suponha que queremos classificá-los de acordo com seus contextos, onde o critério é a igualdade. Esta classificação pode ser feita a partir de $\{\mathcal{B}_i\}_{i=1}^N$ ou $\{\mathcal{B'}_i\}_{i=1}^N$, o que for mais conveniente, uma vez que, se $\mathcal{B}_j = \mathcal{B}_k$ então $\mathcal{B'}_j = \mathcal{B'}_k$. Podemos escolher uma das coleções e chamá-la de $\{\mathcal{D}_i\}_{i=1}^N$. Caso tenhamos informações adicionais sobre o conjunto $\{\mathcal{D}_i\}_{i=1}^N$, estas podem ser utilizadas para simplificar a comparação. Como exemplo, suponha que o conjunto $\mathcal{G} \subseteq \bigcap_{i=1}^N \mathcal{D}_i$ é conhecido, esta informação poderá ser utilizada para simplificar o processo de determinação de igualdade, que poderá ser realizado sobre o conjunto $\{\mathcal{D}_i \backslash \mathcal{G}\}_{i=1}^N$ no lugar de $\{\mathcal{D}_i\}_{i=1}^N$.

Dada uma linguagem qualquer L (não necessariamente fatorial), uma importante característica de uma palavra $w \in \mathcal{A}^*$ é o seu conjunto sintático, definido como $C(w) = \{(x, y) | (x, y) \in \mathcal{A}^* \times \mathcal{A}^*$ e $xwy \in L\}$ [17, p. 210]. O complemento deste conjunto é $C'(w) = \{(p, s) | (p, s) \in \mathcal{A}^* \times \mathcal{A}^*$ e $pws \notin L\}$. Considerando L como a linguagem de um sistema simbólico fechada, dada uma palavra $w \in L$, $C'(w) = C'(w) \cup ((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*))$, já que para qualquer $(p, s) \in ((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*))$, $pws \notin L$. Concluímos que $((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*))$ equivale ao conjunto $\mathcal{G}$ discutido no parágrafo anterior. Portanto, o conjunto sintático de uma palavra $w \in L$ pode ser especificado por um subconjunto de $L \times L$, sendo este $C'(w) \backslash ((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*))$. Mostraremos na próxima seção que este conjunto não é a forma mais simples de se representar o conjunto sintático de uma palavra. Como a linguagem de um SFT é fatorial, para todo $(p, s) \in C'(w) \backslash ((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*)) \subseteq L \times L$, o conjunto $\{(p'p, ss') | (p', s') \in (Lp^{-1}) \times (s^{-1}L)\}$ é um subconjunto de $C'(w) \backslash ((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*))$, o que motiva à Definição 3.1.

Definição 3.1. Seja $(p, s) \in L \times L$, então $[(p, s)] = \{(p'p, ss') | (p', s') \in (Lp^{-1}) \times (s^{-1}L)\}$. Como extensão, seja $\mathcal{A} \subseteq L \times L$, se $\mathcal{A} \neq \emptyset$ então $[\mathcal{A}] = \{(p', s') | (p', s') \in [(p, s)]$ para todo $(p, s) \in \mathcal{A}\}$, caso contrário $[\mathcal{A}] = \emptyset$.

Exemplo 3.1. Seja $\mathcal{A} = \{0, 1\}$ e $\mathcal{F} = \{011, 110, 001, 100\}$ o conjunto proibido do SFT de linguagem $L = \mathcal{B}(X_\mathcal{F})$. Assim, para $(11, 0) \in L \times L$ temos que $[(1, 0)] = \{(p, q) \mid p \in 1^*1$ e $q \in 0(0^* + (10)^*)\}$, onde $'+'$ indica união e $w^* = \{\varepsilon, w, ww, www, \ldots\}$.

Segue diretamente da definição que $(p, s) \in [(p, s)]$, pois $\varepsilon \in L$. A próxima definição apresenta um subconjunto de $C'(w) \backslash ((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*)) \subseteq L \times L$ que determina $C(w)$.

Definição 3.2. Um conjunto das restrições de $w \in \mathcal{A}^*$, dada uma linguagem L, denominado de

$\mathcal{C}(w)$, satisfaz a equação $[\mathcal{C}(w)] = \{(p,s)|(p,s) \in L \times L \text{ e } pws \notin L\} = C'(w)\backslash((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*))$.

Empregando a Definição 3.2 podemos escrever o conjunto sintático a partir do conjunto das restrições como $C(w) = \{\ [\mathcal{C}(w)] \cup ((\mathcal{A}^*\mathcal{F}\mathcal{A}^* \times \mathcal{A}^*) \cup (\mathcal{A}^* \times \mathcal{A}^*\mathcal{F}\mathcal{A}^*))\ \}'$.

Exemplo 3.2. Considerando o SFT apresentado no Exemplo 3.1, temos que $\mathcal{C}(0) = \{(\varepsilon,01), (\varepsilon,11), (11,\varepsilon), (10,\varepsilon), (1,0), (0,1)\}$ ou $\mathcal{C}(0) = \{(\varepsilon,01), (\varepsilon,11), (11,1), (11,0), (11,\varepsilon), (10,\varepsilon), (1,0), (0,1)\}$, já que $[\{(\varepsilon,01), (\varepsilon,11), (11,\varepsilon), (10,\varepsilon), (1,0), (0,1)\}] = [\{(\varepsilon,01), (\varepsilon,11), (11,1), (11,0), (11,\varepsilon), (10,\varepsilon), (1,0), (0,1)\}]$.

Exemplo 3.3. Se $w \notin L$ então para qualquer $(p,s) \in L \times L$ temos que $pws \notin L$, portanto, um possível conjunto das restrições para w é $\mathcal{C}(w) = \{(\varepsilon,\varepsilon)\}$, pois que ε é prefixo e sufixo de qualquer palavra. Assim, para $w \notin L$ temos que $C'(w) = \mathcal{A}^* \times \mathcal{A}^*$.

No que segue, nosso interesse se restringe a três subconjuntos de $\mathcal{C}(w)$, são eles: $\mathcal{C}^l(w) = \{(p,s)|p \neq \varepsilon \text{ e } s = \varepsilon\}$, $\mathcal{C}^c(w) = \{(p,s)|p \neq \varepsilon \text{ e } s \neq \varepsilon\}$ e $\mathcal{C}^r(w) = \{(p,s)|p = \varepsilon \text{ e } s \neq \varepsilon\}$, sendo chamados de conjunto das restrições à esquerda, condicional e à direita, respectivamente.

Nota 3.1. Seja $(\varepsilon,s') \in [\mathcal{C}(w)]$, portanto há $(p,s) \in \mathcal{C}(w)$ tal que $(\varepsilon,s') \in [(p,s)]$, assim $p \in \mathcal{S}(\varepsilon)$ e $s \in \mathcal{P}(s')$, logo $p = \varepsilon$. Como $p \neq \varepsilon$ para todo $(p,s) \in \mathcal{C}^l(w) \cup \mathcal{C}^r(w)$, então $(\varepsilon,s') \notin [\mathcal{C}^l(w) \cup \mathcal{C}^c(w)]$, portanto $(\varepsilon,s') \in [\mathcal{C}(w)\backslash(\mathcal{C}^l(w) \cup \mathcal{C}^c(w))] = [\mathcal{C}^r(w)]$.

Proposição 3.1. *Para todo $w \in L$, $(\varepsilon,s) \in [\mathcal{C}^r(w)]$ se, e somente se, $s \in \mathsf{F}'(w)$. Logo, $[\mathcal{C}^r(w)] = [\mathcal{C}^r(w')]$ se, e somente se, $\mathsf{F}(w) = \mathsf{F}(w')$.*

Demonstração: O contexto à direita de uma palavra $w \in L$ pode ser escrito como $\mathsf{F}(w) = L\backslash\mathsf{F}'(w)$, onde $\mathsf{F}'(w) = \{s|ws \notin L \text{ e } s \in L\}$. Contudo, se $s \in \mathsf{F}'(w)$ então $ws \notin L$, logo $(\varepsilon,s) \in [\mathcal{C}(w)]$, da Nota 3.1 concluímos que $(\varepsilon,s) \in [\mathcal{C}^r(w)]$. Por outro lado, se $(\varepsilon,s) \in [\mathcal{C}^r(w)]$ então $ws \notin L$, logo $s \in \mathsf{F}'(w)$. Concluímos que os conjuntos $\mathsf{F}(w)$ e $\mathsf{F}(w')$ podem ser determinados unicamente dos conjuntos $[\mathcal{C}^r(w)]$ e $[\mathcal{C}^r(w')]$, respectivamente. ■

Podemos usar os conjuntos $\mathcal{C}^r(w)$ para determinar os conjuntos $\mathsf{F}'(w)$ e $\mathsf{F}(w)$, já que $\mathsf{F}'(w) = \{s|(\varepsilon,s) \in [\mathcal{C}^r(w)]\}$ e $\mathsf{F}(w) = \{s|s \in L \text{ and } (\varepsilon,s) \notin [\mathcal{C}^r(w)]\}$, respectivamente. Na próxima seção abordamos o problema de determinar palavras com o mesmo contexto à direita a partir dos conjuntos $\mathcal{C}^r(w)$.

3.2 Determinação de palavras com o mesmo contexto à direita

Provamos na Proposição 3.1 que a igualdade dos conjuntos $[\mathcal{C}^r(w)]$ e $[\mathcal{C}^r(w')]$ implica que w e w' têm o mesmo contexto à direita. Mas esta igualdade não assegura que $\mathcal{C}^r(w) = \mathcal{C}^r(w')$, já que, em geral, estes conjuntos podem ser escritos de diferentes maneiras (ver Exemplo 3.2). Propomos nesta seção uma construção particular para o conjunto $\mathcal{C}^r(w)$, que nos permite determinar palavras de mesmo contexto à direita pela simples igualdade deste conjunto.

Definição 3.3. Seja $w \in L$ e $\mathcal{O}$ a coleção mínima de palavras proibidas de um SFT. Então $\mathcal{C}^r_{\mathcal{O}}(w) = \{(\varepsilon, s) | s \in ((\mathcal{S}(w)\backslash\{\varepsilon\})^{-1}\mathcal{O})\}$, sendo o conjunto de todos os pares ordenados $(\varepsilon, s) \in L \times L$, $|s| \geq 1$, para os quais há $p \in \mathcal{S}(w)\backslash\{\varepsilon\}$ tal que $ps \in \mathcal{O}$.

Nota 3.2. Caso $(\mathcal{S}(w)\backslash\{\varepsilon\}) \cap \mathcal{P}(\mathcal{O}\mathcal{A}^{-1}) = \emptyset$ então $ws \in L$ para todo $s \in L$, assim $\mathcal{C}^r_{\mathcal{O}}(w) = \emptyset$, portanto $\mathcal{C}^r_{\mathcal{O}}(u) = \emptyset$ para todo $u \in \mathcal{S}(w)\backslash\{\varepsilon\}$. Contudo, se $\mathcal{C}^r_{\mathcal{O}}(w) = \emptyset$ então, para todo $s \in L$, ws não possui fatores em $\mathcal{O}$, implicando que $\mathcal{C}^r(w) = \emptyset$, uma vez que $\mathcal{O}$ é a coleção mínima de palavras proibidas. Similarmente, se $\mathcal{C}^r(w) = \emptyset$ então $\mathcal{C}^r_{\mathcal{O}}(w) = \emptyset$.

Na próxima proposição provaremos que $[\mathcal{C}^r_{\mathcal{O}}(w)]$ é um conjunto das restrições de uma palavra $w \in L$.

Proposição 3.2. *Seja $w \in L$ e $\mathcal{O}$ a coleção mínima de palavras proibidas do SFT. Então $[\mathcal{C}^r(w)] = [\mathcal{C}^r_{\mathcal{O}}(w)]$.*

Demonstração: Da Nota 3.2 $\mathcal{C}^r_{\mathcal{O}}(w) = \emptyset$ se, e somente se, $\mathcal{C}^r(w) = \emptyset$, assim $[\mathcal{C}^r_{\mathcal{O}}(w)] = [\mathcal{C}^r(w)] = \emptyset$ para este caso.

Suponha que $\mathcal{C}^r_{\mathcal{O}}(w)$ e $\mathcal{C}^r(w)$ não são conjuntos vazios. Se $(\varepsilon, s) \in [\mathcal{C}^r_{\mathcal{O}}(w)]$ então $ws \notin L$, implicando que $(\varepsilon, s) \in [\mathcal{C}^r(w)]$, logo $[\mathcal{C}^r_{\mathcal{O}}(w)] \subseteq [\mathcal{C}^r(w)]$. Se supormos que $(\varepsilon, s) \in [\mathcal{C}^r(w)]$, então $ws \notin L$, implicando que há $p = p_1 \dots p_n \in \mathcal{S}(w)$ e $q = q_1 \dots q_m \in \mathcal{P}(s)$ tal que $pq \notin L$, $p_2 \dots p_n q_1 \dots q_m \in L$ e $p_1 \dots p_n q_1 \dots q_{m-1} \in L$, logo $p_1 \dots p_n q_1 \dots q_m \in \mathcal{O}$ e ws tem um fator em $\mathcal{O}$, portanto $(\varepsilon, s) \in [\mathcal{C}^r_{\mathcal{O}}(w)]$, o que nos permite concluir que $[\mathcal{C}^r(w)] \subseteq [\mathcal{C}^r_{\mathcal{O}}(w)]$ e o resultado segue. ■

Uma forma de cálculo dos $\mathcal{C}^r_{\mathcal{O}}(w)$ é por recursão. Para um dado $w \in L$, $w = a_1 a_2 \dots a_n$, o cálculo é feito do prefixo de menor comprimento não nulo (a_1) para o de maior comprimento

$(a_1a_2\ldots a_n)$, empregando a definição $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w)) = \{s \mid (\varepsilon, s) \in \mathcal{C}^r_{\mathcal{O}}(w)\}$, temos que:

$$\begin{aligned}
\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1)) &= a_1^{-1}\mathcal{O}, \\
\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1a_2)) &= a_2^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1)) \cup a_2^{-1}\mathcal{O}, \\
&\vdots \\
\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1a_2\ldots a_n)) &= a_n^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1\ldots a_{n-1})) \cup a_n^{-1}\mathcal{O}.
\end{aligned} \tag{3.1}$$

A prova desta relação segue pelo desenvolvimento do conjunto $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w))$:

$$\begin{aligned}
\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w)) &= (\mathcal{S}(w)\backslash\{\varepsilon\})^{-1}\mathcal{O} \\
&= (\mathcal{S}(a_1\ldots a_{n-1})a_n)^{-1}\mathcal{O} \\
&= (\mathcal{S}(a_1\ldots a_{n-1})a_n \cup a_n)^{-1}\mathcal{O} \\
&= a_n^{-1}\mathcal{S}(a_1\ldots a_{n-1})^{-1}\mathcal{O} \cup a_n^{-1}\mathcal{O} \\
&= a_n^{-1}((\mathcal{S}(a_1\ldots a_{n-1})\backslash\{\varepsilon\})^{-1}\mathcal{O}) \cup a_n^{-1}\mathcal{O} \cup a_n^{-1}\mathcal{O} \\
&= a_n^{-1}((\mathcal{S}(a_1\ldots a_{n-1})\backslash\{\varepsilon\})^{-1}\mathcal{O}) \cup a_n^{-1}\mathcal{O} \\
&= a_n^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1\ldots a_{n-1})) \cup a_n^{-1}\mathcal{O}.
\end{aligned}$$

Exemplo 3.4. Seja $w = 1010$ uma palavra pertencente a linguagem de um SFT com símbolos no alfabeto $\mathcal{A} = \{0, 1, 2\}$ e especificado pelo conjunto $\mathcal{O} = \{002, 100, 1011, 1012, 111, 112, 20, 211, 212\}$. Começamos calculando $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(1)) = 1^{-1}\mathcal{O} = \{00, 011, 012, 11, 12\}$, seguimos calculando $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(10)) = 0^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(1)) \cup 0^{-1}\mathcal{O} = \{0, 11, 12, 02\}$, $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(101)) = 1^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(10)) \cup 1^{-1}\mathcal{O} = \{1, 2, 00, 011, 012, 11, 12\}$ e finalmente $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(1010)) = 0^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(101)) \cup 0^{-1}\mathcal{O} = \{0, 11, 12, 02\}$.

Observamos que quando $(\varepsilon, s), (\varepsilon, s') \in \mathcal{C}^r_{\mathcal{O}}(w)$ e $s' \in \mathcal{P}(s)$, implicando que $(\varepsilon, s) \in [(\varepsilon, s')]$, então $[\mathcal{C}^r_{\mathcal{O}}(w)] = [\mathcal{C}^r_{\mathcal{O}}(w)\backslash(\varepsilon, s)]$. Um conjunto $\mathcal{C}^r(w)$ será escrito como $\tilde{\mathcal{C}}^r(w)$ quando esta eliminação de elementos não puder ser realizada.

Exemplo 3.5. A partir do conjunto $\mathcal{C}^r_{\mathcal{O}}(10)$ determinado no Exemplo 3.4, $\tilde{\mathcal{C}}^r_{\mathcal{O}}(10) = \{(\varepsilon, s) | s \in \{0, 11, 12\}\}$, uma vez que $0 \in \mathcal{P}(02)$.

No próximo teorema mostraremos que palavras com o mesmo contexto à direita podem ser determinadas pela igualdade dos seus conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$.

Teorema 3.3. *Seja $w, w' \in L$, então $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ se, e somente se, $\mathsf{F}(w) = \mathsf{F}(w')$.*

Demonstração: Se $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ então $[\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)] = [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w')]$, o que das Proposições 3.2 e 3.1 implica que $\mathsf{F}(w) = \mathsf{F}(w')$.

Supondo que $(\varepsilon, s) \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ e $(\varepsilon, s) \notin \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ concluiremos que $[\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)] \neq [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w')]$, portanto da Proposição 3.1 temos que $\mathsf{F}(w) \neq \mathsf{F}(w')$. Portanto, a hipótese que $\mathsf{F}(w) = \mathsf{F}(w')$ implica que $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$, caso contrário, se $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) \neq \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ concluímos que $\mathsf{F}(w) \neq \mathsf{F}(w')$, o que contradiz a hipótese. Se $(\varepsilon, s) \notin [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w')]$ então $[\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)] \neq [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w')]$, logo consideraremos que $(\varepsilon, s) \in [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w')]$. Para que $(\varepsilon, s) \in [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w')]$ é necessário que exista $(\varepsilon, s') \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ tal que $s' \in \mathcal{P}(s)$ e $|s'| < |s|$. Já que $(\varepsilon, s) \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$, então não há prefixos próprios u de s tal que $(\varepsilon, u) \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$, portanto, concluímos que $(\varepsilon, s') \notin [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)]$ e $(\varepsilon, s') \in [\tilde{\mathcal{C}}^r_{\mathcal{O}}(w')]$. ■

3.3 Construção do m-SDP

Nas últimas seções definimos o conjunto $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ e mostramos como utilizá-lo para determinar se palavras em $\mathcal{A}^*$ possuem o mesmo contexto à direita. Nesta seção utilizamos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ para construir a apresentação determinística, de sincronização e reduzida do SFT. Mostraremos que esta apresentação é isomorfa ao grafo dos contextos do SFT. Portanto, as propriedades demonstradas para a apresentação propostas são estendidas para o grafo dos contextos.

O grafo dos contextos é uma apresentação determinística do SFT [10, p. 73] gerada a partir do conjunto $\mathcal{C} = \{\mathsf{F}(w) | w \in L\}$, que é o conjunto de todos os contextos à direita de palavras em L ($\mathcal{C}$ também pode ser interpretado como o conjunto dos vértices do grafo dos contextos). Há um ramo neste grafo com rótulo $a \in \mathcal{A}$ partindo de um vértice $\mathcal{I} = \mathsf{F}(w) \in \mathcal{C}$, se $wa \in L$, sendo o vértice terminal $\mathcal{J} = \mathsf{F}(wa) \in \mathcal{C}$. Pela aplicação deste procedimento a $\mathcal{I} \in \mathcal{C}$ e $a \in \mathcal{A}$ obtemos o grafo dos contextos do SFT. O conjunto $\mathcal{C}$ de um SFT é finito [10, Teorema 3.2.10], logo este procedimento é realizável.

Seja $G = (\mathcal{V}, \mathcal{E}, \mathcal{L})$ a componente essencial do grafo dos contextos de um SFT. Propomos um algoritmo alternativo para gerar G. Este algoritmo segue as etapas: (i) Encontrar um conjunto $\mathcal{W} \subseteq L$ tal que $\mathcal{V} \subseteq \{\mathsf{F}(w) | w \in \mathcal{W}\}$; (ii) determinar a partição $\mathcal{P} = \{C_i\}_{i=1}^n$ de $\mathcal{W}$, satisfazendo a condição $w, w' \in C_i$ se, e somente se, $\mathsf{F}(w) = \mathsf{F}(w')$. Cada C_i é um subconjunto de $\mathcal{W}$ contendo todas as palavras deste com o mesmo contexto à direita. (iii) Definir uma função de transição de $\mathcal{P} \times \mathcal{A}$ para $\mathcal{P}$, como segue: seja $C_i \in \mathcal{P}$ e $w \in C_i$, se $wa \in L$, $a \in \mathcal{A}$, então há um ramo com rótulo a de C_i para C_j, tal que $w' \in C_j$ onde $\mathsf{F}(w') = \mathsf{F}(wa)$; finalmente, extraímos a componente essencial do grafo obtido no terceiro passo. Seguiremos determinando o conjunto $\mathcal{W}$, propondo um método para realizarmos a partição deste e definindo a função de transição.

Em [24] é apresentado um algoritmo para construir um autômato completo com um único vértice

inicial i e apresentando uma linguagem fatorial. Os vértices terminais deste autômato formam um sub-grafo que apresenta a linguagem de um SSR. Este sub-grafo não é, necessariamente, essencial nem reduzido. À cada vértice do conjunto $\mathcal{V}$ deste sub-grafo é associada uma palavra do conjunto $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$, da seguinte forma: todo vértice I é alcançado a partir de i por uma palavra $w \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$, implicando que $\mathsf{F}(I) = \mathsf{F}(w)$. Seja $\mathsf{F}(u) \in \mathcal{V}$, como o sub-grafo é uma apresentação do SSR, este tem um caminho π, tal que $i(\pi) = i$, $\mathcal{L}(\pi) = u$ e $\mathsf{F}(t(\pi)) = \mathsf{F}(u)$. Suponha que $w \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$ alcança $t(\pi)$ a partir de i, então $\mathsf{F}(w) = \mathsf{F}(u)$. Concluímos que o conjunto de vértices $\mathcal{V}$ da componente essencial do grafo dos fatores é um subconjunto de $\{\mathsf{F}(w) | w \in \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})\}$, portanto consideramos $\mathcal{W} = \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$.

A partição $\mathcal{P}$ é determinada pelos conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ como especificado a seguir. Seja $\mathcal{P}' = \{C'_i\}_{i=1}^m$ uma partição de $\mathcal{W}$, tal que: $w, w' \in C'_i$ se, e somente se, $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$. Empregando o Teorema 3.3, $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ se, e somente se, $\mathsf{F}(w) = \mathsf{F}(w')$, o que nos leva a concluir que $\mathcal{P} = \mathcal{P}'$.

Já definimos o conjunto $\mathcal{W}$ e a partição $\mathcal{P}$. Antes de definirmos a função de transição apresentaremos a relação que segue e por uma restrição no domínio desta relação obteremos a função de transição.

Definição 3.4. *Definimos a relação* $\delta : \mathcal{P} \times (\mathcal{A} \cup \{\varepsilon\}) \longrightarrow \mathcal{P} \cup \{\mathcal{O}\}$ *como*

$$\delta(C_i, a) = \{\mathcal{Q} | \textit{ fixado } w \in C_i, v \in \mathcal{Q}, |v| = \max_{u \in \mathcal{T}(wa)} |u|\},$$

onde $\mathcal{Q} \in \mathcal{P} \cup \{\mathcal{O}\}$ *e* $\mathcal{T}(w) = \mathcal{S}(w) \cap (\mathcal{W} \cup \mathcal{O})$.

A imagem $\delta(C_i, a)$ de (C_i, a) é um elemento $\mathcal{Q}$ em $\mathcal{P} \cup \{\mathcal{O}\}$ associado com uma palavra fixada $w \in C_i$, tal que $v \in \mathcal{Q}$ é o maior sufixo de wa em $\mathcal{W} \cup \mathcal{O}$. Segue desta definição que, se $wa \in \mathcal{O}$ então $\delta(C_i, a) = \{\mathcal{O}\}$.

O Teorema 3.6 demonstra, para qualquer $(C_i, a) \in \mathcal{P} \times (\mathcal{A} \cup \{\varepsilon\})$, que o valor de $\delta(C_i, a)$ não depende da escolha de uma palavra $w \in C_i$ específica. O que nos permite concluir que δ é uma função bem definida sobre o conjunto $\mathcal{P} \times (\mathcal{A} \cup \{\varepsilon\})$. Para demonstração deste teorema precisamos dos resultados dos próximos dois lemas.

Lema 3.4. *Para* $w, w' \in L$, *seja* $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ *e* $a \in \mathcal{A}$, *então* $\tilde{\mathcal{C}}^r_{\mathcal{O}}(wa) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w'a)$.

Demonstração: Pelo Teorema 3.3, $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ se, e somente se, $\mathsf{F}(w) = \mathsf{F}(w')$. Logo $waz \in L$ se, e somente se, $w'az \in L$, portanto $\mathsf{F}(wa) = \mathsf{F}(w'a)$, ou $\tilde{\mathcal{C}}^r_{\mathcal{O}}(wa) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w'a)$. ■

Lema 3.5. *Seja* $w \in L$ *e* v *o mais longo elemento em* $\mathcal{S}(w) \cap \mathcal{W}$, *então* $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(v)$.

Demonstração: Segue do processo de construção dos conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ e $\tilde{\mathcal{C}}^r_{\mathcal{O}}(v)$ que $\tilde{\mathcal{C}}^r_{\mathcal{O}}(v) \subseteq \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$, uma vez que $v \in \mathcal{S}(w)$. Provaremos agora a inclusão reversa. Seja $(\varepsilon, s) \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$, logo existe $p \in \mathcal{S}(w)\backslash\{\varepsilon\}$ tal que $ps \in \mathcal{O}$, portanto $p \in \mathcal{S}(w) \cap \mathcal{W}$. Já que v é a palavra mais longa em $\mathcal{S}(w) \cap \mathcal{W}$, então $p \in \mathcal{S}(v)$ e, portanto, $(\varepsilon, s) \in \mathcal{C}^r_{\mathcal{O}}(v)$. Se existe $s' \in \mathcal{P}(s)$ para o qual $(\varepsilon, s') \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(v)$, então $(\varepsilon, s') \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$, pois $\tilde{\mathcal{C}}^r_{\mathcal{O}}(v) \subseteq \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$, contradizendo a hipótese de que $(\varepsilon, s) \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$, logo $(\varepsilon, s) \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(v)$. Assim, concluímos que $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) \subseteq \tilde{\mathcal{C}}^r_{\mathcal{O}}(v)$, por fim $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(v)$. ■

Teorema 3.6. *A relação δ é uma função bem definida.*

Demonstração: Sejam $w, w' \in C_i$, C_i um elemento de $\mathcal{P}$, e $a \in \mathcal{A}$. Como $\mathsf{F}(w) = \mathsf{F}(w')$, se $wa \notin L$ então $w'a \notin L$, logo $\delta(C_i, a) = \{\mathcal{O}\}$. Portanto, consideremos que $wa, w'a \in L$, sendo v e v' as palavras mais longas em $\mathcal{S}(wa) \cap \mathcal{W}$ e $\mathcal{S}(w'a) \cap \mathcal{W}$, respectivamente. Do Lema 3.5 temos que $\tilde{\mathcal{C}}^r_{\mathcal{O}}(wa) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(v)$ e $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w'a) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(v')$. Por fim, do Lema 3.4 $\tilde{\mathcal{C}}^r_{\mathcal{O}}(v) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(v')$. ■

A Tabela 3.1 resume o algoritmo proposto para gerar uma apresentação de um SFT a partir do seu conjunto $\mathcal{O}$.

Exemplo 3.6. Seja $\mathcal{O} = \{002, 100, 1011, 1012, 111, 112, 20, 211, 212\}$, o mesmo conjunto usado no Exemplo 3.4. O conjunto dos prefixos próprios de $\mathcal{O}$, $\mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$, é $\mathcal{W} = \{\varepsilon, 0, 1, 2, 00, 10, 101, 11, 21\}$. A Tabela 3.2 mostra os conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ para $w \in \mathcal{W}$, a partição $\mathcal{P}$ (ou classes de equivalência) e o resultado do cálculo da função δ, mostrando-se as palavras mais longas em $\mathcal{S}(wa) \cap \mathcal{W}$ para $wa \in L$. Se $wa \notin L$, a entrada respectiva na tabela é preenchida com $\{\mathcal{O}\}$. Podemos observar que $\tilde{\mathcal{C}}^r_{\mathcal{O}}(2) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(10)$, assim o conjunto $\{2, 10\} \in \mathcal{W}$ forma a classe C_4. Prosseguindo, obtemos as classes $C_1 = \{\varepsilon\}$, $C_2 = \{0\}$, $C_3 = \{1\}$, $C_5 = \{00\}$ e $C_6 = \{11, 21, 101\}$. Portanto, $\mathcal{P} = \{\{\varepsilon\}, \{0\}, \{1\}, \{2, 10\}, \{00\}, \{11, 21, 101\}\}$. Observemos também que a palavra mais longa em $\mathcal{S}(w1) \cap \mathcal{W}$, $w \in C_4$, é 21 se $w = 2$ ou 101 se $w = 10$, contudo $\{21, 101\} \in C_6$, o que nos leva a concluir que $\delta(C_4, 1) = C_6$.

Como estamos interessados em uma apresentação de um SFT, podemos restringir a função δ para $\mathcal{A} = \{(C_i, a) | \delta(C_i, a) \in \mathcal{P}\}$. O que estamos fazendo é eliminando os ramos que incidem no estado $\mathcal{O}$, que por construção não possui ramos partindo dele, podendo ser removido do grafo juntamente com os ramos que partem ou incidem nele, sem que com isso alteremos a linguagem do SSR apresentada pelo grafo. Esta restrição é representada por $\delta|\mathcal{A}$ e chamada função de transição, mas por simplicidade de notação, iremos representá-la por δ. No próximo teorema mostraremos que o grafo H gerado pelas linhas 1 a 16 do algoritmo descrito na Tabela 3.1 é isomorfo ao grafo dos

Tabela 3.1: Descrição do Algoritmo.

PASSO	INSTRUCÃO
1.	calcule $\mathcal{W} = \mathcal{P}(\mathcal{O}\mathcal{A}^{-1})$;
2.	faça $\mathcal{P} = \emptyset$;
3.	faça $i = 0$;
4.	**para** cada $w \in \mathcal{W}$
5.	calcule $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$;
6.	**se** $\exists\ C_i \in \mathcal{P}$ tal que $w' \in C_i$ e $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$
7.	faça $C_i = C_i \cup \{w\}$;
8.	**se não**
9.	faça $i = i + 1$;
10.	crie C_i em $\mathcal{P}$;
11.	faça $C_i = \{w\}$;
12.	**para** cada $(C_i, a) \in \mathcal{P} \times \mathcal{A}$
13.	**se** $\delta(C_i, a) \in \mathcal{P}$
14.	crie um ramo com rótulo a de C_i para $\delta(C_i, a)$;
15.	**se não**
16.	não crie um novo ramo;
17.	faça $G' = (\mathcal{V}', \mathcal{E}', \lambda')$ a componente essencial do grafo gerado nos passos 12-16;
18.	**saída** $G' = (\mathcal{V}', \mathcal{E}', \lambda')$;

Os passos 4-11 determinam a partição $\mathcal{P}$. Os passos 12-16 calculam a função de transição. Denotaremos por H o grafo dos contextos gerado pelas linhas 1-16 do algoritmo.

Tabela 3.2: Cálculo das classes de equivalência e da função $\delta(C_i, a)$, para $C_i \in \mathcal{P}$ e $a \in \mathcal{A}$.

Cálculo das Classes			$\delta(C_i, a)$ - {*Palavra mais longa em* $\mathcal{S}(wa) \cap \mathcal{W}$}		
$w \in \mathcal{W}$	$\{s \mid (\varepsilon, s) \in \tilde{\mathcal{C}}^r_{\mathcal{O}}(w)\}$	*Classes*	$\delta(C_i, 0) - \{v\}$	$\delta(C_i, 1) - \{v\}$	$\delta(C_i, 2) - \{v\}$
ε	$\emptyset$	C_1	$C_2 - \{0\}$	$C_3 - \{1\}$	$C_4 - \{2\}$
0	02	C_2	$C_5 - \{00\}$	$C_3 - \{1\}$	$C_4 - \{2\}$
1	00, 11, 12, 011, 012	C_3	$C_4 - \{10\}$	$C_6 - \{11\}$	$C_4 - \{2\}$
2	0, 11, 12	C_4	$\{\mathcal{O}\}$	$C_6 - \{21\}$	$C_4 - \{2\}$
00	2, 02	C_5	$C_5 - \{00\}$	$C_3 - \{1\}$	$\{\mathcal{O}\}$
10	0, 11, 12	C_4	$\{\mathcal{O}\}$	$C_6 - \{101\}$	$C_4 - \{2\}$
11	1, 2, 00, 011, 012	C_6	$C_4 - \{10\}$	$\{\mathcal{O}\}$	$\{\mathcal{O}\}$
21	1, 2, 00, 011, 012	C_6	$C_4 - \{10\}$	$\{\mathcal{O}\}$	$\{\mathcal{O}\}$
101	1, 2, 00, 011, 012	C_6	$C_4 - \{10\}$	$\{\mathcal{O}\}$	$\{\mathcal{O}\}$

contextos, portanto as componentes essenciais destes também são isomorfas.

Teorema 3.7. *O grafo H é isomorfo ao grafo dos contextos do SFT.*

Demonstração: Iniciamos definindo a função $\Phi : \mathcal{C} \to \mathcal{P}$ como $C_i = \Phi(\mathsf{F}(w))$, então C_i é a classe que contém a palavra mais longa em $\mathcal{S}(w) \cap \mathcal{W}$. Seguimos provando que Φ é uma função sobrejetiva, injetiva e é um homomorfismo, respectivamente.

Suponha que $\mathsf{F}(w) = \mathsf{F}(w')$. Dos Lemas 3.4 e 3.5, se v e v' são as palavras mais longas em $\mathcal{S}(w) \cap \mathcal{W}$ e $\mathcal{S}(w') \cap \mathcal{W}$, respectivamente, então $\mathsf{F}(v) = \mathsf{F}(v')$, implicando que v e v' pertencem à mesma classe de equivalência de $\mathcal{P}$, logo $\Phi(\mathsf{F}(w)) = \Phi(\mathsf{F}(w'))$. Concluímos que Φ é uma função bem definida.

Para todo $C_i \in \mathcal{P}$ e qualquer $w \in C_i$, temos que $w \in L$ e $\Phi(\mathsf{F}(w)) = C_i$, ou seja, Φ é sobrejetiva.

Suponha que $\Phi(\mathsf{F}(w)) = \Phi(\mathsf{F}(w'))$, isto implica que as palavras mais longas em $\mathcal{S}(w) \cap \mathcal{W}$ e $\mathcal{S}(w') \cap \mathcal{W}$ pertencem à mesma classe de equivalência. Do Lemma 3.5 isto implica que $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$ e do Teorema 3.3 temos que $\mathsf{F}(w) = \mathsf{F}(w')$. Concluímos que Φ é injetiva.

A classe $\delta(\Phi(\mathsf{F}(w)), a)$ contém a palavra mais longa v em $\mathcal{S}(ua) \cap \mathcal{W}$, onde u é a palavra mais longa em $\mathcal{S}(w) \cap \mathcal{W}$. Supondo que v' pertence a $\mathcal{S}(wa) \cap \mathcal{W}$. Se $v' \neq \varepsilon$, podemos escrever $v' = u'a$, onde $u' \in \mathcal{S}(w) \cap \mathcal{W}$, já que o prefixo de um elemento em $\mathcal{W}$ pertence a $\mathcal{W}$. Isto implica que u' é um sufixo de u, logo $v' = u'a \in \mathcal{S}(ua) \cap \mathcal{W}$. Como $\mathcal{S}(ua) \subseteq \mathcal{S}(wa)$ então $v \in \mathcal{S}(wa)$, logo v é a palavra mais longa em $\mathcal{S}(wa) \cap \mathcal{W}$ e, portanto, $\delta(\Phi(\mathsf{F}(w)), a) = \Phi(\mathsf{F}(wa))$. Concluímos que Φ é um homomorfismo. ■

Corolário 3.8. *O grafo G' é uma apresentação do SFT.*

Demonstração: Devido ao isomorfismo entre o grafo H e o grafo dos contextos, qualquer caminho bi-infinito ($\ldots e_{-1}e_0e_1\ldots$) em H está associado a um único caminho bi-infinito ($\ldots f_{-1}f_0$ $f_1\ldots$) no grafo dos contextos e vice-versa, tal que $\Phi(t(e_j)) = t(f_j)$, $\Phi^{-1}(t(f_j)) = t(e_j)$ e e_j, f_j possuem o mesmo rótulo. Estes caminhos também estão presentes nas componentes essenciais $G' = \{\mathcal{V}', \mathcal{E}', \lambda'\}$ e $G = \{\mathcal{V}, \mathcal{E}, \lambda\}$ de H e do grafo dos contextos, respectivamente, permitindo a definição do isomorfismo $\Psi : \mathcal{V} \to \mathcal{V}'$ entre G e G', tal que $\Psi = \Phi|\mathcal{V}$. ■

A Figura 3.1 mostra o grafo obtido diretamente da Tabela 3.2 pela conexão das classes de equivalência (vértices) de acordo com a função de transição δ. A Figura 3.2 mostra a componente essencial do grafo apresentado na Figura 3.1.

O próximo lema expõe uma propriedade de G' que nos permitirá provar no Teorema 3.11, que G' é o mínimo SDP, ou o Shannon cover para um SFT irredutível.

Lema 3.9. *Seja π um caminho em G' conectando os vértices $C_i = i(\pi)$ e $C_j = t(\pi)$, para o qual $\mathcal{L}(\pi) = z$. Para qualquer $w \in C_i$, a palavra mais longa em $\mathcal{S}(wz) \cap \mathcal{W}$ pertence a C_j.*

Demonstração: A prova é por indução no comprimento $|z|$. Inicialmente, seja $|z| = 0$, logo $z = \varepsilon$, portanto $wz = w\varepsilon = w$, que é a palavra mais longa em $\mathcal{S}(w\varepsilon) \cap \mathcal{W}$, concordando com a equação $\delta(C_i, \varepsilon) = C_i$.

Consideremos que a afirmação é verdadeira para $|z| = n$. Seja $\tau\alpha$ um caminho em G', $i(\tau\alpha) = C_i$ e $z = \mathcal{L}(\tau\alpha) = \mathcal{L}(\tau)\mathcal{L}(\alpha) = ua$, tal que $|u| = n$ e $a \in \mathcal{A}$. Se u' é a palavra mais longa em $\mathcal{S}(wu) \cap \mathcal{W}$, então concluímos da hipótese indutiva que $u' \in t(\tau)$. A partir da definição de δ, a palavra mais longa em $\mathcal{S}(u'a) \cap \mathcal{W}$ pertence a $\delta(t(\tau), a)$. Caso $\mathcal{S}(u'a) \cap \mathcal{W} = \{\varepsilon\}$ então $\mathcal{S}(wz) \cap \mathcal{W} = \{\varepsilon\}$, do contrário, se $x'a \in \mathcal{S}(wz) \cap \mathcal{W}$ então $x' \in \mathcal{S}(u')$ e, portanto, $x'a \in \mathcal{S}(u'a) \cap \mathcal{W}$, o que é uma contradição. Assim, o lema é válido para este caso.

Consideremos que a palavra mais longa em $\mathcal{S}(u'a) \cap \mathcal{W}$ seja não nula e igual a $q'a$. Agora, seja qa a palavra mais longa em $\mathcal{S}(wz) \cap \mathcal{W}$, então $q \in \mathcal{S}(wu) \cap \mathcal{W}$ (já que todo prefixo de uma palavra em $\mathcal{W}$ pertence a $\mathcal{W}$), isto implica que $q \in \mathcal{S}(u')$, logo $q \in \mathcal{S}(q')$. Por fim, como $u'a \in \mathcal{S}(wz)$, a palavra mais longa em $\mathcal{S}(u'a) \cap \mathcal{W}$ pertence a $\mathcal{S}(wz) \cap \mathcal{W}$, portanto $q' \in \mathcal{S}(q)$. Concluímos que $q'a = qa$. ■

Proposição 3.10. *$G' = (\mathcal{V}', \mathcal{E}', \mathcal{L}')$ é uma apresentação determinística, de sincronização e reduzida.*

Demonstração: Como o grafo dos contextos é determinístico, todo sub-grafo dele também é determinístico. Já que G' é isomorfo a G, então G' é determinístico.

Prosseguiremos provando que G' é de sincronização. Seja M o máximo comprimento de uma palavra em $\mathcal{W}$, logo para todo $w, u \in L$, $|u| \geq M$, as palavras mais longas em $\mathcal{S}(wu) \cap \mathcal{W}$ e $\mathcal{S}(u) \cap \mathcal{W}$ são iguais. Portanto, segue do Lema 3.9 que todo $z \in L$, $|z| \geq M$, z é uma palavra de sincronização

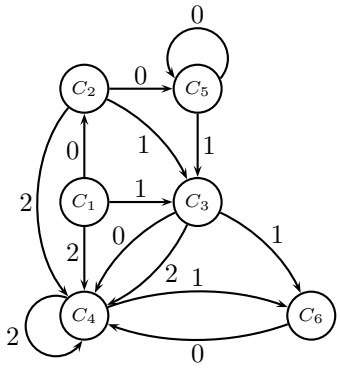


Figura 3.1: Grafo obtido da Tabela 3.2.

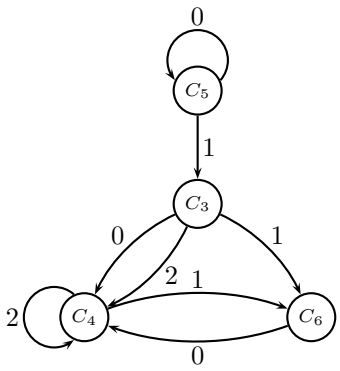


Figura 3.2: Componente essencial do grafo apresentado na Figura 3.1.

de G'. Suponha que há um vértice $C_i \in \mathcal{V}'$ que não possui uma palavra de sincronização, então todo caminho π em G', $t(\pi) = C_i$, tem comprimento menor que M. Considerando os caminhos em G' com vértice terminal C_i, seja π' um entre os mais longos destes caminhos, então não há ramos com vértice terminal $i(\pi')$. Como G' é uma grafo essencial, concluímos que há uma palavra de sincronização para todo vértice de G'.

Agora provaremos que G' é reduzido. Seja $z \in L$, $|z| \geq M$, uma palavra de sincronização para $C_i \in \mathcal{V}'$. Portanto, a palavra mais longa em $\mathcal{S}(z) \cap \mathcal{W}$ pertence a C_i. Segue do Lema 3.5, que $\tilde{\mathcal{C}}^r_{\mathcal{O}}(z) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(u)$ para todo $u \in C_i$, logo, do Teorema 3.3 obtemos que $\mathsf{F}(z) = \mathsf{F}(u)$. Como $\mathsf{F}(z) = \mathsf{F}(C_i)$, podemos concluir que $\mathsf{F}(C_i) = \mathsf{F}(u)$. Portanto, G' é reduzido, já que para todo $u \in C_i$ e $u' \in C_j$, $\mathsf{F}(u) \neq \mathsf{F}(u')$ se, e somente se, $i \neq j$. ■

Teorema 3.11. *$G' = (\mathcal{V}', \mathcal{E}', \mathcal{L}')$ é o Shannon cover de um SFT irredutível e o mínimo SDP de um SFT redutível.*

Demonstração: Da Proposição 2.2 todas as SDPs reduzidas de um SSR são isomorfas, portanto, segue da Proposição 3.10 que G' é a mínima SDP. Como já havíamos definido, o Shannon cover é a única (por um mapeamento isomorfo) apresentação determinística, reduzida e irredutível de um SSR irredutível, de [10, Teorema 3.4.17] este também é de sincronização. Logo, G' é o Shannon cover de um SFT irredutível. ■

3.4 Análise da complexidade do algoritmo

Inicialmente analisaremos a complexidade para obter-se o conjunto $\mathcal{P}$ dado o conjunto $\mathcal{W}$, o que equivale à especificação dos vértices. Em seguida, analisaremos a complexidade do cálculo da função

de transição.

3.4.1 Complexidade do Cálculo do Conjunto $\mathcal{P}$

Para determinarmos $\mathcal{P}$ dado o conjunto $\mathcal{W}$, utilizaremos o Teorema 3.3 para determinarmos as palavras em $\mathcal{W}$ com mesmo contexto à direita. O cálculo de $\mathcal{P}$ será dividido em três etapas: (1) cálculo dos conjuntos $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w))$ para todo $w \in \mathcal{W}$; (2) cálculo dos conjuntos $\mathcal{S}(\tilde{\mathcal{C}}^r_{\mathcal{O}}(w))$ para todo $w \in \mathcal{W}$; (3) determinar a partição do conjunto $\mathcal{W}$ (pela relação de equivalência: $w \sim w'$ se, e somente se $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w) = \tilde{\mathcal{C}}^r_{\mathcal{O}}(w')$).

Complexidade do cálculo dos conjuntos $\mathcal{C}^r_{\mathcal{O}}(w)$

Consideremos $a_i \in \mathcal{A}$ e $a_1 a_2 \dots a_n \in \mathcal{W}$. Do cálculo recursivo apresentado na Equação 3.1, $|\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1))|$ é proporcional a $|\mathcal{O}|$, $|\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1 a_2))| \leq |a_2^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1))| + |a_2^{-1}\mathcal{O}|$ é proporcional a $2 \cdot |\mathcal{O}|$, seguindo este raciocínio, $|\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1 \dots a_n))| \leq |a_n^{-1}\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(a_1 \dots a_{n-1}))| + |a_n^{-1}\mathcal{O}|$ é proporcional a $n \cdot |\mathcal{O}|$, sendo que $n \leq M$, onde $M+1$ é maior comprimento das palavras em $\mathcal{O}$. Assim, a cardinalidade dos conjuntos $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w))$, conseqüentemente dos conjuntos $\mathcal{C}^r_{\mathcal{O}}(w)$, é proporcional a $|\mathcal{O}|$, com coeficiente de proporcionalidade não superior a M. Portanto, a complexidade assintótica para o cálculo de $\mathcal{C}^r_{\mathcal{O}}(w_1 \dots w_n)$, dado $\mathcal{C}^r_{\mathcal{O}}(w_1 \dots w_{n-1})$, é $\mathcal{O}(|\mathcal{O}|)$.

Pela definição do conjunto $\mathcal{W}$ obtemos a relação $\mathcal{P}(\mathcal{W}) = \mathcal{W}$, assim, aplicaremos a expressão recursiva (3.1) para determinarmos o conjunto $\{\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w)) \mid w \in \mathcal{W}\}$. O que implica em calcularmos $M-1$ conjuntos $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w))$ (já que $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(\varepsilon)) = \emptyset$). Portanto, a complexidade assintótica para o cálculo do conjunto $\{\mathcal{C}^r_{\mathcal{O}}(w) \mid w \in \mathcal{W}\}$ é $\mathcal{O}(|\mathcal{O}| \cdot |\mathcal{W}|)$.

Complexidade do cálculo dos conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$

A determinação de $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ a partir de $\mathcal{C}^r_{\mathcal{O}}(w)$ pode ser realizada usando-se uma árvore enraizada rotulada, de construção especificada a seguir. Dos nós não-terminais da árvore, partem no máximo $|\mathcal{A}|$ ramos com rótulos em $\mathcal{A}$, dispostos em ordem lexicográfica definida pelos rótulos. O rótulo de um caminho começando no nó raiz da árvore é a seqüência dos rótulos dos ramos que constituem o caminho. A partir das seqüências de menor comprimento, marca-se como um nó terminal o nó alcançado por um caminho com rótulo em $\mathcal{C}^r_{\mathcal{O}}(w)$. Os elementos em $\mathcal{S}(\tilde{\mathcal{C}}^r_{\mathcal{O}}(w))$ são associados a nós terminais (folhas) da árvore. Seqüências em $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w)) \backslash \mathcal{S}(\tilde{\mathcal{C}}^r_{\mathcal{O}}(w))$ não podem ser associadas a caminhos na árvore.

Como exemplo, consideremos o conjunto $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(101)) = \{1, 2, 00, 011, 012, 11, 12\}$ apresentado no Exemplo 3.4. A árvore associada a este é mostrada na Figura 3.3. Ao percorrermos a árvore

utilizando o elemento 11 ou 12, iremos alcançar a folha 1, o que indica que estes elementos possuem um prefixo próprio no conjunto $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(101))$, portanto as folhas 11 e 12 não são criadas, o que é representado na Figura 3.3 pelas linhas pontilhadas.

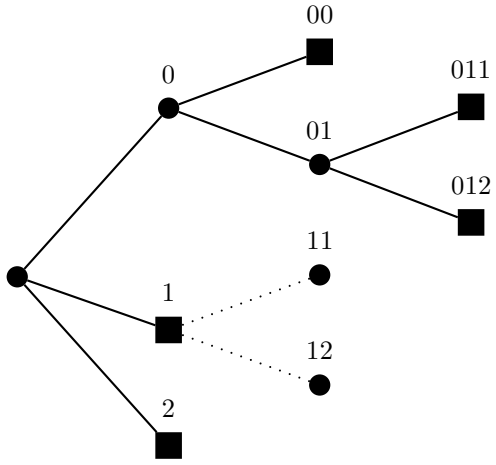


Figura 3.3: Árvore para o cálculo do conjunto $\tilde{\mathcal{C}}^r_{\mathcal{O}}(101)$.

Dado um conjunto $\mathcal{S}(\mathcal{C}^r_{\mathcal{O}}(w))$, $w \in \mathcal{W}$, a complexidade deste procedimento é proporcional ao número de caminhos que iremos percorrer na árvore, ou seja, $\mathcal{O}(|\mathcal{O}|)$. Como este cálculo será realizado para todos os elementos do conjunto $\mathcal{W}$, a complexidade desta etapa é $\mathcal{O}(|\mathcal{O}| \cdot |\mathcal{W}|)$.

Complexidade do cálculo das classes de equivalência

Para determinarmos os conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ que são iguais, podemos, seguindo a mesma ordem lexicográfica, sobrepor as árvores dos conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ e indicar nos nós os conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ para os quais estes nós são folhas. Este processo pode ser realizado simultaneamente com o cálculo dos conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$. Os conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$ iguais são os que ocorrem nos mesmos nós. A Figura 3.4 mostra a árvore associada ao Exemplo 3.6.

As classes de conjuntos iguais são as mesmas calculadas na Tabela 3.2. A complexidade do algoritmo é proporcional ao número de nós e a cardinalidade do conjunto $\mathcal{W}$, ou seja, $\mathcal{O}(|\mathcal{O}| \cdot |\mathcal{W}|)$.

3.4.2 Complexidade do Cálculo da Função de Transição

Para a determinação da função de transição, δ, empregamos uma variação da *função de falha* definida em [24], que chamaremos de f e a definiremos a seguir.

Definição 3.5. Seja C_ε a classe em $\mathcal{P}$ que contém a palavra ε. Definimos a função $f : \mathcal{P}\backslash\{C_\varepsilon\} \longrightarrow \mathcal{P}$, tal que, dado $v = au$, $a \in \mathcal{A}$, uma das palavras de menor comprimento em C_i, então $f(C_i)$ é o conjunto em $\mathcal{P}$ que contém o mais longo sufixo de u em $\mathcal{W}$.

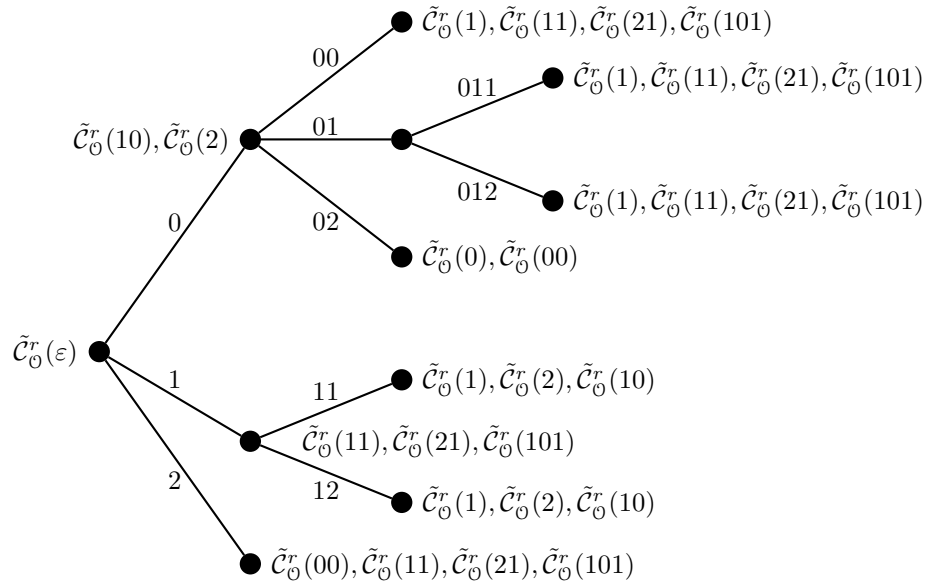


Figura 3.4: Árvore para o cálculo das classes de equivalência.

Dado o conjunto $\mathcal{P}$, a função de transição δ será calculada primeiramente para as classes em $\mathcal{P}$ que contêm as menores palavras em $\mathcal{W}$. Assim, começamos pela classe que contém a palavra ε, em seguida calculamos δ para as classes que contêm as palavras de comprimento um, e assim sucessivamente. Finalizamos quando a função δ tiver sido calculada para todas as classes do conjunto $\mathcal{P}$.

O algoritmo apresentado na Tabela 3.3 calcula a função δ para todo $C_i \in \mathcal{P}$ e $a \in \mathcal{A}$. Seguiremos definindo algumas notações que são empregadas neste algoritmo. Para todo $v \in \mathcal{W} \cup \mathcal{O}$, chamamos de C_v a classe em $\mathcal{P} \cup \{\mathcal{O}\}$ que contém v. Dado um conjunto de palavras $\mathcal{A}$, chamaremos de $\mathcal{A}^{(i)}$ o subconjunto de palavras em $\mathcal{A}$ de comprimento i.

No algoritmo da Tabela 3.3 o conjunto $\mathcal{O}$ pertence a imagem da função de transição δ, o que não ocorre para o algoritmo na Tabela 3.1. Isto se deve ao uso da função de falha, mas não acarreta em acréscimo da complexidade, pois o vértice associado ao conjunto $\mathcal{O}$ não possui ramos de saída, portanto, dispensando qualquer verificação, podemos excluí-lo ao final do algoritmo, juntamente com os ramos que incidem neste.

Com o conhecimento das classes de equivalência e indexação destas através de seus elementos de menor comprimento, as operações das linhas 7,10,11 e 21 possuem complexidade $\mathcal{O}(|\mathcal{P}|)$. O cálculo da função de transição, δ, para qualquer $C_i \in \mathcal{P}$ tem complexidade assintótica $\mathcal{O}(|\mathcal{A}|)$. Assim, o cálculo de δ para todos os elementos de $\mathcal{P}$ tem complexidade assintótica $\mathcal{O}(|\mathcal{A}| \cdot |\mathcal{P}|)$. Portanto, a etapa de cálculo da função de transição, quando realizada pelo algoritmo apresentado na Tabela 3.3,

Tabela 3.3: Cálculo da função de transição δ.

PASSO	INSTRUCÃO
1.	**para cada** $a \in \mathcal{A}$
2.	**se** $a \in \mathcal{W}$
3.	faça $\delta(C_\varepsilon, a) = C_a$;
4.	faça $f(C_a) = C_\varepsilon$;
5.	**se não**
6.	faça $\delta(C_\varepsilon, a) = C_\varepsilon$;
7.	faça $P_1 = \mathcal{W} \backslash C_\varepsilon$;
8.	**enquanto** $P_1 \neq \emptyset$
9.	$i = \min\{\|w\| : \ w \in P_1\}$;
10.	faça $P_2 = P_1^{(i)}$;
11.	faça $P_1 = P_1 \backslash \bigcup_{v \in P_2} C_v$;
12.	**enquanto** $P_2 \neq \emptyset$
13.	seja $v \in P_2$;
14.	**para cada** $a \in \mathcal{A}$
15.	**se** $va \in \mathcal{W} \cup \mathcal{O}$
16.	faça $\delta(C_v, a) = C_{va}$;
17.	**se** $va \in \mathcal{W}$
18.	faça $f(C_{va}) = \delta(f(C_v), a)$;
19.	**se não**
20.	faça $\delta(C_v, a) = \delta(f(C_v), a)$;
21.	faça $P_2 = P_2 \backslash C_v$;

tem complexidade assintótica $\mathcal{O}(|\mathcal{A}| \cdot |\mathcal{P}|)$.

3.4.3 Complexidade do Algoritmo para Determinação da m-SDP

Empregamos duas etapas para determinação da m-SDP através da técnica apresentada. Na primeira, determinamos $\mathcal{P}$ por meio de um algoritmo de complexidade assintótica $\mathcal{O}(|\mathcal{O}| \cdot |\mathcal{W}|)$. Na segunda, calculamos a função de transição δ, utilizamos para isto um algoritmo de complexidade assintótica $\mathcal{O}(|\mathcal{A}| \cdot |\mathcal{P}|)$. Concluímos que as duas etapas podem ser realizadas por um algoritmo de complexidade $\mathcal{O}(|\mathcal{O}| \cdot |\mathcal{W}|) + \mathcal{O}(|\mathcal{A}| \cdot |\mathcal{P}|)$.

CAPÍTULO 4

DETERMINAÇÃO DO CONTEXTO À DIREITA PARA PFT IRREDUTÍVEL: APLICAÇÃO NA CONSTRUÇÃO DE APRESENTAÇÕES DETERMINÍSTICAS E REDUZIDAS

Os códigos restritivos são comumente empregados em sistemas que utilizam equalização com resposta parcial e detecção por seqüência, sendo usualmente chamados em sistemas de gravação magnética de *códigos restritivos para aumento de distância.* O seu emprego visa aumentar a distância mínima na saída do canal com interferência inter-simbólica pela proibição de um número finito de seqüências binárias. Como comentado no Capítulo 2, o estudo e projeto destes codificadores pode ser realizado aplicando-se a teoria de dinâmica simbólica. Lembrando que, para o caso em que o conjunto das seqüências proibidas é finito, os sistemas simbólicos fechados são sistemas simbólicos regulares de memória finita (SFT).

As restrições de um SFT são globais, ou seja, estas independem da coordenada na seqüência bi-infinita. No entanto, restrições têm sido apresentadas para as quais as proibições ocorrem de forma periódica, sendo chamadas de *restrições periódicas, e.g.*, a restrição TMTR (do inglês *time-varying maximum-transition-run*) proíbe que a palavra 111 seja iniciada em índices ímpares de uma seqüência. Os sistemas simbólicos dinâmicos periódicos ou PFT (do inglês *periodic shift of finite type*) foram introduzidos em [25] como a classe de sistemas simbólicos fechados adequada para modelagem destas restrições (se for possível descrevê-las por um conjunto finito), sendo um PFT

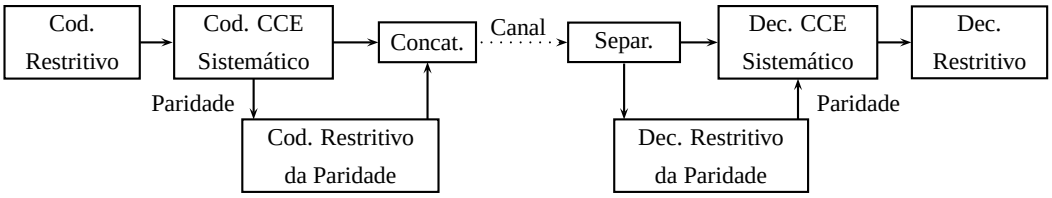


Figura 4.1: Diagrama de concatenação reversa.

especificado por um conjunto finito de palavras proibidas indexadas (índices representando fases) e um período T.

Restrições periódicas também aparecem em aplicações que empregam codificação conjunta CCE/ restritiva. No Capítulo 1 mencionamos a possibilidade de melhorar o desempenho de um sistema de comunicações adaptando-se a seqüência transmitida às características do canal. Isto pode ser obtido pelo projeto de códigos que tanto possam corrigir como restringir a seqüência transmitida, no entanto, na prática cascateia-se os codificadores. Apresentamos na Figura 1.1 um arranjo padrão para os codificadores. A desvantagem deste arranjo é a possibilidade de propagação de erros pelo decodificador restritivo antes que estes sejam corrigidos pelo decodificador CCE. Como alternativa há a *concatenação reversa*, mostrada na Figura 4.1, onde a posição dos codificadores é invertida, neste caso algumas precauções devem ser seguidas para evitar que as restrições impostas às seqüências não sejam violadas. Isto é obtido pelo uso de um *codificador CCE sistemático* e um *codificador restritivo da paridade* com um algoritmo de decodificação que limite a propagação de erros. A saída do codificador restritivo da paridade é concatenada com a saída do *codificador CCE sistemático* pelo *concatenador*, com a possível inserção de bits entre as seqüências de saída, possibilitando que a seqüência resultante satisfaça a restrição. Na decodificação, o *separador* envia os bits de informação para o *decodificador CCE sistemático* e os bits de paridade codificados para o *decodificador restritivo da paridade* antes de irem para o decodificador CCE sistemático. Este processo possibilita a correção de parte dos erros gerados pelo canal antes que a seqüência recebida passe pelo decodificador restritivo, o que reduz a propagação de erros na decodificação restritiva. Quando o codificador restritivo tiver alta taxa, isto será mantido por este sistema, já que a seqüência de paridade é curta quando comparada com a seqüência de informação.

Consideremos que não seja empregada a métrica de Hamming na decodificação (*soft-decision decoding*). A informação para decodificação associada aos bits de informação é diretamente disponi-

bilizada para o decodificador CCE, já que o codificador CCE é sistemático e só os bits de paridade passam por um codificador restritivo antes de serem enviados ao canal. Neste caso, a informação para decodificação contida na seqüência de paridade gerada pelo codificador CCE terá que ser extraída de uma seqüência codificada pelo codificador restritivo da paridade e com possíveis erros gerados pelo canal. Este procedimento pode ser simplificado e realizado com maior precisão se o codificador restritivo da paridade for sistemático. Isto poderia reduzir a taxa do codificador restritivo da paridade, o que só estaria relacionado à pequena porção da seqüência codificada relativa aos bits de paridade, tendo pouca representatividade na taxa do sistema completo.

Relativo a um codificador restritivo da paridade sistemático, Wijngoorden e Immink sugeriram o arranjo apresentado na Figura 4.2 [26]. A idéia proposta é o codificador restritivo gerar seqüências com posições não restritas, ou seja, nestas posições o bit pode assumir arbitrariamente o valor 0 ou 1 sem violar a restrição, portanto os bits de paridade gerados pelo codificador CCE sistemático podem ser inseridos nestas posições (realizado pelo bloco *Insere* da Figura 4.2) sem violar a restrição. O sistema obtido pelo conjunto de todas as seqüências bi-infinitas com *posições irrestritas* geradas pelo arranjo proposto por Wijngoorden e Immink forma um PFT [3].

Uma das possibilidades no projeto de um sistema que utilize codificação conjunta, como o representado na Figura 4.2, é a construção do maior subsistema de um SFT S, tal que toda posição módulo T em U é irrestrita, sendo U um subconjunto de T. Este sistema é chamado de *subsistema* (U,T)*-irrestrito* de S. Conhecendo-se este subsistema, podemos calcular a entropia topológica do conjunto de seqüências que satisfazem uma dada restrição e são irrestritas em um conjunto de posições. Aplicando-se os procedimentos para construção de códigos de linha, podemos projetar codificadores para este subsistema; como estes procedimentos atuam em uma apresentação do subsistema, deve-se procurar apresentações com número mínimo de vértices.

Neste capítulo apresentaremos conceitos que serão empregados para determinar uma apresentação determinística e reduzida de um PFT irredutível. Isto será possível pois a linguagem de um

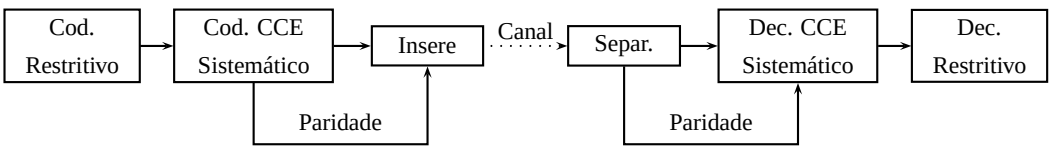


Figura 4.2: Diagrama de concatenação Wijngaarden-Immink.

PFT é regular [25]. Para isto, iremos generalizar a idéia apresentada no Capítulo 3, para uma coleção mínima periódica de palavras proibidas e para construção do grafo a partir de classes obtidas de uma lista indexada de palavras da linguagem.

4.1 Alguns Conceitos e Definições

Os PFT são sistemas simbólicos dinâmicos fechados com restrições variantes no tempo, sendo os sistemas dinâmicos simbólicos adequados para representar conjunto de seqüências que proíbem a ocorrência de palavras em coordenadas periódicas da seqüência módulo um período T.

Determinado um inteiro positivo T, seja $\mathcal{F}$ uma coleção finita de palavras em $\mathcal{A}^*$, tal que, para cada $w_i \in \mathcal{F}$ é associado um inteiro k_i em $\{0, 1, \ldots, T-1\}$ que recebe o nome de *fase.* Descrevemos o conjunto $\mathcal{F}$ por $\mathcal{F} = \{(w_1, k_1), (w_2, k_2), \ldots, (w_n, k_n)\}$ e o chamamos de *conjunto proibido periódico.* As restrições associadas a uma fase k são escritas por $\mathcal{F}^{(k)} = \{w|\ (w, n) \in \mathcal{F}$ e $n = k\}$. Seja uma palavra finita w um fator de uma seqüência finita ou infinita x na coordenada i, portanto $w = x_{[i,i+|w|-1]}$, isto é representado por $w \prec_i x$, o contrário é representado por $w \not\prec_i x$. Escrevemos $w \prec x$ quando uma palavra w é um fator de uma seqüência x, sem considerar uma coordenada específica, o contrário é representado por $w \not\prec x$. Uma palavra w é um fator próprio de uma seqüência x, quando $w \prec x$ e $w \neq x$. Um PFT com período T e conjunto proibido periódico $\mathcal{F}$, representado por $\mathsf{X}_{\{\mathcal{F},T\}}$, é definido a seguir.

Definição 4.1. Uma seqüência bi-infinita x pertence a $\mathsf{X}_{\{\mathcal{F},T\}}$ se, e somente se, existe um inteiro $k \in \{0, \ldots, T-1\}$ tal que, para todo inteiro i, se $w \prec_i \sigma^k(x)$ então $w \notin \mathcal{F}^{(i \mod T)}$.

Para todo PFT é possível construir um grafo $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \lambda)$ que o apresente, ou seja, um PFT é um SSR [25, Teorema 1]. $\mathcal{G}$ é dito *T-partite* se $\mathcal{V}$ pode ser dividido em T subconjuntos $D_0,\ D_1, \ldots,$ D_{T-1}, tal que, qualquer ramo que parte de um vértice em D_i alcança um vértice em $D_{i+1 \mod T}$. Para $I, J \in \mathcal{V}$, o número de ramos que partem de I e alcançam J é A_{IJ}. A *matriz adjacência* de $\mathcal{G}$ é quadrada, de ordem $|\mathcal{V}|$ e é denotada por $A_{\mathcal{G}} = \left[A_{IJ}\right]$. O *período de um vértice* I, representado por $\mathsf{per}(I)$, é o máximo divisor comum dos inteiros n para os quais $(A^n_{\mathcal{G}})_{II} > 0$. Caso este número não exista, então denota-se $\mathsf{per}(I) = \infty$. O *período de uma matriz* A, representado por $\mathsf{per}(A)$, é o máximo divisor comum dos números $\mathsf{per}(I)$ que são finitos, ou é ∞ se $\mathsf{per}(I) = \infty$ para todo I. O *período de um grafo* $\mathcal{G}$, representado por $\mathsf{per}(\mathcal{G})$, é o período de $A_{\mathcal{G}}$, *i.e*, $\mathsf{per}(\mathcal{G}) \triangleq \mathsf{per}(A_{\mathcal{G}})$. Se $\mathcal{G}$ é irredutível, todos os estados têm o mesmo período [10, Lema 4.5.3], se $\mathsf{per}(A_{\mathcal{G}}) = T$, então $\mathcal{G}$ é *T-partite* e os conjuntos $D_1,\ D_2, \ldots,\ D_{T-1}$ são as T *classes periódicas* do grafo.

Se o PFT é irredutível, podemos determinar seu Shannon cover aplicando métodos convencionais para construção de grafos determinísticos a partir de conjuntos de palavras proibidas seguidos de algoritmos de minimização [25], [3]. Em nossa abordagem, sempre estaremos considerando o período como sendo o do Shannon cover, portanto, lembramos que parte dos resultados só são válidos para PFT irredutíveis, o que deixaremos claro pelo enunciado destes.

Associado ao conceito de restrição por fase está o de linguagem por fase, que empregaremos para descrever as restrições de palavras em $\mathcal{A}^*$ para uma fase específica, como também na obtenção de propriedades de apresentações do PFT.

Definição 4.2. Seja $\mathfrak{L}_x \subseteq \{0, \ldots, T-1\}$ o conjunto de deslocamentos de um ponto $x \in \mathcal{A}^{\mathbb{Z}}$ tal que $\forall i \in \mathbb{Z}$ e $\ell_x \in \mathfrak{L}_x$ se $v \prec_i \sigma^{\ell_x}(x)$ então $v \notin \mathcal{F}^{(i \mod T)}$. Definimos uma linguagem associada com a fase k como $L^{(k)} = \{u \prec_j \sigma^{\ell_x}(x) |\ x \in \mathsf{X}_{\{\mathcal{F},T\}},\ \ell_x \in \mathfrak{L}_x \text{ e } j \equiv k \mod T\}$.

Um conceito similar ao de coleção mínima de palavras proibidas de um SFT existe para um PFT, sendo chamado de *conjunto mínimo periódico de palavras proibidas*, contendo os elementos $(w,k) \in \mathcal{A}^* \times \{0, \ldots, T-1\}$, $w = w_0 w_1 \ldots w_n$, que não pertencem a linguagem $L^{(k)}$ mas com todos os fatores próprios $u \prec_i w$ pertencendo a linguagem $L^{(k+i \mod T)}$. Portanto, $(w,k) \in \mathcal{O}$ se, e somente se, $w \notin L^{(k)}$, $w_{[0,|w|-2]} \in L^{(k)}$ e $w_{[1,|w|-1]} \in L^{(k+1 \mod T)}$. Em [25, Teorema 4] é demonstrado que $\mathcal{O}$ é único (sendo T o período do Shannon cover).

Exemplo 4.1. Como exemplo da dependência da linguagem com a fase, consideremos $\mathcal{O} = \{(101, 0), (111, 1)\}$ e $T = 2$. Observe que $11100 \in L^{(0)}$ e $1010 \in L^{(1)}$, no entanto, $1010 \notin L^{(0)}$ e $11100 \notin L^{(1)}$.

Motivados pela Definição 4.2, quando queremos enfatizar que $w \in L^{(k)}$ escrevemos (w,k). Se (w,k) não é definido, logo $w \notin L^{(k)}$, implicando que os elementos em $\mathcal{O}^{(k)}$ proíbem a ocorrência de w, o que não impede que $w \in L$, pois esta poderá pertencer a uma linguagem $L^{(j)}$ com $j \neq k$. Como uma extensão dessa definição, dizemos que (s,j) é um fator de (w,k) se $s \prec_t w$ e $j \equiv k + t \mod T$. Se $s \prec_0 w$ então (s,k) é um prefixo de (w,k) (prefixo próprio se $|s| < |w|$). Quando $s \prec_t w$ e $k + t + |s| \equiv k + |w| \mod T$ então $(s, k + t \mod T)$ é um sufixo de (w,k) (sufixo próprio se $t > 0$). Na próxima definição estenderemos o conceito de contexto à direita de uma palavra $w \in \mathcal{A}^*$ para um elemento (w,k).

Definição 4.3. O contexto à direita de $w \in L$ associado a uma fase k é $\mathsf{F}(w,k) = \{s |\ ws \in L^{(k)}\}$. O complemento do contexto à direita em relação a linguagem L é dado por $\mathsf{F}'(w,k) = L \backslash \mathsf{F}(w,k)$, ou $\mathsf{F}'(w,k) = \{s \in L |\ ws \notin L^{(k)}\}$.

Exemplo 4.2. Como exemplo do contexto à direita de uma palavra associado a uma fase, consideremos o conjunto $\mathcal{O}$ e período do Exemplo 4.1. Para $w = 1$, temos que $11 \in \mathsf{F}(1,0)$ e $01 \in \mathsf{F}(1,1)$, no entanto, $01 \notin \mathsf{F}(1,0)$ e $11 \notin \mathsf{F}(1,1)$.

Caso (w,k) não seja definido, então para todo $s \in L$ teremos que $ws \notin L^{(k)}$. Portanto, para todo $w \notin L^{(k)}$, segue da Definição 4.3 que $\mathsf{F}(w,k) = \emptyset$, implicando que $\mathsf{F}'(w,k) = L$. Prosseguindo com o paralelismo com relação aos conceitos apresentados no Capítulo 3, a próxima definição apresenta o *conjunto das restrições* para o caso periódico.

Definição 4.4. Seja $(w,k) \in L \times \{0,\ldots,T-1\}$. Um conjunto $\mathcal{C}(w,k) \subseteq L \times \mathbb{N} \cup \{0\}$, chamado conjunto das restrições de (w,k), satisfaz as propriedades:

a. Seja $(u,j) \in \mathcal{C}(w,k)$, para todo $s \in L$ se $u \prec_j s$ então $ws \notin L^{(k)}$;

b. Seja $s \in L$ e $ws \notin L^{(k)}$, então existe $(u,j) \in \mathcal{C}(w,k)$ tal que $u \prec_j s$.

Quando $w \notin L^{(k)}$, definimos $\mathcal{C}(w,k) = (\varepsilon, 0)$.

Definimos a seguir um operador que quando aplicado ao conjunto das restrições de um elemento (w,k) determina o contexto à direita da palavra w, o que será demonstrado na proposição seguinte.

Definição 4.5. Seja $\mathcal{C}(w,k)$ o conjunto das restrições de $(w,k) \in L \times \{0,\ldots,T-1\}$. Se $w \in L^{(k)}$ então $[\mathcal{C}(w,k)] = \{s \in L |\ \exists\, (u,j) \in \mathcal{C}(w,k)$ tal que $u \prec_j s\}$, caso contrário $[\mathcal{C}(w,k)] = L$.

Proposição 4.1. *Seja L a linguagem de um PFT. Se $w \in L$ então $[\mathcal{C}(w,k)] = \mathsf{F}'(w,k)$. Portanto, $[\mathcal{C}(w,k)] = [\mathcal{C}(w',j)]$ se, e somente se, $\mathsf{F}(w,k) = \mathsf{F}(w',j)$.*

Demonstração: Se $w \notin L^{(k)}$ então $ws \notin L^{(k)}$ para todo $s \in L$, logo $\mathsf{F}'(w,k) = L$. Da Definição 4.5, se $w \notin L^{(k)}$ então $[\mathcal{C}(w,k)] = L$. A seguir consideramos que $w \in L$.

Se $s \in \mathsf{F}'(w,k)$ então $ws \notin L^{(k)}$, logo da Definição 4.4-b existe $(u,j) \in \mathcal{C}(w,k)$ tal que $u \prec_j s$ e, portanto, $s \in [\mathcal{C}(w,k)]$, conseqüentemente $\mathsf{F}'(w,k) \subseteq [\mathcal{C}(w,k)]$. De modo reverso, se $s \in [\mathcal{C}(w,k)]$, então pela Definição 4.5 existe $(u,j) \in \mathcal{C}(w,k)$ tal que $u \prec_j s$. Da Definição 4.4-a $ws \notin L^{(k)}$ e, portanto, $s \in \mathsf{F}'(w,k)$, logo $[\mathcal{C}(w,k)] \subseteq \mathsf{F}'(w,k)$.

Se $[\mathcal{C}(w,k)] = [\mathcal{C}(w',j)]$ então $\mathsf{F}'(w,k) = \mathsf{F}'(w',j)$, logo da Definição 4.3 concluímos que $\mathsf{F}(w,k) = \mathsf{F}(w',j)$. Se agora considerarmos que $\mathsf{F}(w,k) = \mathsf{F}(w',j)$, então $\mathsf{F}'(w,k) = \mathsf{F}'(w',j)$ e, portanto, $[\mathcal{C}(w,k)] = [\mathcal{C}(w',j)]$. ■

4.2 Determinação de palavras com mesmo contexto à direita

Vimos no Capítulo 3 como determinar palavras de mesmo contexto à direita empregando o conjunto $\mathcal{O}$ do SFT. Para este caso, as restrições independem da coordenada, implicando que uma seqüência com um fator em $\mathcal{O}$ não pertence a $\mathsf{F}(w)$ para qualquer w em $\mathcal{B}(\mathsf{X}_{\mathcal{O}})$. No caso de um PFT as restrições estão associadas a fases, portanto, geralmente dependem da fase escolhida como referência, *e.g.*, se a partir da fase 0 qualquer palavra w com $v \prec_i w$ e $v \in \mathcal{O}^{(i)}$ não pode ocorrer, em relação a fase j as palavras u com $v \prec_{(i-j \mod T)} u$ não poderão ocorrer. Logo, com relação a fase j o conjunto das restrições considerado é $\{(v, i-j \mod T) | \ (v,i) \in \mathcal{O}\}$.

A dependência das restrições com a fase torna a definição do contexto à direita de uma palavra w dependente não só desta e do conjunto $\mathcal{O}$, como também da fase. Assim, para representarmos as restrições de uma palavra w com relação a uma fase k, iremos dividir $\mathcal{C}(w,k)$ em dois subconjuntos, enquanto um depende dos sufixos de w e da fase k, o outro depende do comprimento de w e da fase k.

Definição 4.6. Seja $\mathcal{A} \subseteq L \times \{0 \ldots T-1\}$. Representamos a expansão do conjunto $\mathcal{A}$ para um período T por $< \mathcal{A} >= \{(s,j) | \ (s, j \mod T) = (s,i) \text{ para todo } (s,i) \in \mathcal{A} \text{ e } j \geq 0\}$.

Definição 4.7. Seja w uma palavra da linguagem de um PFT. Se $w \in L^{(k)}$ então $\mathcal{C}^d_{\mathcal{O}}(w,k) = \{(s,0) | \ s \in L^{(k+|w| \mod T)}$ e $\exists\, p \in \mathcal{S}(w) \backslash \{\varepsilon\}$ para o qual $ps \in \mathcal{O}^{(k+|w|-|p| \mod T)}\}$ e $\mathcal{C}^i_{\mathcal{O}}(w,k) = \{(s,j) | \ (s,j) = (s, i-k-|w| \mod T)$ para algum $(s,i) \in \mathcal{O}\}$. Se $w \notin L^{(k)}$ definimos $\mathcal{C}^d_{\mathcal{O}}(w,k) = \mathcal{C}^i_{\mathcal{O}}(w,k) = (\varepsilon, 0)$. Definimos $\mathcal{C}_{\mathcal{O}}(w,k) = \mathcal{C}^d_{\mathcal{O}}(w,k) \cup < \mathcal{C}^i_{\mathcal{O}}(w,k) >$ como o conjunto de todas as restrições indexadas de (w,k).

Lema 4.2. *Se $\mathcal{C}_{\mathcal{O}}(w,k) = \mathcal{C}_{\mathcal{O}}(u,j)$ então $\mathcal{C}^d_{\mathcal{O}}(w,k) = \mathcal{C}^d_{\mathcal{O}}(u,j)$ e $\mathcal{C}^i_{\mathcal{O}}(w,k) = \mathcal{C}^i_{\mathcal{O}}(u,j)$.*

Demonstração: Suponha que existe $(s,0) \in \mathcal{C}^d_{\mathcal{O}}(w,k)$, tal que $(s,0) \in \mathcal{C}^i_{\mathcal{O}}(u,j)$. Como $(s,0) \in \mathcal{C}^d_{\mathcal{O}}(w,k)$, então $ps \in \mathcal{O}^{(k+|w|-|p| \mod T)}$, onde $p \in \mathcal{S}(w) \backslash \{\varepsilon\}$. Assim, $(ps, -|p| \mod T) \in \mathcal{C}^i_{\mathcal{O}}(w,k)$, portanto, temos que $(ps, -|p| \mod T) \in \mathcal{C}^i_{\mathcal{O}}(u,j)$ e por suposição que $(s,0) \in \mathcal{C}^i_{\mathcal{O}}(u,j)$, por conseqüência $s \in \mathcal{O}^{(|u|+j \mod T)}$ é um fator de $ps \in \mathcal{O}^{(|u|+j-|p| \mod T)}$, mas, como todo fator próprio de $(ps, |u|+j-|p| \mod T)$ pertence a linguagem, isto é uma contradição.

De forma análoga, se supusermos que $(s,0) \in \mathcal{C}^d_{\mathcal{O}}(u,j)$ e $(s,0) \in \mathcal{C}^i_{\mathcal{O}}(w,k)$, concluiremos que isto é uma contradição. Portanto, se $\mathcal{C}_{\mathcal{O}}(w,k) = \mathcal{C}_{\mathcal{O}}(u,j)$, então $(s,0) \in \mathcal{C}^d_{\mathcal{O}}(w,k)$ se, e somente se $(s,0) \in \mathcal{C}^d_{\mathcal{O}}(u,j)$. ■

Exemplo 4.3. Considere um PFT dado pelo conjunto $\mathcal{O} = \{(11,0),(1,2),(01,3)\}$ e com período

$T = 5$. Então para $(w,k) = (1,0)$ teremos $\mathcal{C}_{\mathcal{O}}^{d}(1,0) = \{(1,0)\}$ e $\mathcal{C}_{\mathcal{O}}^{i}(1,0) = \{(1,1), (01,2), (11,4)\}$, para $(w,k) = (0,3)$ teremos $\mathcal{C}_{\mathcal{O}}^{d}(0,3) = \{(1,0)\}$ e $\mathcal{C}_{\mathcal{O}}^{i}(0,3) = \{(11,1),(1,3), (01,4)\}$. Observamos que $[\mathcal{C}_{\mathcal{O}}^{d}(1,0)] = [\mathcal{C}_{\mathcal{O}}^{d}(0,3)]$, contudo a seqüência $001011 \in [\mathcal{C}_{\mathcal{O}}(0,1)]$ pois $11 \prec_4 001011$, no entanto, $001011 \notin [\mathcal{C}_{\mathcal{O}}(0,3)]$, pois não há fatores do conjunto $< \mathcal{C}_{\mathcal{O}}^{i}(0,3) >$ em 001011. Portanto, $[\mathcal{C}_{\mathcal{O}}(1,0)] \neq [\mathcal{C}_{\mathcal{O}}(0,3)]$.

Na próxima proposição demonstraremos que $\mathcal{C}_{\mathcal{O}}(w,k)$ é um possível conjunto das restrições para uma palavra $w \in L^{(k)}$.

Proposição 4.3. *Seja $w \in L^{(k)}$, em que L é a linguagem de $X_{\{\mathcal{O},T\}}$, então $[\mathcal{C}(w,k)] = [\mathcal{C}_{\mathcal{O}}(w,k)]$.*

Demonstração: Para todo $s \in [\mathcal{C}_{\mathcal{O}}(w,k)]$ há um fator $u \prec_j s$ tal que $(u,j) \in \mathcal{C}_{\mathcal{O}}^{d}(w,k) \cup < \mathcal{C}_{\mathcal{O}}^{i}(w,k) >$. Se $(u,j) \in < \mathcal{C}_{\mathcal{O}}^{i}(w,k) >$ então $(u, j+k+|w| \mod T) \in \mathcal{O}$ e portanto $ws \notin L^{(k)}$. No entanto, se $(u,j) \in \mathcal{C}_{\mathcal{O}}^{d}(w,k)$ então existe $p \in \mathcal{S}(w)\backslash\{\varepsilon\}$ tal que $pu \in \mathcal{O}^{(k+|w|-|p| \mod T)}$, como $pu \prec_{|w|-|p|} ws$ então $ws \notin L^{(k)}$. Logo, $[\mathcal{C}_{\mathcal{O}}(w,k)] \subseteq [\mathcal{C}(w,k)]$.

Se considerarmos que $s \in [\mathcal{C}(w,k)]$, decorre da Definição 4.5 que existe $(u,j) \in \mathcal{C}(w,k)$ tal que $u \prec_j s$. Da Definição 4.4.a temos que $ws \notin L^{(k)}$, assim existe $v \in \mathcal{O}^{(k+t \mod T)}$ tal que $v \prec_t ws$. No entanto $v \not\prec_t w$ (pois $w \in L^{(k)}$). Caso $t \geq |w|$, então $s \in [< \mathcal{C}_{\mathcal{O}}^{i}(w,k) >]$ pois $(v, t-|w| \mod T) \in \mathcal{C}_{\mathcal{O}}^{i}(w,k)$. Se $t < |w|$, então existe $p \in \mathcal{S}(w)\backslash\{\varepsilon\}$ e $q \in \mathcal{P}(s)\backslash\{\varepsilon\}$ tal que $v = pq \in \mathcal{O}^{(k+|w|-|p| \mod T)}$, logo $s \in [\mathcal{C}_{\mathcal{O}}^{d}(w,k)]$. Concluímos que $[\mathcal{C}(w,k)] \subseteq [\mathcal{C}_{\mathcal{O}}(w,k)]$, e o resultado segue. ■

Como no caso de um SFT há uma forma recursiva para calcularmos os conjuntos $\mathcal{C}_{\mathcal{O}}^{d}(w,k)$. Para uma palavra não nula $w \in L^{(k)}$, $w = a_0 a_2 \ldots a_{n-1}$, o cálculo é realizado do prefíxo de menor comprimento (a_0) para o de maior comprimento $a_0 \ldots a_{n-1}$, empregando a definição $\mathcal{R}(\mathcal{C}_{\mathcal{O}}^{d}(w,k)) = \{s \mid (s,0) \in \mathcal{C}_{\mathcal{O}}^{d}(w)\}$, temos que:

$$\begin{aligned}
\mathcal{R}(\mathcal{C}_{\mathcal{O}}^{d}(a_0,k)) &= a_0^{-1}\mathcal{O}^{(k)}, \\
\mathcal{R}(\mathcal{C}_{\mathcal{O}}^{d}(a_0a_1,k)) &= a_1^{-1}\mathcal{R}(\mathcal{C}_{\mathcal{O}}^{d}(a_0,k)) \cup a_1^{-1}\mathcal{O}^{(k+1 \mod T)}, \\
&\vdots \\
\mathcal{R}(\mathcal{C}_{\mathcal{O}}^{d}(a_0 \ldots a_{n-1},k)) &= a_{n-1}^{-1}\mathcal{R}(\mathcal{C}_{\mathcal{O}}^{d}(a_0 \ldots a_{n-2},k)) \cup a_{n-1}^{-1}\mathcal{O}^{(k+n-1 \mod T)}.
\end{aligned} \tag{4.1}$$

A prova desta relação é apresentada à seguir pelo desenvolvimento do conjunto $\mathcal{R}(\mathcal{C}_{\mathcal{O}}^{d}(w,k))$, a partir da descrição do conjunto $\mathcal{C}_{\mathcal{O}}^{d}(w,k)$ dada na Definição 4.7.

$$
\begin{aligned}
\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(a_0\dots a_{n-1},k)) &= \bigcup_{i=1}^{n}(a_{n-i}\dots a_{n-1})^{-1}\mathcal{O}^{(k+n-i \mod T)}\\
&= \Bigg(\bigcup_{i=2}^{n}(a_{n-i}\dots a_{n-1})^{-1}\mathcal{O}^{(k+n-i \mod T)}\Bigg)\cup a_{n-1}^{-1}\mathcal{O}^{(k+n-1 \mod T)}\\
&= \Bigg(\bigcup_{i=2}^{n}a_{n-1}^{-1}(a_{n-i}\dots a_{n-2})^{-1}\mathcal{O}^{(k+n-i \mod T)}\Bigg)\cup a_{n-1}^{-1}\mathcal{O}^{(k+n-1 \mod T)}\\
&= a_{n-1}^{-1}\Bigg(\bigcup_{i=2}^{n}(a_{n-i}\dots a_{n-2})^{-1}\mathcal{O}^{(k+n-i \mod T)}\Bigg)\cup a_{n-1}^{-1}\mathcal{O}^{(k+n-1 \mod T)}\\
&= a_{n-1}^{-1}\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(a_0\dots a_{n-2},k))\cup a_{n-1}^{-1}\mathcal{O}^{(k+n-1 \mod T)}.
\end{aligned}
$$

Exemplo 4.4. Seja $T = 3$ o período do Shannon cover de um PFT. Dados os conjuntos $\mathcal{O}^{(0)} = \{1111, 1011\}$, $\mathcal{O}^{(1)} = \emptyset$ e $\mathcal{O}^{(2)} = \{111, 1101\}$, segue o cálculo dos conjuntos $\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(w,k))$ para $(111,0)$ e $(110,2)$:

$$
\begin{aligned}
&\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(1,0)) = 1^{-1}\mathcal{O}^{(0)} = \{111, 011\},\\
&\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(11,0)) = 1^{-1}\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(1,0))\cup 1^{-1}\mathcal{O}^{(1)} = \{11\},\\
&\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(111,0)) = 1^{-1}\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(11,0))\cup 1^{-1}\mathcal{O}^{(2)} = \{1, 11, 101\},\\
&\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(1,2)) = 1^{-1}\mathcal{O}^{(2)} = \{11, 101\},\\
&\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(11,2)) = 1^{-1}\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(1,2))\cup 1^{-1}\mathcal{O}^{(0)} = \{1, 01, 111, 011\},\\
&\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(110,2)) = 1^{-1}\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(11,2))\cup 1^{-1}\mathcal{O}^{(1)} = \{1, 11\}.
\end{aligned}
$$

O Lema 4.4 nos permite concluir, para um PFT irredutível de linguagem L, que, para qualquer $w \in L^{(k)}$ e $(v,n) \in \mathcal{C}^i_{\mathcal{O}}(w,k)$ existe uma palavra uv em $[\mathcal{C}_{\mathcal{O}}(w,k)]$ que só possui $v \prec_{|u|} uv$ como fator proibido, onde $|u| \equiv n \mod T$. Portanto, quando consideramos a linguagem indexada $L^{(k)}$, o único fator proibido de (wuv,k) é $(v, |wu|+k \mod T)$, onde $(|wu|+k)-(k+|w|) \equiv n \mod T$. Assim, qualquer elemento em $\mathcal{C}^i_{\mathcal{O}}(w,k)$ é uma restrição não dispensável do conjunto $\mathcal{C}_{\mathcal{O}}(w,k)$, ou seja, retirá-la do conjunto $\mathcal{C}_{\mathcal{O}}(w,k)$ implica que a Proposição 4.1 não seria verificada.

Lema 4.4. *Seja $X_{\{\mathcal{O},T\}}$ um PFT irredutível. Para qualquer $v = pa \in \mathcal{A}^*$, $a \in \mathcal{A}$, $v \in \mathcal{O}^{(n)}$ se, e somente se,*

(i) $\forall s \prec_\ell v$, $s \in L^{(n+\ell \mod T)}$;

(ii) $wuv \notin L^{(k)}$ *para todo* $wup \in L^{(k)}$ *em que* $k + |wu| \equiv n \mod T$, *qualquer* $k \in \{0,\dots,T-1\}$ *e* $w \in L^{(k)}$.

Demonstração: Chamaremos de $\mathcal{G}$ o Shannon cover de $X_{\{\mathcal{O},T\}}$ (a irredutibilidade de $X_{\{\mathcal{O},T\}}$ garante sua existência), $T = \mathsf{per}(A_{\mathcal{G}})$ e $D_0,\dots,D_{T-1}$ as classes periódicas de $\mathcal{G}$. A implicação

direta segue da definição dos elementos em $\mathcal{O}$ para afirmação (i). Para afirmação (ii), seja $k \in \{0,\ldots,T-1\}$ e $w \in L^{(k)}$, portanto, existe pelo menos um caminho ξ em $\mathcal{G}$ tal que $\mathcal{L}(\xi) = w$, $i(\xi) \in D_k$ e $t(\xi) \in D_{(k+|w| \mod T)}$. Tomemos um caminho π em $\mathcal{G}$ para o qual $\mathcal{L}(\pi) = p$ e $i(\pi) \in D_n$, como $p \in L^{(n)}$ então este caminho existe. Já que $\mathcal{G}$ é irredutível, existe um caminho ϕ em $\mathcal{G}$ para o qual $i(\phi) = t(\xi)$ e $t(\phi) = i(\pi)$, logo temos que $w\mathcal{L}(\phi)p \in L^{(k)}$ e $w\mathcal{L}(\phi)v \notin L^{(k)}$, uma vez que o vértice inicial do caminho com rótulo v em $w\mathcal{L}(\phi)v$ pertence à classe D_n. O que concluí a prova direta, pois, w é uma palavra qualquer em $L^{(k)}$ e π é qualquer caminho, tal que, $\mathcal{L}(\pi) = p$ e $i(\pi) \in D_n$.

Da afirmação (i), temos que $v_{[0,|v|-2]} \in L^{(n)}$ e $v_{[1,|v|-1]} \in L^{(n+1 \mod T)}$. Da afirmação (ii), temos que $v \notin L^{(n)}$. Portanto, pela definição dos elementos do conjunto $\mathcal{O}$, concluímos que $v \in \mathcal{O}^{(n)}$.

■

Os conjuntos $\mathcal{C}_{\mathcal{O}}(w,k)$ podem ser reduzidos pela eliminação de elementos $(s,k) \in \mathcal{C}_{\mathcal{O}}(w,k)$ que possuem prefixos próprios em $\mathcal{C}_{\mathcal{O}}(w,k)$, o que é justificado observando-se que para todo prefixo (s',k) de (s,k) temos que $[(s,k)] \subset [(s',k)]$, portanto concluímos que $[\mathcal{C}_{\mathcal{O}}(w,k)] = [\mathcal{C}_{\mathcal{O}}(w,k) \backslash \{(s,k)\}]$. Escreveremos $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k)$ para indicar que eliminações deste tipo não podem ser realizadas no conjunto $\mathcal{C}_{\mathcal{O}}(w,k)$. Observamos que um elemento $(s,j) \in \mathcal{C}_{\mathcal{O}}(w,k)$ só poderá ter um fator próprio $(s',i) \in \mathcal{C}_{\mathcal{O}}(w,k)$ se $(s',i) \in \mathcal{C}^d_{\mathcal{O}}(w,k)$, ou seja $i = j = 0$, caso contrário $s' \in \mathcal{O}^{(i+k+|w| \mod T)}$ será um fator próprio de $s \in \mathcal{O}^{(j+k+|w| \mod T)}$, o que contradiz a definição dos elementos de $\mathcal{O}$.

Exemplo 4.5. Os conjuntos $\mathcal{R}(\mathcal{C}^d_{\mathcal{O}}(w,k))$ calculados no Exemplo 4.4 possuem os seguintes conjuntos $\mathcal{R}(\tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k))$ correspondentes:

$$\begin{aligned}
&\mathcal{R}(\tilde{\mathcal{C}}^d_{\mathcal{O}}(1,0)) = \{111, 011\},\\
&\mathcal{R}(\tilde{\mathcal{C}}^d_{\mathcal{O}}(11,0)) = \{11\},\\
&\mathcal{R}(\tilde{\mathcal{C}}^d_{\mathcal{O}}(111,0)) = \{1\},\\
&\mathcal{R}(\tilde{\mathcal{C}}^d_{\mathcal{O}}(1,2)) = \{11, 101\},\\
&\mathcal{R}(\tilde{\mathcal{C}}^d_{\mathcal{O}}(11,2)) = \{1, 01, 011\},\\
&\mathcal{R}(\tilde{\mathcal{C}}^d_{\mathcal{O}}(110,2)) = \{1\}.
\end{aligned}$$

No próximo teorema mostraremos que elementos em $L \times \{0,\ldots,T-1\}$ com o mesmo contexto à direita possuem conjuntos $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k)$ iguais, dado que o PFT é irredutível.

Teorema 4.5. *Seja $w \in L^{(k)}$ e $w' \in L^{(j)}$, então $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(w',j)$ se, e somente se, $\mathsf{F}(w,k) = \mathsf{F}(w',j)$.*

Demonstração: Se $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(w',j)$ então $[\tilde{\mathcal{C}}_{\mathcal{O}}(w,k)] = [\tilde{\mathcal{C}}_{\mathcal{O}}(w',j)]$, o que das Proposições 4.1 e 4.3 implica que $\mathsf{F}(w,k) = \mathsf{F}(w',j)$.

Para provarmos a recíproca, inicialmente demonstraremos que se $\mathsf{F}(w,k) = \mathsf{F}(w',j)$ então $\tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j)$ e usaremos isto para provar que $\tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$. Seja $(v,n) \in \tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k)$. Supondo que $v = pa$, $a \in \mathcal{A}$, para todo $up \in \mathsf{F}(w,k)$, tal que $|u| \equiv n \mod T$, temos que $upa = uv \notin \mathsf{F}(w,k)$, logo a igualdade $\mathsf{F}(w,k) = \mathsf{F}(w',j)$ implica que $uv \notin \mathsf{F}(w',j)$. Da definição dos elementos do conjunto $\tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k)$, para todo fator próprio $s \prec_\ell v$ de v existe u', tal que $|u'| \equiv n + \ell \mod T$, satisfazendo $u's \in \mathsf{F}(w,k)$ e, portanto, $u's \in \mathsf{F}(w',j)$, implicando que $s \in L^{(n+j+|w'|+\ell \mod T)}$. Assim, a partir do Lema 4.4 temos que $v \in \mathcal{O}^{(n+j+|w'| \mod T)}$, logo $(v,n) \in \tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j)$, de onde concluímos que $\tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k) \subseteq \tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j)$. Supondo que $(v,n) \in \tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j)$, podemos concluir de forma semelhante que $\tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j) \subseteq \tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k)$.

Dadas as igualdades $\mathsf{F}(w,k) = \mathsf{F}(w',j)$ e $\tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j)$, demonstraremos que a hipótese $\tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k) \neq \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$ gera uma contradição. Assim, seja $\tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k) \neq \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$ e s um dos elementos entre os de menor comprimento em $\{u|\ (u,0) \in (\tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k) \backslash \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)) \cup (\tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j) \backslash \tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k))\}$. Iremos supor, sem perda de generalidade, que $(s,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k)$ e portanto que $s \in [\tilde{\mathcal{C}}_{\mathcal{O}}(w,k)]$. Existe $v = ps$ tal que $v \in \mathcal{O}^{(k+|w|-|p| \mod T)}$ e $p \in \mathcal{S}(w) \backslash \{\varepsilon\}$, portanto, se considerarmos que $s \in [< \tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j) >] = [< \tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k) >]$, então existe $s' \prec_{|p+t|} v$ para algum $(s',t) \in \tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k)$, isto implica que $s' \in \mathcal{O}^{(k+|w|+t \mod T)}$ é um fator de $v \in \mathcal{O}^{(k+|w|-|p| \mod T)}$, o que é uma contradição. Logo, podemos afirmar que $s \notin [< \tilde{\mathcal{C}}^i_{\mathcal{O}}(w',j) >]$. Nos limitando aos elementos de $\tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$, temos três casos a considerar (i) $s \notin [\{(u,0)|\ (u,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$ e $|u| > |s|\}]$, uma vez que $u \not\prec_0 s$ para qualquer u tal que $|u| > |s|$; (ii) $s \notin [\{(u,0)|\ (u,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$ e $|u| = |s|\}]$, já que por hipótese $(s,0) \notin \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$; (iii) $s \notin [\{(u,0)|\ (u,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$ e $|u| < |s|\}]$, pois da definição da palavra s, temos que $\{(u,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k)|\ |u| < |s|\} = \{(u,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)|\ |u| < |s|\}$ implicando que não há $(v,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w',j)$ tal que $v \not\prec_0 s$. Logo, $s \notin [\tilde{\mathcal{C}}_{\mathcal{O}}(w',j)]$ e, portanto, da Proposição 4.1, concluímos que $\mathsf{F}(w,k) \neq \mathsf{F}(w',j)$, o que contradiz a hipótese de (w,k) e (w',j) terem o mesmo contexto à direita. ■

4.3 Construção de uma apresentação reduzida

Nesta seção proporemos um método para gerar uma apresentação $G = (\mathcal{V}, \mathcal{E}, \lambda)$ de um PFT a partir de um conjunto mínimo periódico de palavras proibidas $\mathcal{O}$. Começamos por definir o conjunto $\mathcal{V}$, que tem como elementos as classes de equivalência provenientes da partição de $\mathcal{W} \cup \mathcal{O}$, sendo $\mathcal{W}$

definido a seguir.

Definição 4.8. Dado o conjunto $\mathcal{O}$ associado a um PFT, definimos o conjunto $\mathcal{W} = \bigcup_{k=0}^{T-1} \mathcal{W}^{(k)}$, onde:

$$\mathcal{W}^{(k)} = \begin{cases} \{(u,k) |\ u \in \mathcal{P}(w\mathcal{A}^{-1}),\ \forall\ (w,k) \in \mathcal{O}^{(k)}\}\ , \text{ se } \mathcal{O}^{(k)} \neq \emptyset, \\ \{(\varepsilon, k)\} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad , \text{ c.c. .} \end{cases}$$

A partição do conjunto $\mathcal{W} \cup \mathcal{O}$ é determinada pela relação de equivalência: $(w,k) \sim (u,j)$ se, e somente se, $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(u,j)$. Portanto, $(w,k) \sim (u,j)$ se, e somente se, $\mathsf{F}(w,k) = \mathsf{F}(u,j)$, justificado pelo Teorema 4.5. Decorre dessa relação que o conjunto $\mathcal{O}$ forma uma classe, já que para todo $(v,k) \in \mathcal{O}$ e $x \in L$ temos que $vx \notin L^{(k)}$.

A partir de um conjunto $\mathcal{V}$, iremos determinar o conjunto de ramos e seus rótulos. Seja $C_i, C_j \in \mathcal{V}$ e $C_i \neq \mathcal{O}$, criaremos um ramo de C_i para C_j com rótulo a, se para algum dos $(w,k) \in C_i$ o mais longo sufixo de (wa,k) em $\mathcal{W} \cup \mathcal{O}$ pertence a C_j. Descrevemos com maior precisão o método de determinação dos ramos na próxima definição.

Definição 4.9. Seja $C_i \in \mathcal{V}\backslash\{\mathcal{O}\}$ e $(w,k) \in C_i$. A relação $\delta : \mathcal{V}\backslash\{\mathcal{O}\} \times \mathcal{A} \cup \{\varepsilon\} \to \mathcal{V}$ é definida como

$$\delta(C_i, a) = \{\mathcal{Q} |\ (v, k + r \mod T) \in \mathcal{Q},\ a \in \mathcal{A} \cup \{\varepsilon\} \text{ e } v = wa_{[r,|wa|-1]} \text{ é o mais longo sufixo de } (wa,k) \text{ tal que } (v, k + r \mod T) \in \mathcal{W} \cup \mathcal{O}\}.$$

A partir da Definição 4.9, observamos que não há ramos partindo da classe $\mathcal{O}$. Tendo definido o conjunto de vértices, os ramos e seus rótulos, seguiremos demonstrando que G é uma apresentação determinística e reduzida de um PFT irredutível especificado por um conjunto mínimo periódico de palavras proibidas $\mathcal{O}$ e período T. Nos próximos dois lemas apresentaremos algumas propriedades dos conjuntos $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k)$ que nos permitirão concluir que G é um grafo determinístico.

Lema 4.6. *Para todo $w \in L^{(k)}$ e $u \in L^{(j)}$, se $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(u,j)$ então $\tilde{\mathcal{C}}_{\mathcal{O}}(wa,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(ua,j)$.*

Demonstração: Pelo Teorema 4.5, $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(u,j)$ se, e somente se, $\mathsf{F}(w,k) = \mathsf{F}(u,j)$. Logo $waz \in L^{(k)}$ se, e somente se, $waz \in L^{(j)}$, portanto $\mathsf{F}(wa,k) = \mathsf{F}(ua,j)$, ou $\tilde{\mathcal{C}}_{\mathcal{O}}(wa,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(ua,j)$. ■

Lema 4.7. *Seja $w \in L^{(k)}$. Se (v,ℓ) é o mais longo sufixo de (w,k) em $\mathcal{W}$, então $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(v,\ell)$.*

Demonstração: Supondo que $v = w_{[r,|w|-1]}$, então o comprimento de v é $|v| = |w| - r$. Logo, a partir da equação $\ell \equiv k + r \mod T$, temos que $|v| + \ell \equiv |w| - r + k + r \equiv |w| + k \mod T$, portanto $\mathcal{C}^i_{\mathcal{O}}(w,k) = \mathcal{C}^i_{\mathcal{O}}(v,\ell)$.

Observemos que $\mathcal{C}^d_{\mathcal{O}}(v,\ell) \subseteq \mathcal{C}^d_{\mathcal{O}}(w,k)$, pois (v,ℓ) é uma sufixo de (w,k). Para a demonstração da inclusão reversa, consideremos que $(s,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k)$, logo existe $p = w_{[i,|w|-1]}$ tal que $ps \in \mathcal{O}^{(k+i \mod T)}$, portanto $(p, k+i \mod T)$ é um sufixo de (w,k) em $\mathcal{W}$. Uma vez que (v,ℓ) é o mais longo sufixo de (w,k) em $\mathcal{W}$, então $(p, k+i \mod T)$ também é um sufixo de (v,ℓ), o que implica que $(s,0) \in \mathcal{C}^d_{\mathcal{O}}(v,\ell)$. Se $(s',0)$ é um prefixo de $(s,0)$ contido em $\mathcal{C}^d_{\mathcal{O}}(v,\ell)$, então $(s',0) \in \mathcal{C}^d_{\mathcal{O}}(w,k)$ implicando que $(s,0) \notin \tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k)$, o que é uma contradição. Logo, temos que $\tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k) \subseteq \tilde{\mathcal{C}}^d_{\mathcal{O}}(v,\ell)$. Concluímos que $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(v,\ell)$. ■

Proposição 4.8. *O grafo G é determinístico.*

Demonstração: Para $C_i \neq \mathcal{O}$ e $(w,k),(u,j) \in C_i$, consideremos (w',k') e (u',j') os mais longos sufixos de (wa,k) e (ua,j) em $\mathcal{W} \cup \mathcal{O}$, respectivamente. Do Lema 4.6 $\tilde{\mathcal{C}}_{\mathcal{O}}(wa,k) = \tilde{\mathcal{C}}_{\mathcal{O}}(ua,j)$, então para $(w',k') \in C_t$ e $(u',j') \in C_\ell$ temos que $C_t = C_\ell$, pois do Lema 4.7 decorre a igualdade $\tilde{\mathcal{C}}_{\mathcal{O}}(w',k') = \tilde{\mathcal{C}}_{\mathcal{O}}(u',j')$. ■

Como apresentada na Definição 4.9 a relação δ é uma função, a Proposição 4.8 nos permite concluir que esta função é bem definida.

Seja $C_i \neq \mathcal{O}$, uma palavra u é dita definida por C_i se, e somente se, existe um caminho partindo de C_i com rótulo u. O que nos leva a uma natural modificação do domínio da função δ para grafos determinísticos, de $\mathcal{V} \times \mathcal{A} \cup \{\varepsilon\}$ para $\mathcal{V} \times \mathcal{A}^*$. Para qualquer $C_i \in \mathcal{V} \backslash \{\mathcal{O}\}$ e $u \in \mathcal{A}^*$, caso o estado $\mathcal{O}$ seja alcaçado a partir de C_i por um prefixo próprio de u, então u não é definida por C_i, pois $\mathcal{O}$ não possui ramos partindo deste; caso contrário, $\delta(C_i,u)$ é o estado alcançado a partir de C_i por u.

No próximo teorema empregaremos este conceito para concluir que G é uma apresentação de $X_{\{\mathcal{O},T\}}$. Por G ser determinístico, para qualquer palavra u definida por um vértice C_i há um único caminho π em G tal que $\lambda(\pi) = u$ e $i(\pi) = C_i$, portanto, se $\delta(C_i,u) = \mathcal{O}$ então $u \notin F(C_i)$, pois não há ramos partindo de $\mathcal{O}$ e portanto este não pertence a componente essencial de G.

Lema 4.9. *Seja $(w,k) \in C_i$ e $C_i \neq \mathcal{O}$. Se u é definido por C_i, então o mais longo sufixo de (wu,k) contido em $\mathcal{W} \cup \mathcal{O}$ pertence a $\delta(C_i,u)$.*

Demonstração: Realizaremos uma prova por indução no comprimento de u. Inicialmente, se $u = \varepsilon$ então a afirmação é satisfeita. Portanto, iremos supor que o lema é satisfeito para algum

$|u| = r > 0$. Seja $x = ua$ uma palavra definida por C_i, $a \in \mathcal{A}$, portanto, existe um ramo partindo do vértice $\delta(C_i, u)$ com rótulo a para o vértice $\delta(\delta(C_i, u), a)$. Da hipótese indutiva o mais longo sufixo de (wu, k) em $\mathcal{W} \cup \mathcal{O}$ pertence a $\delta(C_i, u)$ e $\delta(C_i, u) \subseteq \mathcal{W}$.

Seja $(u', k') \in \delta(C_i, u)$ o mais longo sufixo de (wu, k) em $\mathcal{W}$ e seja $v = u'a_{[t,|u'a|-1]}$, tal que $(v, k' + t \mod T)$, o mais longo sufixo de $(u'a, k')$ em $\mathcal{W} \cup \mathcal{O}$. Consideremos que $v'a = wua_{[t',|ua|-1]}$ seja um sufixo de (wua, k) em $\mathcal{W} \cup \mathcal{O}$. Então $(v', t' + k \mod T)$ é um sufixo de (wu, k) em $\mathcal{W} \cup \mathcal{O}$, portanto $(v', t' + k \mod T)$ é um sufixo de (u', k'), implicando que $(v'a, t' + k \mod T)$ é um sufixo de $(u'a, k')$. Assim, concluímos que $(v, k' + t \mod T)$ é o mais longo sufixo de (wua, k) em $\mathcal{W} \cup \mathcal{O}$. ■

Teorema 4.10. *A componente essencial de G é uma apresentação reduzida do PFT.*

Demonstração: Seja x uma palavra não nula e $C_i \in \mathcal{V}\backslash\{\mathcal{O}\}$, tal que $x \in \mathsf{F}(C_i)$, $(w, k) \in C_i$ e $\delta(C_i, x) \neq \mathcal{O}$. Considerando que (wx, k) possui fatores em $\mathcal{O}$, seja $v = wx_{[t,|v|+t-1]}$ o fator de (wx, k) em $\mathcal{O}$ que inicia no menor coeficiente t de wx. Então, do Lema 4.9 temos que $\delta(C_i, wx_{[0,t+|v|-1]}) = \mathcal{O}$, logo $x \notin \mathsf{F}(C_i)$, o que é uma contradição. Isso implica que $\mathsf{F}(C_i) \subseteq \mathsf{F}(w, k)$. Reciprocamente, seja $x \in \mathsf{F}(w, k)$, então (wx, k) não possui fatores em $\mathcal{O}$. Se $(w, k) \in C_i$, então, para todo prefixo x' de x temos que $\delta(C_i, x') \neq \mathcal{O}$, logo $x \in \mathsf{F}(C_i)$. Isso implica que $\mathsf{F}(w, k) \subseteq \mathsf{F}(C_i)$. Assim, concluímos que $\mathsf{F}(C_i) = \mathsf{F}(w, k)$ para qualquer $(w, k) \in C_i$.

Dada uma palavra $w \in L^{(k)}$ e sendo (w', k') seu mais longo sufixo em $\mathcal{W}$ (como $w \in L^{(k)}$, (w, k) não possui fatores em $\mathcal{O}$), a partir do Lema 4.7 e do Teorema 4.5, concluímos que $\mathsf{F}(w, k) = \mathsf{F}(w', k') = \mathsf{F}(C_i)$, tal que $(w', k') \in C_i$. Uma vez que $\mathsf{F}(\varepsilon, k) = L^{(k)}$ e para qualquer palavra $x \in L$ temos que $x \in L^{(j)}$ para alguma fase $j \in \{0, \ldots, T-1\}$, concluímos que $\bigcup_{C_i \in \mathcal{V}\backslash\{\mathcal{O}\}} \mathsf{F}(C_i) = L$. Portanto, o subgrafo de G com conjunto de vértices $\mathcal{V}\backslash\{\mathcal{O}\}$ é uma apresentação do PFT, logo, a componente essencial de G também é uma apresentação do PFT. Para demonstração que G é uma apresentação reduzida, sejam $C_i, C_\ell \in \mathcal{V}\backslash\{\mathcal{O}\}$ tal que $\mathsf{F}(C_i) = \mathsf{F}(C_\ell)$, se $(w, k) \in C_i$ e $(u, j) \in C_\ell$ então, do Teorema 4.5, temos que $\tilde{\mathcal{C}}_{\mathcal{O}}(w, k) = \tilde{\mathcal{C}}_{\mathcal{O}}(u, j)$, logo $C_i = C_\ell$, o que implica que a componente essencial é reduzida. ■

4.4 Sistemas restritos com posições irrestritas

Sistemas restritos com posições irrestritas são definidos em [2] como: Seja X um sistema simbólico fechado, T um período e $U \subseteq \{0, \ldots, T-1\}$ um conjunto de *posições irrestritas.* Para

qualquer seqüência finita x da linguagem (respectivamente, infinita, bi-infinita), uma U-permutação de x é uma seqüência finita y (respectivamente, infinita, bi-infinita), tal que $y_i = x_i$ sempre que $i \mod T \notin U$. Se $\mathcal{A}$ é o alfabeto binário $\{0, 1\}$, uma U-permutação é obtida pela inversão ou não dos dígitos nas posições irrestritas. O conjunto de todas as U-permutações das palavras em um conjunto S é chamado *U-fechamento* de S.

Define-se por $X_{U,T}$ o conjunto de todas as seqüências x de X (respectivamente, infinita, bi-infinita) para as quais:

- ▷ Toda U-permutação de x pertence a X;
- ▷ $x_i = 1$ para toda posição i para a qual $i \mod T \in U$.

As posições irrestritas são forçadas a serem 1 para estabelecer um *líder* em cada classe de U-permutações. O importante é que os valores nas posições irrestritas possam ser alterados independentemente sem com isto violar a restrição de X. Os deslocamentos das seqüências em $X_{U,T}$ podem não estar em $X_{U,T}$ (*i.e*, $X_{U,T}$ não é um sistema simbólico fechado). O conjunto de todos os deslocamentos das seqüências em $X_{U,T}$ é denotado por $X^{\sigma}_{U,T}$. O conjunto de todas as seqüências bi-infinitas e deslocadas decorrentes do conjunto $X^{\sigma}_{U,T}$ é denotado por $\overline{X^{\sigma}_{U,T}}$. A conexão entre os conjuntos $\overline{X^{\sigma}_{U,T}}$, $X^{\sigma}_{U,T}$ e os PFT é dado na próxima proposição.

Proposição 4.11. [3, Proposição 1] *Seja X um SFT, T um período e U um conjunto de posições irrestritas. Os sistemas simbólicos fechados $\overline{X^{\sigma}_{U,T}}$ e $X^{\sigma}_{U,T}$ são PFT.*

Este resultado é provado construtivamente pela definição dos conjuntos proibidos associados, e pela demonstração de que estes realmente geram os sistemas simbólicos fechados. Dado um conjunto $U \subseteq \{0, \ldots, T-1\}$ e sendo k um inteiro, o conjunto $U + k$ é definido como $\{i + k \mod T | \ i \in U\}$. Seja $X = \mathsf{X}_{\mathcal{F}}$, então $\overline{X^{\sigma}_{U,T}} = \mathsf{X}_{\mathcal{G}',T}$ tal que $\mathcal{G}'^{(k)}$ é o $(U-k)$-fechamento de $\mathcal{F}$, em que $k \in \{0, \ldots, T-1\}$; $X^{\sigma}_{U,T} = \mathsf{X}_{\mathcal{G},T}$ tal que $\mathcal{G}^{(k)} = \mathcal{G}'^{(k)}$ se $k \in U$ e $\mathcal{G}^{(k)} = \{0\} \cup \mathcal{G}'^{(k)}$ se $k \in \{0, \ldots, T-1\} \backslash U$.

Exemplo 4.6. Considerando a restrição MTR(3), o conjunto proibido associado é $\mathcal{F} = \{1111\}$, portanto, para um período $T = 3$ e conjunto de posições irrestritas $U = \{1\}$, temos que $\mathcal{G}'^{(0)} = \{1111, 1011\}$, $\mathcal{G}'^{(1)} = \{1111, 0111\}$ e $\mathcal{G}'^{(2)} = \{1111, 1101\}$. O conjunto $\mathcal{O}$ associado é o apresentado no Exemplo 4.4. Calculando os conjuntos $\tilde{\mathcal{C}}^{d}_{\mathcal{O}}(w, k)$ restantes, temos que $\tilde{\mathcal{C}}^{d}_{\mathcal{O}}(10, 0) = \{(11, 0)\}$ e $\tilde{\mathcal{C}}^{d}_{\mathcal{O}}(101, 0) = \{(1, 0)\}$. Os conjuntos $\tilde{\mathcal{C}}^{i}_{\mathcal{O}}(w, k)$ associados a cada elemento de $\mathcal{W}$ são apresentados na Tabela 4.6. Assim, temos as seguintes classes de equivalência $C_1 = \{(\varepsilon, 0)\}$, $C_2 = \{(\varepsilon, 1)\}$, $C_3 = \{(\varepsilon, 2)\}$, $C_4 = \{(1, 0)\}$, $C_5 = \{(11, 0), (10, 0)\}$, $C_6 = \{(111, 0), (101, 0)\}$, $C_7 = \{(1, 2)\}$,

$C_8 = \{(11, 2)\}$ e $C_9 = \{(110, 2)\}$. O grafo associado é apresentado na Figura 4.3. Os conjuntos $\tilde{\mathcal{C}}^d_{\mathcal{O}}(w, k)$ e $\tilde{\mathcal{C}}^i_{\mathcal{O}}(w, k)$ dos elementos de $\mathcal{W}$, as classes as quais esses pertencem e o cálculo da função de transição são apresentados na Tabela 4.6. Da Proposição 4.8 e da Proposição 4.10 sabemos que o grafo da Figura 4.3 apresenta o PFT e é reduzido e determinístico. Verificamos que este também é irredutível, portanto, o grafo é o Shannon cover do PFT que satisfaz a restrição MTR(3), com $T = 3$ e $U = \{1\}$.

Para exemplificarmos o cálculo da função δ, consideremos a classe C_5. Da Tabela 4.6, temos que $C_5 = \{(10, 0), (11, 0)\}$, usando $(10, 0)$ para determinarmos $\delta(C_5, 0)$ e $\delta(C_5, 1)$, observe que os máximos prefixos de $(100, 0)$ e $(101, 0)$ em $\mathcal{W}$ são $(\varepsilon, 0)$ e $(101, 0)$, respectivamente, portanto, a partir de $(10, 0)$ obtemos que $\delta(C_5, 0) = C_1$ e $\delta(C_5, 1) = C_6$. Se empregarmos $(11, 0)$, os máximos prefixos de $(110, 0)$ e $(111, 0)$ em $\mathcal{W}$ são $(\varepsilon, 0)$ e $(101, 0)$, respectivamente, o que conduz aos mesmos resultados obtidos quando empregamos $(10, 0)$.

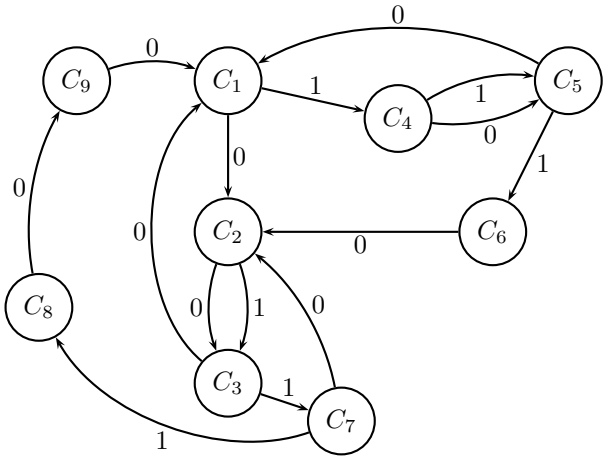


Figura 4.3: Apresentação de um PFT satisfazendo a restrição MTR(3) com $T = 3$ e $U = \{1\}$.

Tabela 4.1: Cálculo das classes de equivalência e da função δ para o PFT do Exemplo 4.6.

$(w,k) \in \mathcal{W}$	$\{s \mid (s,0) \in \tilde{\mathcal{C}}^d_{\mathcal{O}}(w,k)\}$	$\tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k)^\dagger$	*Classes*	$\delta(C_i,0)$	$\delta(C_i,1)$
$(\varepsilon,0)$	$\emptyset$	$\mathcal{A}_0$	C_1	C_2	C_4
$(1,0)$	$\{111,011\}$	$\mathcal{A}_1$	C_4	C_5	C_5
$(10,0)$	$\{11\}$	$\mathcal{A}_2$	C_5	C_1	C_6
$(11,0)$	$\{11\}$	$\mathcal{A}_2$	C_5	C_1	C_6
$(101,0)$	$\{1\}$	$\mathcal{A}_0$	C_6	C_2	$\mathcal{O}$
$(111,0)$	$\{1\}$	$\mathcal{A}_0$	C_6	C_2	$\mathcal{O}$
$(\varepsilon,1)$	$\emptyset$	$\mathcal{A}_1$	C_2	C_3	C_3
$(\varepsilon,2)$	$\emptyset$	$\mathcal{A}_2$	C_3	C_1	C_7
$(1,2)$	$\{11,101\}$	$\mathcal{A}_0$	C_7	C_2	C_8
$(11,2)$	$\{1,01,011\}$	$\mathcal{A}_1$	C_8	C_9	$\mathcal{O}$
$(110,2)$	$\{1\}$	$\mathcal{A}_2$	C_9	C_1	$\mathcal{O}$

† $\mathcal{A}_0 = \{(1111,0),(1011,0),(111,2),(1101,2)\}$.
$\mathcal{A}_1 = \{(111,1),(1101,1),(1111,2),(1011,2)\}$.
$\mathcal{A}_2 = \{(111,0),(1101,0),(1111,1),(1011,1)\}$.

CAPÍTULO 5

CONCLUSÕES

Estabelecemos como objetivo do trabalho a proposição de métodos para geração de apresentações determinísticas para SFT e PFT com número mínimo de estados e que fossem conceitualmente distintos dos encontrados na literatura. A motivação para este trabalho reside na relação de *Myhill-Nerode* e da memória finita dos sistemas simbólicos estudados. O conjunto proibido finito $\mathcal{O}$ com palavras de máximo comprimento M, permite que a verificação que uma palavra da linguagem y pertencente ao contexto à direita de outra palavra da linguagem x, seja realizada determinando-se o subconjunto de $(\mathcal{S}(x)\backslash\{\varepsilon\}) \cdot (\mathcal{P}(y)\backslash\{\varepsilon\})$ contendo todas as palavras com comprimento máximo $M+1$ que não pertencem a $\mathcal{O}$. Exploramos esta propriedade para propor um algoritmo com número finito de passos para a determinação de palavras com mesmo contexto à direita.

Com a determinação em [24] de um conjunto suficiente de palavras para gerar o grafo dos contextos, podemos dividir os procedimentos para geração de uma apresentação mínima, não como a geração de uma apresentação inicial seguida de uma minimização de vértices aplicando-se os algoritmos de Hopcroft ou Moore, mas na determinação das classes de equivalência da relação de Myhill-Nerode (pelo cálculo dos conjuntos das restrições) seguida da construção do grafo, ou seja, propusemos um algoritmo que gera o conjunto de vértices do grafo determinístico mínimo de um SFT irredutível (de maior interesse prático) antes de construir uma apresentação para o SFT. No caso de um SFT redutível, o algoritmo gera a m-SDP que apesar de não ser isomorfa a toda apresentação determinística com número mínimo de estados, possui o número mínimo de estados entre todas as apresentações determinísticas.

Um PFT não é necessariamente um SFT, ou seja, em geral não possui memória finita. A memória finita de um PFT é artificialmente gerada pela inclusão de um novo parâmetro na determinação do

conjunto proibido deste, a saber, a fase. Como as relações de equivalência não são mais definidas sobre as palavras da linguagem, mas sobre as palavras da linguagem indexadas, a relação estabelecida não é mais a de Myhill-Nerode. Observamos com isto, que a determinação das restrições dos elementos passa a depender não só dele e do conjunto proibido, mas também da fase. A partir do conjunto de palavras indexadas proposto em [3], propusemos um procedimento para gerar uma apresentação determinística e reduzida do PFT seguindo o mesmo princípio utilizado para um SFT, ou seja, é determinado o conjunto de estados da apresentação reduzida antes de construir-se uma apresentação para o PFT.

5.1 Trabalhos Futuros

O algoritmo de menor complexidade para gerar uma apresentação inicial de um SFT que temos conhecimento é apresentado em [24]. A complexidade deste algoritmo é linear com o número de estados. Para a obtenção da apresentação mínima, é necessário reduzir a apresentação inicial utilizando os algoritmos de Moore ou Hopcroft. Para o caso de um SFT, no Capítulo 3 propusemos um procedimento para o cálculo dos conjuntos $\tilde{\mathcal{C}}^r_{\mathcal{O}}(w)$. Este pode ser aperfeiçoado com o objetivo de obter-se as classes de equivalência com uma complexidade de no máximo $n \cdot \log n$, que é a complexidade do algoritmo de Hopcroft. Como a função de transição é calculada para um número menor de vértices quando comparado ao algoritmo proposto em [24], teríamos um algoritmo de menor complexidade para geração da apresentação mínima.

Para o caso de um PFT irredutível, conjecturamos que todo conjunto proibido periódico que satisfaça as condições do conjunto $\mathcal{O}$ deva ter um período múltiplo do período do Shannon cover. A obtenção deste resultado simplificaria o cálculo dos conjuntos $\tilde{\mathcal{C}}_{\mathcal{O}}(w,k)$, já que o conjunto $\tilde{\mathcal{C}}^i_{\mathcal{O}}(w,k)$ poderia ser eliminado, pois, dois elementos $(w,k),(w',k')$ com $|w|+k$ e $|w'|+k'$ incongruentes (quando reduzidos pelo período do Shannon cover) teriam contextos à direita distintos.

Como o método proposto para a geração de uma apresentação de um PFT gera uma apresentação reduzida e determinística, deve-se verificar se esta também é irredutível. Portanto, teríamos o Shannon cover do PFT. Pode-se estudar ainda a viabilidade de um algoritmo similar para o caso de um PFT redutível. Por fim, como no caso não periódico, deve-se realizar a análise de complexidade do algoritmo proposto.

BIBLIOGRAFIA

[1] E. R. Berlekamp, “Technology of error-correcting codes,” *Proc. IEEE*, vol. 68, pp. 564–593, 1980.

[2] J. C. de Souza, B. H. Marcus, R. New, and B. A. Wilson, “Constrained systems with unconstrained positions,” *IEEE Trans. Inform. Theory*, vol. 48, pp. 866–879, April 2002.

[3] M.-P. Béal, M. Crochemore, and G. Fici, “Presentations of constrained systems with unconstrained positions,” *IEEE Trans. Inform. Theory*, vol. 51, pp. 1891–1900, May 2005.

[4] 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.

[5] K. A. S. Immink, “EFMPlus: The coding format of the multimedia compact disc,” *IEEE. Trans. Consum. Electron.*, vol. 41, pp. 491–497, August 1995.

[6] W. Hirt, M. Hassner, and N. Heise, “IrDA-VFIr (16 Mb/s): Modulation code and system design,” *IEEE Personal Communications*, vol. 8, pp. 58–71, February 2001.

[7] P. Funk, “Run-length-limited codes with multiple spacing,” *Trans. Magnetics*, vol. 18, pp. 772–775, 1982.

[8] B. H. Marcus and P. H. Siegel, “On codes with spectral nulls at rational sub-multiples of the symbol frequency,” *IEEE Trans. Inform. Theory*, vol. 33, pp. 557–568, July 1987.

[9] M. Morse, “Recurrent geodesics on a surface of negative curvature,” *Trans. Amer. Math. Soc.*, no. 22, pp. 84–110, 1921.

[10] D. Lind and B. H. Marcus, *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, 1995.

[11] R. L. Adler, D. Coppersmith, and M. Hassner, "Algorithms for sliding block codes: An application of symbolic dynamics to information theory," *IEEE Trans. Inform. Theory*, vol. 29, pp. 5–22, January 1983.

[12] M.-P. Béal and D. Perrin, "Symbolic dynamics and finite automata," in *Handbook of formal languages, vol. 2: linear modeling: background and application*, pp. 463–505, Springer-Verlag New York, 1997.

[13] N. Jonaska, "Sofic shifts with synchronizing presentations," *Theoretical Computer Science*, no. 158, pp. 81–115, 1996.

[14] N. T. Sindhushayana, "Symbolic dynamics, automata theory and the theory of coding: A comparative study and applications," Master's thesis, Cornell University, 1992.

[15] S. Eilenberg, *Automata, Languages, and Machines*, vol. A. Academic Press, 1974.

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

[17] M. V. Lawson, *Finite Automata*. Chapman & Hall/CRC, 2004.

[18] R. Johnsonbaugh and W. E. Pfaffenberger, *Foundations of Mathematical Analysis*. Dover Publications, 2002.

[19] 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.

[20] D. P. B. Chaves, C. Pimentel, and B. F. Uchôa-Filho, "On the shannon cover of shifts of finite type," in *Anais do XX Simpósio Brasileiro de Telecomunicações-SBT'03*, 2003.

[21] D. P. B. Chaves, C. Pimentel, and B. F. Uchôa-Filho, "An iterative matrix-based procedure for finding the shannon cover for constrained sequences," *Lecture Notes in Computer Science*, vol. 3124, pp. 88–93, 2004.

[22] K. A. S. Immink, "A survey of codes for optical disk recording," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 756–764, April 2001.

[23] B. H. Marcus, P. H. Siegel, and J. K. Wolf, "Finite-state modulation codes for data storage," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 5–37, January 1992.

[24] M. Crochemore, F. Mignosi, and A. Restrivo, “Automata and forbidden words,” *Information Processing Letters*, vol. 67, pp. 111–117, 1998.

[25] B. Moision and P. H. Siegel, “Periodic-finite-type shift spaces,” in *Proc. IEEE Int. Symp. Inform. Theory*, (Washington, DC), p. 65, June 2001.

[26] A. Wijngaarden and K. S. Immink, “Maximum run-length limited codes with error control properties,” *IEEE J. Select. Areas Commun.*, vol. 19, pp. 602–611, April 2001.

SOBRE O AUTOR

O autor nasceu em Recife, Pernambuco, no dia 09 de maio de 1977. Formado em Engenharia Elétrica (com Láurea), modalidade Eletrônica, pela Universidade Federal de Pernambuco (UFPE) em agosto de 2004.

Entre suas áreas de interesse estão teoria de códigos, processamento digital de sinais, teoria da informação, dinâmica simbólica e teoria algébrica da linguagem.

Endereço: Rua Dr. Efigênio Barbosa, 387
Bancários
João Pessoa – PB, Brasil
C.E.P.: 58.052 – 310

e-mail: ddselec@click21.com.br

Esta dissertação foi diagramada usando LaTeX 2_ε [1] pelo autor.

[1] LaTeX 2_ε é uma extensão do LaTeX. LaTeX é uma coleção de macros criadas por Leslie Lamport para o sistema TeX, que foi desenvolvido por Donald E. Knuth. TeX é uma marca registrada da Sociedade Americana de Matemática ($\mathcal{AMS}$). O estilo usado na formatação desta dissertação foi escrito por Dinesh Das, Universidade do Texas. Modificado em 2001 por Renato José de Sobral Cintra, Universidade Federal de Pernambuco, e em 2005 por André Leite Wanderley.