Luiz Fernando Bossa

Dinâmica Simbólica e Teoria da Computação

Florianópolis, SC
2016

Luiz Fernando Bossa

Dinâmica Simbólica e Teoria da Computação

Trabalho de Conclusão de Curso apresentado ao Curso de Matemática do Departamento de Matemática do Centro de Ciências Físicas e Matemáticas da Universidade Federal de Santa Catarina para obtenção de grau de Bacharel em Matemática e Computação Científica.

Universidade Federal de Santa Catarina
Centro de Ciências Físicas e Matemáticas
Departamento de Matemática

Orientador: Dr. Gilles Golçalves de Castro

Florianópolis, SC
2016

Esta monografia foi julgada adequada como TRABALHO DE CONCLUSÃO DE CURSO no Curso de Matemática - Habilitação Bacharelado, e aprovada em sua forma final pela Banca Examinadora designada pela Portaria 19/2016/CCM.

Prof. Dra. Silvia Martini De Holanda Janesch
Coordenadora do Curso

Banca Examinadora:

Prof Dr. Gilles Gonçalves de Castro
Orientador

Prof. Dr. Fernando de Lacerda Mortari

Prof. Dr. Marcelo Sobottka

Agradecimentos

Em primeiro lugar gostaria de agradecer aos meus pais Roseli e Genir. Tudo o que sou hoje é fruto da criação, aconselhamento, suporte e amor de vocês. Obrigado também a minha irmã Thais Fernanda por todo o carinho.

À minha namorada Giulia, obrigado por todo o amor, carinho e compreensão.

Obrigado a todo mundo que conviveu comigo durante o meu intercâmbio, por terem-no tornado a experiência mais enriquecedora que vivi. Obrigado à CAPES pelo suporte financeiro, e a Thaís, Cris e Yuri por terem me ajudado com as burocracias.

Aos amigos de longa data, aos que conheci na graduação ou em situações inusitadas dessa vida, obrigado por todas as situações boas e todos os perrengues que já vivemos. Saudações a todas as pessoas que já moraram comigo durante todos esses anos de graduação, seja na Morada Pitangueira, no Reino de Oh'Milka ou no 401, vocês fizeram parte de todo meu amadurecimento como pessoa.

Obrigado ao meu orientador Gilles por todos os seis cursos ministrados com maestria durante a graduação, e por toda a disponibilidade durante o desenvolvimento desse projeto. Obrigado aos professores Fernando e Marcelo por terem aceitado fazer parte da minha banca.

Finalmente, obrigado ao CNPq pela bolsa concedida para a realização da pesquisa que resultou nesse trabalho.

Resumo

O primeiro tratamento formal da dinâmica simbólica foi desenvolvido por Morse e Hedlund em 1938 [6], embora a ideia tenha sido apresentada pela primeira vez por Jacques Hadamard em 1898 [2]. Neste trabalho, apresentamos as definições e os principais resultados dessa teoria e estabelecemos uma conexão entre subshifts sóficos e autômatos finitos.

Palavras-chave: sistemas dinâmicos discretos, dinâmica simbólica, autômatos finitos.

Abstract

The first formal treatment of symbolic dynamics was developed by Morse and Hedlund in 1938 [6], although the idea was first introduced by Jacques Hadamard in 1898 [2]. In this work, we present the definitions and main results from that theory, and estabilish a connection between sofic shifts and finite automata.

Palavras-chave: discrete dynamical system, symbolic dynamics, finite automata.

Sumário

Lista de ilustrações

Introdução

Considere um conjunto X e uma função $\phi : X \to X$. Podemos pensar em X como o conjunto de configurações de um sistema físico, por exemplo. Os elementos de X evoluem com o tempo seguindo a função ϕ: dado um elemento $x \in X$ em um instante de tempo $t = 0$, ele será transformado no elemento $\phi(x)$ no instante $t = 1$, e em seguida em $\phi(\phi(x))$ no instante $t = 2$, e $\phi^n(x)$ no instante $t = n$, onde o expoente denota a composição de funções.

Dizemos que o par (X, ϕ) é um *sistema dinâmico discreto*, e um elemento de X é chamado de *estado* do sistema. No nosso estudo, consideramos ϕ inversível. Definimos a *órbita* de x como o conjunto de estados passados de x bem como o conjunto de estados para os quais x evolui, isto é,

$$\{\ldots, \phi^{-2}(x), \phi^{-1}(x), \phi^0(x), \phi^1(x), \phi^2(x), \ldots\}. \tag{1}$$

Considere então que ao invés de nos focarmos em estados específicos, particionamos X em um número finito de subconjuntos disjuntos, digamos $X = P_1 \cup \ldots \cup P_n$, com $P_i \cap P_j = \emptyset$, para todo $1 \leq i \neq j \leq n$. Dada uma órbita de um elemento x, produzimos uma sequência bi-infinita

$$(\ldots, x_{-2}, x_{-1}, x_0, x_1, x_2, \ldots) \tag{2}$$

em que cada x_k é o índice do subconjunto que contém $\phi^k(x)$, i.e., $\phi^k(x) \in P_{x_k}$. Para simplificar a notação, usamos um ponto para distinguir os índices naturais dos índices negativos e omitimos as vírgulas:

$$\ldots x_{-2}x_{-1}.x_0x_1x_2\ldots \tag{3}$$

Assim, ao invés de estudarmos a dinâmica de cada órbita para cada estado, agrupamos os estados em conjuntos e estudamos a dinâmica dos símbolos que representam cada conjunto de estados.

O Capítulo 1 introduz os principais objetos de estudo do nosso texto: shifts completos sobre um alfabeto finito e o operador de shift. São estudados os subconjuntos do shift completo que também são sistemas dinâmicos discretos, os chamados subshifts. Caracterizamos subshifts através de conjuntos de palavras proibidas, bem como estudamos o conjunto das palavras permitidas. Introduzimos uma métrica e caracterizamos os subshifts topologicamente. Também apresentamos os morfismos entre shifts completos.

No Capítulo 2, estudamos o que ocorre quando o conjunto das palavras proibidas é finito. Introduzimos os shifts-arestas, que são subshifts originados por grafos. Provamos que todo subshift do tipo finito pode ser codificado em um shift-aresta. Definimos a operação de separação de estados, e demonstramos o Teorema da Decomposição, que relaciona conjugações entre shifts-aresta com composições de separação de estados.

O Capítulo 3 apresenta os subshifts sóficos, que são subsfhits originados através de grafos rotulados. Mostramos que o conjunto dos subshifts sóficos é o menor subconjunto do shift completo que contém os subshifts do tipo finito e é fechado por códigos fatores. Também apresentamos os autômatos finitos, e mostramos que um subshift é sófico se e somente se sua linguagem é regular.

Nossa principal referência para a parte de dinâmica simbólica dos Capítulos 1, 2 e 3 é [5], *An introduction to symbolic dynamics and coding*, de Douglas Lind e Brian Marcus. O estudo de autômatos segue as referências [7], [3] e [1].

Assume-se que o leitor tenha conhecimentos de teoria de espaços métricos, conforme [4].

Notação e Convenções

Nesse trabalho, $\mathbb{Z}$ denota o conjunto dos números inteiros, $\mathbb{N}$ o conjunto dos inteiros não-negativos e $\mathbb{N}^* = \{1, 2, 3, \ldots\}$.

Dado um conjunto X qualquer, denotamos a função identidade sobre X por id_X.

A diferença entre dois conjuntos é denotada por $A \backslash B = \{x \in A : x \notin B\}$. Quando um conjunto A está contido em um conjunto universo U, denotamos seu complementar por $A^c = U \backslash A$.

Usaremos a seguinte notação para composição de funções: dada $f : X \to X$, defina $f^0 := \mathrm{id}_X$, e para $n \in \mathbb{N}^*$, $f^n := f \circ f^{n-1}$ é a composição de f com ela mesma, $n-1$ vezes; se f for inversível, defina $f^{-n} := (f^{-1})^n$.

Sendo $f : A \to B$ uma função qualquer e dado um conjunto $C \subseteq A$, a imagem direta de C por f é o conjunto $f(C) := \{f(y) \mid y \in C\}$. Para um conjunto $D \subseteq B$, sua imagem inversa por f é $f^{-1}(D) := \{x \in A : f(x) \in D\}$.

Uma partição de um conjunto A é uma família $\{A_\lambda\}_{\lambda \in L}$ de subconjuntos de A indexados por um conjunto L de índices, com a propriedade de que $A = \cup_{\lambda \in L} A_\lambda$ e $A_\lambda \cap A_\gamma = \emptyset$ para quaisquer $\lambda, \gamma \in L$ com $\lambda \neq \gamma$.

1 Dinâmica Simbólica

Neste Capítulo definiremos nosso principal objeto de estudo: conjuntos de sequências infinitas em ambas as direções (adiante chamadas de *bi-infinitas*) cujas entradas pertencem a um conjunto finito de símbolos.

1.1 Shift Completo

Definição 1.1. Seja $\mathcal{A}$ um conjunto finito não vazio, que chamaremos de *alfabeto*. Os elementos de $\mathcal{A}$ serão chamados de *letras*. Uma *sequência bi-infinita* de letras de $\mathcal{A}$ é uma função $f : \mathbb{Z} \to \mathcal{A}$. O *$\mathcal{A}$-shift completo* $\mathcal{A}^{\mathbb{Z}}$ é o conjunto de todas as sequências bi-infinitas.

Em geral, dada uma $f \in \mathcal{A}^{\mathbb{Z}}$, a representamos por extenso como

$$\ldots x_{-2}x_{-1}.x_0x_1x_2\ldots, \tag{1.1}$$

na qual $x_k = f(k)$, para todo $k \in \mathbb{Z}$.

Por vezes, usamos a notação $(x_k)_{k\in\mathbb{Z}}$ como abreviação para a sequência bi-infinita $\ldots x_{-1}.x_0x_1\ldots$ Quando tomamos um elemento $w \in \mathcal{A}^{\mathbb{Z}}$ fica subentendido que sua representação como sequência é $w = \ldots w_{-1}.w_0w_1\ldots$ Por fim, podemos chamar um elemento do shift completo simplesmente de um *ponto* no shift completo.

Exemplo 1.2. Para $r \in \mathbb{N}^*$, quando $\mathcal{A} = \{0, 1, \ldots, r-1\}$, chamamos o $\mathcal{A}$-shift completo de *r-shift completo*. Em particular, quando $\mathcal{A} = \{0, 1\}$, chamamos um elemento de $\mathcal{A}^{\mathbb{Z}}$ de uma *sequência binária*. ◁

Definição 1.3. Um *bloco* ou *palavra* u sobre $\mathcal{A}$ é uma sequência finita de símbolos de $\mathcal{A}$. O *comprimento* do bloco é o número de símbolos que ele contém, denotado por $|u|$. Um bloco u com $|u| = k$ é por vezes chamado de um k-bloco. Denotamos por $\mathcal{A}^k$ o conjunto de todos os k-blocos. Um *sub-bloco* de um bloco $u = a_1 \ldots a_k$ é um bloco da forma $v = a_i \ldots a_j$ com $1 \leq i \leq j \leq k$, e denotamos por $v \subset u$. Consideramos também o símbolo ε, que não pertence ao alfabeto, como o *bloco vazio*. Por definição, o bloco vazio é sub-bloco de todo bloco e definimos $|\varepsilon| = 0$.

Por vezes, quando queremos fazer referência para determinadas partes ou blocos de uma sequência, usamos a seguinte notação: dado $x \in \mathcal{A}^{\mathbb{Z}}$, e um intervalo não-vazio $[a, b] \subseteq \mathbb{Z}$, definimos a palavra $x_{[a,b]} := x_a x_{a+1} \ldots x_b$. Naturalmente, podemos estender essa notação para qualquer tipo de intervalo: quando o intervalo for aberto em alguma extremidade, apenas excluímos o símbolo correspondente. Assim, por exemplo, $x_{[a,b)} = x_a \ldots x_{b-1}$. Se o intervalo for vazio, definimos $x_\emptyset = \varepsilon$. Também definimos as sequências

infinitas de letras $x_{[a,\infty)} = x_a x_{a+1} \dots$ e $x_{(-\infty,b]} = \dots x_{b-1} x_b$. O $(2k+1)$-*bloco central* de x é o bloco $x_{[-k,k]}$.

Exemplo 1.4. Seja $x = \dots 01242203412.41232123 \dots$ um ponto no 5-shift completo. Então $x_{[-3,2]} = 41241$, $x_{(-7,-2]} = 20341$, $x_{(0,\infty)} = 1232123 \dots$, e o 3-bloco central de x é 241. ◁

Dados dois blocos, podemos criar um novo bloco formado pelas letras dos dois primeiros blocos colocados em sequência.

Definição 1.5. Dados dois blocos $u = u_1 \dots u_k$ e $v = v_1 \dots v_m$ definimos a *concatenação* de u e v por

$$uv := u_1 \dots u_k v_1 \dots v_m. \tag{1.2}$$

Para qualquer bloco u, definimos $u\varepsilon = u = \varepsilon u$. Também define-se $u^0 := \varepsilon$, e para $n \in \mathbb{N}^*$, $u^n := uu^{n-1}$. Denotamos por u^∞ a sequência $\dots uu.uu \dots$

Exemplo 1.6. Considerando o alfabeto $\mathcal{A} = \{a, b, c\}$, seja $u = aabc$ e $v = bbb$; então $uv = aabcbbb$ e $vu = bbbaabc$. Já no alfabeto $\mathcal{D} = \{0, 1\}$, sendo $w = 01$, temos $w^2 = 0101$, $w^4 = 01010101$. No alfabeto $\mathcal{A} \cup \mathcal{D}$, temos $uvw = aabcbbb01$, $(vw)^2 u = bbb01bbb01aabc$. ◁

Definiremos agora uma função $\sigma : \mathcal{A}^{\mathbb{Z}} \to \mathcal{A}^{\mathbb{Z}}$ que tornará o par $(\mathcal{A}^{\mathbb{Z}}, \sigma)$ um sistema dinâmico discreto.

Definição 1.7. O *operador de shift* σ em $\mathcal{A}^{\mathbb{Z}}$ é a função $\sigma : \mathcal{A}^{\mathbb{Z}} \to \mathcal{A}^{\mathbb{Z}}$ que associa a cada elemento $x \in \mathcal{A}^{\mathbb{Z}}$ um elemento $\sigma(x) \in \mathcal{A}^{\mathbb{Z}}$ de modo que

$$\sigma(x)_i = x_{i+1} \quad \forall i \in \mathbb{Z}. \tag{1.3}$$

Exemplo 1.8. Sendo $\mathcal{A} = \{0, \alpha\}$, e $x = \dots 000.\alpha 000 \dots$, então $\sigma(x) = \dots 000\alpha.000 \dots$ ◁

Exemplo 1.9. Seja $\sigma : \mathcal{A}^{\mathbb{Z}} \to \mathcal{A}^{\mathbb{Z}}$ o operador de shift e fixe qualquer $n \in \mathbb{Z}$. Usando a notação para composição de funções, veja que para qualquer $x \in \mathcal{A}^{\mathbb{Z}}$ vale que

$$\sigma^n(x)_i = x_{i+n} \quad \forall i \in \mathbb{Z}. \tag{1.4}$$

Assim, sejam $k, m \in \mathbb{N}^*$, u um k-bloco sobre $\mathcal{A}$, w um m-bloco sobre $\mathcal{A}$ e x um ponto da forma $x = x_{(-\infty,-m)} w.u x_{[k,\infty)}$. Temos que $\sigma^k(x) = x_{(-\infty,-m)} wu.x_{[k,\infty)}$ e $\sigma^{-m}(x) = x_{(-\infty,-m)}.wux_{[k,\infty)}$. ◁

Definição 1.10. Um ponto x é *periódico* se existe um $n \in \mathbb{N}^*$ tal que $\sigma^n(x) = x$, e dizemos que x *tem período* n. Se x é periódico, o menor $n \in \mathbb{N}^*$ tal que $\sigma^n(x) = x$ é chamado *menor período* de x. Dizemos que x é um *ponto fixo* para σ se $\sigma(x) = x$.

Proposição 1.11. *Um ponto $x \in \mathcal{A}^{\mathbb{Z}}$ tem período n se e somente se existe um n-bloco u tal que $x = u^\infty$.*

Demonstração. ($\Rightarrow$) Se x tem período n, então $\sigma^n(x) = x$. Defina $u = x_{[0,n-1]}$. Podemos escrever x da forma $x = x_{(-\infty,-1]}.ux_{[n,\infty)}$. Aplicando σ^n, temos que $\sigma^n(x) = x_{(-\infty,-1]}u.x_{[n,\infty)}$. Então, a equação da periodicidade de x se escreve como

$$x_{(-\infty,-1]}.ux_{[n,\infty)} = x_{(-\infty,-1]}u.x_{[n,\infty)}. \tag{1.5}$$

Olhando para as coordenadas dos blocos centrais, temos que $x_{[-n,-1]} = u$ e $x_{[n,2n-1]} = u$. Assim, x se escreve da forma $x = x_{(-\infty,-n-1]}u.uux_{[2n,\infty)}$. Aplicando σ^n novamente, e usando a equação da periodicidade de x, temos que

$$x_{(-\infty,-n-1]}u.uux_{[2n,\infty)} = x_{(-\infty,-n-1]}uu.ux_{[2n,\infty)}. \tag{1.6}$$

Igualando as coordenadas dos blocos centrais, obtemos que

$$x = x_{(-\infty,-2n-1]}uu.uuux_{[3n,\infty)}. \tag{1.7}$$

Repetindo esse processo k vezes, para $k \in \mathbb{N}^*$, temos que x é da forma

$$x = x_{(-\infty,-kn-1]}u^k.uu^kx_{[(k+1)n,\infty)}. \tag{1.8}$$

Como a igualdade acima é válida para qualquer $k \in \mathbb{N}$, segue que $x = u^\infty$.

($\Leftarrow$) Se existe um n-bloco u tal que $x = u^\infty$, então aplicando σ^n obtemos $\sigma^n(x) = \sigma^n(\ldots u.uu\ldots) = \ldots uu.u\ldots = x$, ou seja, x tem período n. □

Veja que poderíamos escrever um ponto periódico como $x = u^\infty$, bem como poderíamos agrupar esse bloco que se repete em pares, e obter $x = (uu)^\infty$. Também poderíamos agrupar m cópias de u, e escrever $x = (u^m)^\infty$. Assim, se um ponto tem período n, ele também possui período mn para qualquer $m \in \mathbb{N}^*$.

1.2 Métrica

Nessa seção, veremos como definir uma métrica no shift completo. Veremos que o fato de o nosso alfabeto ser finito garante que o espaço métrico obtido é compacto. Além disso, vamos caracterizar funções contínuas entre shifts completos.

Em $\mathcal{A}^{\mathbb{Z}}$, definimos uma métrica que diz que dois pontos são próximos se eles coincidem em um bloco central.

Definição 1.12. Dados $x, y \in \mathcal{A}^{\mathbb{Z}}$, seja

$$\Delta(x,y) = \begin{cases} -1, & \text{se } x_0 \neq y_0 \\ k, & \text{se } k \text{ é o maior natural tal que } x_{[-k,k]} = y_{[-k,k]} \\ \infty, & \text{se } x = y \end{cases} \tag{1.9}$$

A métrica em $\mathcal{A}^{\mathbb{Z}}$ é dada por

$$\rho(x,y) = 2^{-\Delta(x,y)}, \tag{1.10}$$

na qual definimos $2^{-\infty} = 0$.

A seguinte Proposição é usada nas demonstrações dos resultados dessa seção.

Proposição 1.13. *Sejam $x, y \in \mathcal{A}^{\mathbb{Z}}$. Então, para qualquer $n \in \mathbb{N}$, $x_{[-n,n]} = y_{[-n,n]}$ se e somente se $\rho(x,y) \leq 2^{-n}$.*

Demonstração. ($\Rightarrow$) Se $x, y \in \mathcal{A}^{\mathbb{Z}}$ são tais que $x_{[-n,n]} = y_{[-n,n]}$, então $\rho(x,y) \leq 2^{-n}$. Para ver isso, basta notar que $n \leq \Delta(x,y)$, e usando que a exponencial em (1.10) é uma função decrescente, temos $\rho(x,y) = 2^{-\Delta(x,y)} \leq 2^{-n}$.

($\Leftarrow$) Por outro lado, se $x, y \in \mathcal{A}^{\mathbb{Z}}$ são tais que $\rho(x,y) \leq 2^{-n}$, então $2^{-\Delta(x,y)} \leq 2^{-n}$, o que implica que $\Delta(x,y) \geq n$. Logo, $x_{[-n,n]} = y_{[-n,n]}$. □

Proposição 1.14. *$(\mathcal{A}^{\mathbb{Z}}, \rho)$ é um espaço métrico.*

Demonstração. Temos que provar que ρ satisfaz as seguintes propriedades de uma métrica em $\mathcal{A}^{\mathbb{Z}}$, isto é, para quaisquer $x, y, z \in \mathcal{A}^{\mathbb{Z}}$ vale que:

(1) $\rho(x,y) = 0$ se e somente se $x = y$,

(2) $\rho(x,y) = \rho(y,x)$,

(3) $\rho(x,y) \leq \rho(x,z) + \rho(z,y)$.

Veja que (1) segue do fato de que $\rho(x,y) = 0$ se e só se $\Delta(x,y) = \infty$, o que só ocorre se e somente se $x = y$. A propriedade (2) segue da simetria das relações $x = y$, $x_{[-k,k]} = y_{[-k,k]}$ e $x_0 \neq y_0$ e de (1.9). Falta demonstrar (3).

Caso $\Delta(x,y) = \infty$, temos que $\rho(x,y) = 0$ e para qualquer $z \in \mathcal{A}^{\mathbb{Z}}$, $\rho(x,y) = 0 \leq \rho(x,z) + \rho(z,y)$.

Caso $\Delta(x,z) = \infty$, temos que $z = x$ e a desigualdade se transforma em $\rho(x,y) \leq \rho(x,y)$, que é verdadeira. O caso $\Delta(z,y) = \infty$ é análogo.

No caso em que $\Delta(x,y) = -1$, temos dois casos a analisar. Se $\Delta(x,z) = -1$, então a desigualdade vale pois $\rho(x,y) = 2 \leq 2 + \rho(z,y) = \rho(x,z) + \rho(z,y)$ para qualquer y. Se $\Delta(x,z) \neq -1$, temos que $\Delta(x,z) = k$ para algum $k \in \mathbb{N}$. Isso implica $x_{[-k,k]} = z_{[-k,k]}$ e, em particular, $x_0 = z_0$. Agora note que $\Delta(x,y) = -1$ nos dá $x_0 \neq y_0$, logo $z_0 \neq y_0$ e $\Delta(z,y) = -1$. Logo, $\rho(x,y) = 2 \leq 2^{-k} + 2 = \rho(x,z) + \rho(z,y)$, como desejado.

Por fim, suponha que $\Delta(x,y) \in \mathbb{N}$, $\Delta(x,z) = k$ e $\Delta(z,y) = m$, com $k, m \in \mathbb{N}$. Isso significa que $x_{[-k,k]} = z_{[-k,k]}$ e $y_{[-m,m]} = z_{[-m,m]}$. Tome $\ell = \min\{k, m\}$, então $x_{[-\ell,\ell]} = y_{[-\ell,\ell]}$ e logo

$$\rho(x,y) \leq 2^{-\ell} \leq 2^{-k} + 2^{-m} = \rho(x,z) + \rho(z,y), \tag{1.11}$$

como desejado. □

Agora que sabemos que $\mathcal{A}^{\mathbb{Z}}$ pode ser visto como um espaço métrico, podemos falar de conjuntos abertos e fechados.

Definição 1.15. Para $x \in \mathcal{A}^{\mathbb{Z}}$ e $r > 0$, a *bola aberta de raio r e centro x* é o conjunto

$$B_r(x) := \{y \in \mathcal{A}^{\mathbb{Z}} : \rho(x,y) < r\}. \tag{1.12}$$

Definição 1.16. Dado um bloco u sobre $\mathcal{A}$ e $k \in \mathbb{Z}$, definimos o *cilindro* $C_k(u)$ como sendo o conjunto

$$C_k(u) := \{x \in \mathcal{A}^{\mathbb{Z}} : x_{[k,k+|u|-1]} = u\}. \tag{1.13}$$

O cilindro $C_k(u)$ é o conjunto de pontos de $\mathcal{A}^{\mathbb{Z}}$ que contêm o bloco u começando na posição k.

Exemplo 1.17. *(O conjunto $C_k(u)$ é aberto e fechado).* Tome $x \in C_k(u)$, e seja

$$n = \max\{|k|, |k + |u| - 1|\}. \tag{1.14}$$

Provemos que $B_{2^{-n+1}}(x) \subseteq C_k(u)$. Dado $y \in B_{2^{-n+1}}(x)$, então $\rho(x, y) < 2^{-n+1}$, o que em particular implica que $\rho(x, y) \leq 2^{-n}$, e segue da Proposição 1.13 que $y_{[-n,n]} = x_{[-n,n]}$. Mas observando que $u = x_{[k,k+|u|-1]} \subset x_{[-n,n]}$, então $y_{[k,k+|u|-1]} = u$ e logo $y \in C_k(u)$. Desta forma, $C_k(u)$ é aberto.

Usando um argumento análogo ao do parágrafo anterior, mostramos que $\mathcal{A}^{\mathbb{Z}} \backslash C_k(u)$ também é aberto: tome $x \in \mathcal{A}^{\mathbb{Z}} \backslash C_k(u)$, e seja $n = \max\{|k|, |k + |u| - 1|\}$, de maneira que $B_{2^{-n+1}}(x) \subseteq \mathcal{A}^{\mathbb{Z}} \backslash C_k(u)$. Portanto, $C_k(u)$ também é fechado. ◁

Exemplo 1.18. *(Bolas também são cilindros).* Seja $r = 2^{-k+1}$, com $k \in \mathbb{N}$. Então $B_r(x) = C_{-k}(x_{[-k,k]})$.

($\subseteq$) Para qualquer $y \in B_r(x)$ temos que $\rho(x, y) < r = 2^{-k+1}$. Como mostrado no Exemplo 1.17, segue que $y_{[-k,k]} = x_{[-k,k]}$, ou seja, $y \in C_{-k}(x_{[-k,k]})$.

($\supseteq$) Se $y \in C_{-k}(x_{[-k,k]})$ então $y_{[-k,k]} = x_{[-k,k]}$. Logo $\rho(y, x) \leq 2^{-k} < 2^{-k+1}$, ou seja, $y \in B_r(x)$. ◁

Como podemos enxergar a convergência de sequências de pontos do shift completo? Usando os índices sobrescritos, considere $\{x^{(n)}\}_{n\in\mathbb{N}}$ uma sequência em $\mathcal{A}^{\mathbb{Z}}$, e digamos que ela converge para x. Isso significa que para qualquer $\epsilon > 0$ existe um $N \in \mathbb{N}$ tal que $\rho(x^{(n)}, x) < \epsilon$, para todo$n \geq N$. Mas perceba que dado um $\epsilon > 0$, podemos encontrar um $k \in \mathbb{N}$ tal que $2^{-k} < \epsilon$. Portanto, a sequência converge se, para qualquer $k \in \mathbb{N}$, existe um $N \in \mathbb{N}$ tal que $\rho(x^{(n)}, x) \leq 2^{-k}$, para todo $n \geq N$. Além disso, a condição $\rho(x^{(n)}, x) \leq 2^{-k}$ equivale a dizer que $x^{(n)}_{[-k,k]} = x_{[-k,k]}$. Finalmente, podemos dizer que uma sequência $\{x^{(n)}\}_{n\in\mathbb{N}}$ em $(\mathcal{A}^{\mathbb{Z}}, \rho)$ converge para x se para qualquer $k \in \mathbb{N}$ existe um $N \in \mathbb{N}$ tal que para todo $n \geq N$, $x^{(n)}_{[-k,k]} = x_{[-k,k]}$.

Como nosso alfabeto é finito, então o conjunto de todos os k-blocos também é finito, para qualquer $k \in \mathbb{N}^*$. Usando esse fato, e a noção de convergência do parágrafo anterior, podemos provar que o espaço $(\mathcal{A}^{\mathbb{Z}}, \rho)$ é compacto.

Proposição 1.19. *O espaço métrico $(\mathcal{A}^{\mathbb{Z}}, \rho)$ é compacto.*

Demonstração. Considere uma sequência $\{x^{(n)}\}_{n\in\mathbb{N}}$ qualquer. Sem perda de generalidade, podemos supor $\mathcal{A} = \{1, \ldots, r\}$. Vamos construir uma subsequência convergente usando um argumento conhecido como diagonal de Cantor.

Primeiramente veja que podemos particionar o conjunto dos índices seguintes subconjuntos

$$P_0^j = \{n \in \mathbb{N} : x_0^{(n)} = j\}, \quad j = 1, \ldots, r. \tag{1.15}$$

Como $\mathbb{N}$ é infinito e $\mathbb{N} = P_0^1 \cup \ldots \cup P_0^r$, pelo princípio da casa de pombos, pelo menos um desses conjuntos P_0^j deve ser infinito. Isso significa que deve existir um $j_0 \in \{1, \ldots, r\}$ tal que $P_0^{j_0}$ é um conjunto infinito de índices. Denote $N_0 = P_0^{j_0}$. Obtemos assim uma subsequência $\{x^{(n)}\}_{n \in N_0}$ da sequência original na qual todos os elementos possuem a mesma coordenada 0.

Veja que o conjunto de todos os 3-blocos sobre $\mathcal{A}$ é finito: de fato, possui r^3 elementos. Digamos que $\mathcal{A}^3 = \{u_1, u_2, \ldots, u_{r^3}\}$. Então, particionamos o conjunto N_0 nos conjuntos

$$P_1^j = \{n \in N_0 : x_{[-1,1]}^{(n)} = u_j\}, \quad j = 1, \ldots, r^3. \tag{1.16}$$

Como N_0 é infinito e $N_0 = P_1^1 \cup \ldots \cup P_1^{r^3}$, pelo princípio da casa de pombos, pelo menos um destes conjuntos P_1^j deve ser infinito. Assim, existe um j_1 tal que $P_1^{j_1}$ contém infinitos elementos. Denotamos por $N_1 = P_1^{j_1}$. Perceba que $N_1 \subseteq N_0$. Dessa forma, construímos uma subsequência $\{x^{(n)}\}_{n \in N_1}$ da sequência anterior na qual todos os pontos possuem o mesmo 3-bloco central.

Seguimos adiante com essa construção: no k-ésimo passo, listamos todos os $(2k+1)$-blocos sobre $\mathcal{A}$ como $\mathcal{A}^{2k+1} = \{u_1, \ldots, u_{r^{2k+1}}\}$. Então, particionamos o conjunto N_{k-1} obtida no passo anterior em $N_{k-1} = P_k^1 \cup \ldots \cup P_k^{r^{2k+1}}$, em que

$$P_k^j = \{n \in N_{k-1} : x_{[-k,k]}^{(n)} = u_j\}, \quad j = 1, \ldots, r^{2k+1}. \tag{1.17}$$

Pelo princípio da casa de pombos, existe um j_k tal que $P_k^{j_k}$ tem infinitos elementos. Denotamos $N_k = P_k^{j_k}$, e perceba que $N_k \subseteq N_{k-1}$. Assim, obtemos uma subsequência $\{x^{(n)}\}_{n \in N_k}$ na qual todos os $(2k+1)$-blocos centrais são idênticos.

Pela construção acima, $N_0 \supseteq N_1 \supseteq N_2 \supseteq \ldots$ e para quaisquer $k \in \mathbb{N}$, $\ell, m \in N_k$, temos que $x_{[-k,k]}^{(\ell)} = x_{[-k,k]}^{(m)}$.

Finalmente, escolhemos $n_0 \in N_0$, $n_1 \in N_1$ com $n_1 > n_0$ (que existe pois N_1 é infinito), $n_2 \in N_2$ com $n_2 > n_1$ (que existe pois N_2 é infinito), e em geral $n_k \in N_k$ com $n_k > n_{k-1}$ (que existe pois N_k é infinito). A subsequência convergente é $\{x^{(n_k)}\}_{k \in \mathbb{N}}$, e o ponto para o qual ela converge é o ponto x tal que

$$x_{[-k,k]} = x_{[-k,k]}^{(n_k)}. \tag{1.18}$$

Note que pela equação (1.18) e pela Proposição 1.13, temos que para qualquer $k \in \mathbb{N}$, $\rho(x^{(n_k)}, x) \leq 2^{-k}$. Para mostrar que $\{x^{(n_k)}\}_{k \in \mathbb{N}}$ converge para x, seja $\epsilon > 0$ qualquer, e tome um $K \in \mathbb{N}$ tal que $2^{-K} < \epsilon$. Note que para qualquer $k \geq K$, temos que $\rho(x^{(n_k)}, x) \leq 2^{-k} \leq 2^{-K} < \epsilon$, como desejado. □

Conforme [4, p.125], uma função é contínua se e somente se a imagem de qualquer sequência convergente é convergente. Usamos esse resultado para demonstrar a continuidade do operador de shift.

Exemplo 1.20. *(O operador de shift é uma função contínua).* Seja $\{x^{(n)}\}_{n\in\mathbb{N}}$ uma sequência em $(\mathcal{A}^{\mathbb{Z}}, \rho)$ tal que $x^{(n)} \to x$. Mostraremos que $\sigma(x^{(n)}) \to \sigma(x)$.

Seja $\epsilon > 0$ qualquer, e escolha um $k \in \mathbb{N}$ tal que $2^{-k} < \epsilon$ para facilitar as nossas contas. Como $x_n \to x$, então existe um N tal que, para todo $n \geq N$, $\rho(x^{(n)}, x) < 2^{-k}$. Isso significa que, para todo $n \geq N$, $x^{(n)}_{[-k-1,k+1]} = x_{[-k-1,k+1]}$.

Agora note que $\sigma(x^{(n)})_{[-k,k]} = x^{(n)}_{[-k+1,k+1]}$ e $\sigma(x)_{[-k,k]} = x_{[-k+1,k+1]}$. Pelo parágrafo acima, para todo $n \geq N$ vale que $x^{(n)}_{[-k+1,k+1]} = x_{[-k+1,k+1]}$, logo também vale que $\sigma(x^{(n)})_{[-k,k]} = \sigma(x)_{[-k,k]}$. Com efeito, para todo $n \geq N$, $\rho(\sigma(x^{(n)}), \sigma(x)) \leq 2^{-k} < \epsilon$, concluindo que $\sigma(x^{(n)}) \to \sigma(x)$ e, de fato, σ é uma função contínua. ◁

Lembrando que uma função é contínua se e somente se a imagem inversa de um conjunto aberto é um conjunto aberto [4, p.68], provaremos o seguinte Lema.

Lema 1.21. *Sejam $\mathcal{A}$ e $\mathcal{D}$ alfabetos quaisquer. Uma função $f : \mathcal{A}^{\mathbb{Z}} \to \mathcal{D}^{\mathbb{Z}}$ é contínua se e somente se para cada $k \in \mathbb{N}$ existe um $N \in \mathbb{N}$ tal que para qualquer $x \in \mathcal{A}^{\mathbb{Z}}$, $f(x)_{[-k,k]}$ depende apenas de $x_{[-N,N]}$.*

Demonstração. ($\Rightarrow$) Vamos mostrar que para cada $k \in \mathbb{N}$ existe um $N \in \mathbb{N}$ tal que se $z_{[-N,N]} = x_{[-N,N]}$ então $f(z)_{[-k,k]} = f(x)_{[-k,k]}$.

Tome um $f(x) \in \mathcal{D}^{\mathbb{Z}}$ e um $k \in \mathbb{N}$. Note que o cilindro

$$E = C_{-k}(f(x)_{[-k,k]}) \tag{1.19}$$

é o conjunto de todos os pontos de $\mathcal{D}^{\mathbb{Z}}$ cujo $(2k+1)$-bloco central é $f(x)_{[-k,k]}$. Como E é aberto, a continuidade de f garante que o conjunto

$$A = f^{-1}(E) \tag{1.20}$$

é um aberto de $\mathcal{A}^{\mathbb{Z}}$. Logo, de $x \in A$, sabemos que existe um raio δ da forma $\delta = 2^{-N+1}$ tal que $B_\delta(x) \subseteq A$. Agora note que $B_r(x) = C_{-N}(x_{[-N,N]})$ é o conjunto de todos os pontos de $\mathcal{A}^{\mathbb{Z}}$ cujo $(2N+1)$-bloco central é $x_{[-N,N]}$. Assim, se $z \in B_r(x)$ temos que $z_{[-N,N]} = x_{[-N,N]}$ e, além disso, $f(z) \in E$ o que garante que $f(z)_{[-k,k]} = f(x)_{[-k,k]}$.

($\Leftarrow$) Seja $f : \mathcal{A}^{\mathbb{Z}} \to \mathcal{D}^{\mathbb{Z}}$ tal que para cada $k \in \mathbb{N}$ existe um $N \in \mathbb{N}$ tal que, para todo $x \in \mathcal{A}^{\mathbb{Z}}$, $f(x)_{[-k,k]}$ depende apenas de $x_{[-N,N]}$. Seja A um aberto de $\mathcal{D}^{\mathbb{Z}}$ e considere um ponto $x \in f^{-1}(A)$ qualquer. Encontraremos um raio $r > 0$ tal que $B_r(x) \subseteq f^{-1}(A)$, mostrando que $f^{-1}(A)$ é aberto.

Por definição, $f(x) \in A$. Como A é aberto, existe um raio ϵ da forma $\epsilon = 2^{-k+1}$ tal que $B_\epsilon(f(x)) \subseteq A$. Sabemos que $B_\epsilon(f(x)) = C_{-k}(f(x)_{[-k,k]})$ é o conjunto de todos os pontos de $\mathcal{D}^{\mathbb{Z}}$ cujo $(2k+1)$-bloco central é $f(x)_{[-k,k]}$.

Por hipótese, para esse k existe um N tal que se $y \in \mathcal{A}^{\mathbb{Z}}$ é tal que $y_{[-N,N]} = x_{[-N,N]}$, então então $f(y)_{[-k,k]} = f(x)_{[-k,k]}$. Veja que isso é equivalente a dizer que se y pertence a $C_{-N}(x_{[-N,N]})$, então $f(y)$ pertence a $C_{-k}(f(x)_{[-k,k]}) = B_\epsilon(f(x))$. Definindo $r = 2^{-N+1}$, obtemos $C_{-N}(x_{[-N,N]}) = B_r(x)$. Assim, para qualquer $y \in B_r(x)$ temos que $f(y) \in B_\epsilon(f(x))$, o que prova que $B_r(x) \subseteq f^{-1}(A)$. □

1.3 Subshifts

Nesta seção, estudaremos subconjuntos de $\mathcal{A}^{\mathbb{Z}}$ chamados *subshifts*. Veremos que subshifts são exatamente os conjuntos que são fechados e invariantes pelo operador de shift.

Definição 1.22. Dado $x \in \mathcal{A}^{\mathbb{Z}}$ e w bloco sobre $\mathcal{A}$, diremos que w *ocorre* em x se existirem $i, j \in \mathbb{Z}$ tais que $w = x_{[i,j]}$, e escrevemos $w \triangleleft x$. Caso contrário, escrevemos $w \not\triangleleft x$.

Considere um conjunto $\mathcal{F}$ de blocos, os quais serão considerados os *blocos proibidos*. Então, formamos o subconjunto de pontos de $\mathcal{A}^{\mathbb{Z}}$ nos quais os blocos de $\mathcal{F}$ não ocorram.

Definição 1.23. Seja u um bloco sobre $\mathcal{A}$. Defina o subconjunto de $\mathcal{A}^{\mathbb{Z}}$ no qual u não ocorre por

$$\mathsf{X}_u := \{x \in \mathcal{A}^{\mathbb{Z}} : u \not\triangleleft x\}. \tag{1.21}$$

Para um conjunto $\mathcal{F}$ de blocos sobre $\mathcal{A}$, definimos

$$\mathsf{X}_{\mathcal{F}} := \bigcap_{w \in \mathcal{F}} \mathsf{X}_w. \tag{1.22}$$

Definição 1.24. Um *subshift* é um subconjunto W de um shift completo $\mathcal{A}^{\mathbb{Z}}$ tal que $W = \mathsf{X}_{\mathcal{F}}$ para algum conjunto $\mathcal{F}$ de blocos sobre $\mathcal{A}$.

Exemplo 1.25. Como exemplos triviais, tomando $\mathcal{F} = \emptyset$, vemos que o shift completo $\mathcal{A}^{\mathbb{Z}}$ é um subshift. Da mesma forma, tomando $\mathcal{F} = \mathcal{A}$, temos que $X = \emptyset$ é um subshift. ◁

Exemplo 1.26. Veja que dados dois subshifts X_F e X_G, sua interseção também é um subshift, uma vez que

$$\mathsf{X}_F \cap \mathsf{X}_G = \left(\bigcap_{w \in F} \mathsf{X}_w\right) \cap \left(\bigcap_{v \in G} \mathsf{X}_v\right) = \bigcap_{u \in F \cup G} \mathsf{X}_u = \mathsf{X}_{F \cup G}. \tag{1.23}$$

◁

Exemplo 1.27. Sejam $F \subseteq G$ conjuntos de palavras proibidas, então $\mathsf{X}_G \subseteq \mathsf{X}_F$, pois

$$\mathsf{X}_G = \bigcap_{w \in G} \mathsf{X}_w = \left(\bigcap_{w \in F} \mathsf{X}_w\right) \cap \left(\bigcap_{w \in G \setminus F} \mathsf{X}_w\right) \subseteq \bigcap_{w \in F} \mathsf{X}_w = \mathsf{X}_F. \tag{1.24}$$

◁

Exemplo 1.28. Tomando $\mathcal{A} = \{0, 1\}$, considere todas as sequências binárias nas quais não existam dois 1's adjacentes. Essa restrição pode ser posta tomando $\mathcal{F} = \{11\}$. Aqui, o subshift $W = \mathsf{X}_{\mathcal{F}}$ obtido é chamado de *shift da razão áurea*. ◁

Exemplo 1.29. Considere W como sendo o conjunto de todas as sequências binárias nas quais entre dois 1's existe um número par de 0's. Neste caso, tomamos

$$\mathcal{F} = \{10^{2n+1}1 : n \in \mathbb{N}\} \tag{1.25}$$

e chamamos $W = \mathsf{X}_{\mathcal{F}}$ de *shift par*. ◁

Exemplo 1.30. Considere um subconjunto $S \subseteq \mathbb{N}$. Se S for finito, defina $\mathsf{X}(S)$ como o conjunto de todas as sequências binárias nas quais o 1 ocorre infinitamente em ambas as direções e tal que o número de 0's entre quaisquer 1's sucessivos pertence ao conjunto S. Se S for infinito, removemos a exigência de que o 1 ocorra infinitamente em ambas as direções, e permitimos sequências que comecem ou terminem com infinitos zeros. Em ambos os casos, chamamos o subshift $\mathsf{X}(S)$ de *shift S-gap*.

Assim, por exemplo, o shift da razão áurea é obtido tomando $S = \{1, 2, 3, \ldots\}$, o shift par é obtido tomando $S = \{0, 2, 4, \ldots\}$ e o shift completo tomando $S = \mathbb{N}$. ◁

Veja que podemos ter $\mathcal{F}$ finito como no Exemplo 1.28 como infinito no Exemplo 1.29. Em todos os casos, $\mathcal{F}$ é um conjunto enumerável: ordene os blocos de $\mathcal{F}$ por comprimento, e perceba que para cada n, existem apenas finitos n-blocos. Também veja que podemos descrever um mesmo subshift com diferentes conjuntos $\mathcal{F}$.

Lema 1.31. *Seja $W \subseteq \mathcal{A}^{\mathbb{Z}}$ um subshift e $n \in \mathbb{N}^*$. Então existe um conjunto $\mathcal{F}$ de blocos de comprimento pelo menos n tal que $W = \mathsf{X}_{\mathcal{F}}$.*

Demonstração. Como W é um subshift, então existe um conjunto F de blocos tal que $W = \mathsf{X}_F$. Para cada k-bloco u em F, com $k < n$, considere

$$\Gamma_n(u) = \{w \in \mathcal{A}^n : u \subset w\} \tag{1.26}$$

o conjunto de todos os n-blocos que contém u. Defina

$$\Gamma_n(F) = \bigcup_{\substack{u \in F \\ |u| < n}} \Gamma_n(u) \tag{1.27}$$

e tome $\mathcal{F}$ como sendo o conjunto $\mathcal{F} = \{w \in F : |w| \geq n\} \cup \Gamma_n(F)$. Veja que por construção, todos os blocos de $\mathcal{F}$ tem comprimento pelo menos n. Vejamos que $\mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_F = \mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_{\mathcal{F}}$.

($\subseteq$) Tome $x \in \mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_F$. Então existe um bloco $w \in F$ tal que $w \triangleleft x$. Se $|w| \geq n$, então $w \in \mathcal{F}$ e $x \in \mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_{\mathcal{F}}$. Se $|w| < n$, sejam i, j tais que $w = x_{[i,j]}$. Veja que podemos "aumentar" o bloco w e considerar o bloco $u = x_{[i,i+n-1]}$. Nesse caso, $u \triangleleft x$ e $u \in \Gamma_n(w) \subseteq \mathcal{F}$, logo também temos $x \in \mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_{\mathcal{F}}$.

($\supseteq$) Tome $x \in \mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_{\mathcal{F}}$. Então existe um bloco $w \in \mathcal{F}$ tal que $w \triangleleft x$. Se $w \in F$, então $x \in \mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_F$. Se $w \notin F$, então $w \in \Gamma_n(F)$. Logo, existe um bloco $u \in F$ tal que $u \subset w$, e assim $u \triangleleft x$ o que implica $x \in \mathcal{A}^{\mathbb{Z}} \backslash \mathsf{X}_F$.

Finalmente, basta considerar o complementar da igualdade acima e concluir que $W = \mathsf{X}_F = \mathsf{X}_{\mathcal{F}}$. □

Definição 1.32. Um subconjunto X do shift completo $\mathcal{A}^{\mathbb{Z}}$ é dito ser *shift-invariante* se $\sigma(X) \subseteq X$.

Veja que se $X \subseteq \mathcal{A}^{\mathbb{Z}}$ é shit-invariante, então a restrição $\sigma|_X : X \to X$ está bem definida. Vejamos que os subshifts são shift-invariantes.

Proposição 1.33. *Para qualquer conjunto $\mathcal{F}$ de palavras proibidas vale a igualdade*

$$\sigma(\mathsf{X}_{\mathcal{F}}) = \mathsf{X}_{\mathcal{F}}. \tag{1.28}$$

Em particular, todo subshift é shift-invariante.

Demonstração. Primeiro, veja que dado um bloco w, então vale $w \triangleleft x \Leftrightarrow w \triangleleft \sigma(x)$. Isso pois $w \triangleleft x$ se existem i, j tais que $w = x_{[i,j]}$. Aplicando o operador de shift, temos que $\sigma(x)_{[i-1,j-1]} = x_{[i,j]} = w$, ou seja, $w \triangleleft \sigma(x)$. A recíproca é análoga.

Seja dado um conjunto $\mathcal{F}$ de blocos proibidos, mostraremos que $\sigma(\mathsf{X}_{\mathcal{F}}) = \mathsf{X}_{\mathcal{F}}$.

($\subseteq$) Note que um ponto x pertence a $\mathsf{X}_{\mathcal{F}}$ se e só se $\forall w \in \mathcal{F}, w \not\triangleleft x$. Pelo paragrafo anterior, isso significa que $\forall w \in \mathcal{F}, w \not\triangleleft \sigma(x)$, ou seja, $\sigma(x) \in \mathsf{X}_{\mathcal{F}}$. Assim, $\sigma(\mathsf{X}_{\mathcal{F}}) \subseteq \mathsf{X}_{\mathcal{F}}$.

($\supseteq$) Tome um ponto $x = (x_i)_{i \in \mathbb{Z}} \in \mathsf{X}_{\mathcal{F}}$. Considere $y = (x_{i-1})_{i \in \mathbb{Z}}$, e veja que $y \in \mathsf{X}_{\mathcal{F}}$, uma vez que se algum bloco proibido ocorresse em y, ele ocorreria em x. Como $x = \sigma(y)$, então $\mathsf{X}_{\mathcal{F}} \subseteq \sigma(\mathsf{X}_{\mathcal{F}})$. □

Pela igualdade acima, vemos que tanto $\sigma|_{\mathsf{X}_{\mathcal{F}}}$ quanto sua inversa estão bem definidas em um subshift. A shift-invariância é uma condição necessária, mas não suficiente para caracterizar um subshift.

Exemplo 1.34. Considere em $\{0,1\}^{\mathbb{Z}}$ o ponto $\tilde{x} = \ldots 000.1000 \ldots$, isto é, $\tilde{x}_0 = 1$ e $\tilde{x}_i = 0$ para todo $i \in \mathbb{Z} \setminus \{0\}$. Seja subconjunto X formado pela órbita de $\tilde{x}$, ou seja, $X = \{\sigma^n(\tilde{x}) : n \in \mathbb{Z}\}$. Veja que X é invariante pelo operador de shift, uma vez que

$$\sigma(X) = \{\sigma(\sigma^n(\tilde{x})) : n \in \mathbb{Z}\} = \{\sigma^{n+1}(\tilde{x}) : n \in \mathbb{Z}\} = X$$

Entretanto, se X fosse um subshift $\mathsf{X}_{\mathcal{F}}$, não poderíamos ter nenhum bloco 0^k pertencendo a $\mathcal{F}$, para qualquer $k \in \mathbb{N}^*$, já que $\tilde{x}_{[1,k]} = 0^k$. Portanto, teríamos que nenhum sub-bloco de 0^∞ pertenceria a $\mathcal{F}$. Isso implicaria que $0^\infty \in \mathsf{X}_{\mathcal{F}}$, o que não ocorre. ◁

Veja que no exemplo acima, podemos formar uma sequência $\{x^{(n)}\}_{n \in \mathbb{N}}$ de pontos de X, fazendo $x^{(n)} = \sigma^n(\tilde{x})$. Essa sequência converge para 0^∞, que é um ponto que não pertence ao conjunto X.

Proposição 1.35. *Um subshift $\mathsf{X}_{\mathcal{F}} \subseteq \mathcal{A}^{\mathbb{Z}}$ é compacto.*

Demonstração. Dada a sequência $\{x^{(n)}\}_{n\in\mathbb{N}} \subseteq \mathsf{X}_{\mathcal{F}}$, construímos uma subsequência convergente $\{x^{(n_k)}\}_{k\in\mathbb{N}}$, de maneira totalmente análoga à demonstração da Proposição 1.19. Seja x o ponto para o qual $\{x^{(n_k)}\}$ converge. Se algum bloco $u \in \mathcal{F}$ ocorresse em x, então existiriam $i, j \in \mathbb{Z}$ tais que $x_{[i,j]} = u$. Mas ai basta tomar um $k \in \mathbb{N}$ tal que $k \geq \max\{|i|, |j|\}$, de modo que $x_{[i,j]} \subset x_{[-k,k]} = x^{(n_k)}_{[-k,k]}$. Assim, u ocorre em $x^{(n_k)}$, o que é uma contradição com o fato de que $x^{(n_k)} \in \mathsf{X}_{\mathcal{F}}$. Logo, $x \in \mathsf{X}_{\mathcal{F}}$, como desejado. □

Teorema 1.36. *Um subconjunto de $\mathcal{A}^{\mathbb{Z}}$ é um subshift se e somente se é shift-invariante e compacto.*

Demonstração. ($\Rightarrow$) Pelas Proposições1.33 e 1.35, um subshift é shift-invariante e compacto.

($\Leftarrow$) Seja W um subconjunto de $\mathcal{A}^{\mathbb{Z}}$ que é shift-invariante e compacto. Um subconjunto de um espaço métrico compacto é compacto se e somente se é fechado [4, p.211]. Assim, W é fechado, ou seja, W^c é aberto.

Portanto, para cada $y \in W^c$ existe uma bola de centro em y totalmente contida em W^c. Isso significa que existe um raio $r(y)$ tal que $B_{r(y)}(y) \subseteq W^c$. Pela Proposição 1.13, podemos escolher $r(y) = 2^{-k(y)+1}$. Assim, se $x \in B_{r(y)}(y)$, então $x_{[-k(y),k(y)]} = y_{[-k(y),k(y)]}$. Defina então, para cada $y \in W^c$, o bloco $u(y) = y_{[-k(y),k(y)]}$. Seja

$$\mathcal{F} = \{u(y) : y \in W^c\} \tag{1.29}$$

e vejamos que $W = \mathsf{X}_{\mathcal{F}}$.

($\subseteq$) Tome $x \in W$, e suponha por contradição que $x \notin \mathsf{X}_{\mathcal{F}}$. Isso significa que existe um $u(y) \in \mathcal{F}$ tal que $u(y) \triangleleft x$, para algum $y \in W^c$. Por definição, existe um $i \in \mathbb{Z}$ tal que $x_{[i-k(y),i+k(y)]} = u(y)$. Considere $z = \sigma^i(x)$, que pertence a W por shift-invariância. Agora

$$z_{[-k(y),k(y)]} = \sigma^i(x)_{[-k(y),k(y)]} = x_{[i-k(y),i+k(y)]} = u(y) = y_{[-k(y),k(y)]}, \tag{1.30}$$

ou seja, $\rho(z, y) \leq 2^{-k(y)} < 2^{-k(y)+1} = r(y)$. Assim, $z \in B_{r(y)}(y) \subseteq W^c$, o que é uma contradição com o fato de que $z \in W$.

($\supseteq$) Seja $x \in \mathsf{X}_{\mathcal{F}}$, e suponha por contradição que $x \notin W$. Assim, $x \in W^c$, o que significa que $u(x) \in \mathcal{F}$. Mas uma vez que $u(x) \triangleleft x$, temos uma contradição com o fato de que $x \in \mathsf{X}_{\mathcal{F}}$. □

1.4 Linguagens

As vezes é melhor descrever um subshift através dos blocos que são permitidos em seus elementos ao invés de focar nos blocos que são proibidos.

Definição 1.37. Seja $X \subseteq \mathcal{A}^{\mathbb{Z}}$ um subconjunto de um shift completo. Denote por $\mathcal{B}_n(X)$ o conjunto de todos os n-blocos que ocorrem em pontos de X. A *linguagem de* X é o conjunto

$$\mathcal{B}(X) = \bigcup_{n=0}^{\infty} \mathcal{B}_n(X). \tag{1.31}$$

Para um bloco $w \in \mathcal{B}(X)$, dizemos que *w ocorre em X* ou *w é permitido em X*.

Exemplo 1.38. A linguagem do shift da razão áurea é

$$\{\varepsilon, 1, 0, 00, 01, 10, 000, 001, 010, 100, 101, \ldots\}.$$

◁

Pensamos no conjunto $\mathcal{B}(X)$ como o conjunto dos blocos permitidos de X. O próximo resultado mostra quais conjuntos de blocos são linguagens de algum subshift, e que estes últimos podem ser caracterizados por suas linguagens. No Teorema a seguir, usamos que $\mathcal{B}(X) \subseteq \mathcal{B}(\mathcal{A}^{\mathbb{Z}})$, de modo que $\mathcal{B}(X)^c$ denota o conjunto $\mathcal{B}(\mathcal{A}^{\mathbb{Z}}) \setminus \mathcal{B}(X)$.

Teorema 1.39.

(1) *Seja W um subshift e $\mathcal{L} = \mathcal{B}(W)$ sua linguagem. Se $w \in \mathcal{L}$, então*
   (a) *todo sub-bloco de w pertence a $\mathcal{L}$, e*
   (b) *existem blocos não vazios u e v tais que $uwv \in \mathcal{L}$.*

(2) *Se $\mathcal{L}$ é uma coleção de blocos sobre $\mathcal{A}$, então $\mathcal{L} = \mathcal{B}(W)$ para algum subshift W se e somente se $\mathcal{L}$ satisfaz a condição* (1).

(3) *Um subshift é determinado por sua linguagem, isto é, para qualquer subshift W,*

$$W = \mathsf{X}_{\mathcal{B}(W)^c}. \tag{1.32}$$

Demonstração.

(1) Se $w \in \mathcal{L} = \mathcal{B}(W)$, então existe um ponto $x \in W$ tal que w ocorre em x, ou seja, $w = x_{[i,j]}$, com $i \leq j$. Mas então todo sub-bloco de w ocorre em x, e logo pertence a $\mathcal{L}$. Da mesma forma, podemos tomar $u = x_{[k,i)}$ e $v = x_{(j,m]}$, com $k < i$ e $m > j$, de forma que $uwv = x_{[k,m]}$ ocorre em $\mathcal{L}$.

(2) Seja $\mathcal{L}$ uma coleção de blocos satisfazendo a condição (1), e seja $W = \mathsf{X}_{\mathcal{L}^c}$. Mostraremos que $\mathcal{L} = \mathcal{B}(W)$.

($\supseteq$) Tome $w \in \mathcal{B}(W)$. Então w ocorre em algum ponto $x \in \mathsf{X}_{\mathcal{L}^c}$, e por definição, $w \notin \mathcal{L}^c$. Logo, $w \in \mathcal{L}$.

($\subseteq$) Tome $w \in \mathcal{L}$. Aplicando (1b), temos que existem blocos não vazios u_1 e v_1 tais que $u_1 w v_1 \in \mathcal{L}$. Aplicando (1b) para o bloco $u_1 w v_1$, temos que existem blocos não vazios u_2, v_2 tais que $u_2 u_1 w v_1 v_2 \in \mathcal{L}$. Repetindo esse processo *ad infinitum*, obtemos um ponto $x = \ldots u_2 u_1 . w v_1 v_2 \ldots$ tal que todo sub-bloco de x pertence a $\mathcal{L}$, logo $x \in \mathsf{X}_{\mathcal{L}^c}$. Como w é um bloco de x, então $w \in \mathcal{B}(\mathsf{X}_{\mathcal{L}^c}) = \mathcal{B}(W)$.

(3) Seja W um subshift, mostraremos que $W = \mathsf{X}_{\mathcal{B}(W)^c}$.
($\subseteq$) Tome $x \in W$. Uma vez que todos os sub-blocos de x ocorrem em $\mathcal{B}(W)$, então x não contém nenhum bloco de $\mathcal{B}(W)^c$. Assim, $x \in \mathsf{X}_{\mathcal{B}(W)^c}$.
($\supseteq$) Como W é um subshift, existe um conjunto $\mathcal{F}$ tal que $W = \mathsf{X}_{\mathcal{F}}$. Se $x \in \mathsf{X}_{\mathcal{B}(W)^c}$, então todo sub-bloco de x pertence a $\mathcal{B}(W) = \mathcal{B}(\mathsf{X}_{\mathcal{F}})$. Por definição, nenhum sub-bloco de x pertence a $\mathcal{F}$, e $x \in \mathsf{X}_{\mathcal{F}} = W$.

□

Uma linguagem que possui as propriedades (1a) ou (1b) acima é dita ser *fatorial* ou *extensível*, respectivamente. A parte (3) do Teorema acima nos dá uma maneira de verificarmos se determinado ponto $x \in \mathcal{A}^{\mathbb{Z}}$ pertence ou não a um subshift W.

Corolário 1.40. *Seja W um subconjunto do $\mathcal{A}$-shift completo. Então W é um subshift se e somente para todo $x \in \mathcal{A}^{\mathbb{Z}}$ que satisfaz $x_{[i,j]} \in \mathcal{B}(W)$ para todo $i, j \in \mathbb{Z}$, temos que $x \in W$.*

Demonstração. ($\Rightarrow$) Seja W um subshift, e tome $x \in \mathcal{A}^{\mathbb{Z}}$ que satisfaz $x_{[i,j]} \in \mathcal{B}(W)$ para todo $i, j \in \mathbb{Z}$. Note que essa propriedade do ponto x garante que nenhum bloco de $\mathcal{B}(W)^c$ ocorre em x. Logo $x \in \mathsf{X}_{\mathcal{B}(W)^c}$. Como W é um subshift, a equação (1.32) nos garante que $x \in W$.

($\Leftarrow$) Considere que $W \subseteq \mathcal{A}^{\mathbb{Z}}$ é tal que para todo $x \in \mathcal{A}^{\mathbb{Z}}$ que satisfaz $x_{[i,j]} \in \mathcal{B}(W)$, para todo $i, j \in \mathbb{Z}$, tem-se $x \in W$. Vamos mostrar que $W = \mathsf{X}_{\mathcal{B}(W)^c}$, provando que W é um subshift.

($\subseteq$) Dado $x \in W$, então todo sub-bloco de x pertence a $\mathcal{B}(W)$. Portanto, nenhum bloco de $\mathcal{B}(W)^c$ ocorre em x, o que implica que $x \in \mathsf{X}_{\mathcal{B}(W)^c}$.

($\supseteq$) Tome $x \in \mathsf{X}_{\mathcal{B}(W)^c}$. Então, nenhum bloco de $\mathcal{B}(W)^c$ ocorre em x. Isso equivale a dizer que para quaisquer $i, j \in \mathbb{Z}$, $x_{[i,j]} \in \mathcal{B}(W)$. Por hipótese, temos que $x \in W$, como desejado. □

Definição 1.41. Um subshift X é dito *irredutível* se para cada par ordenado de blocos $u, v \in \mathcal{B}(X)$ existe um $w \in \mathcal{B}(X)$ tal que $uwv \in \mathcal{B}(X)$. Um subshift que não satisfaz essa condição é dito *redutível*.

Exemplo 1.42. Sejam $\mathcal{A} = \{0, 1\}$ e $\mathcal{F} = \{01, 10\}$. Então $\mathsf{X}_{\mathcal{F}} = \{0^\infty, 1^\infty\}$. Veja que

$$\mathcal{B}(\mathsf{X}_{\mathcal{F}}) = \{0^n : n \in \mathbb{N}\} \cup \{1^m : m \in \mathbb{N}\}. \tag{1.33}$$

Logo, tomando $u = 1, v = 0$, vemos que não existe nenhum $w \in \mathcal{B}(\mathsf{X}_{\mathcal{F}})$ tal que $uwv \in \mathcal{B}(\mathsf{X}_{\mathcal{F}})$, já que não existe palavra começada em 1 e terminada em 0 na linguagem de $\mathsf{X}_{\mathcal{F}}$. Este é um exemplo de shift redutível. ◁

Exemplo 1.43. Seja W shift da razão áurea do Exemplo 1.28. Sejam $u = u_1 \dots u_n, v = v_1 \dots v_k \in \mathcal{B}(W)$. Se $u_n v_1 \in \{00, 01, 10\}$, escolha $w = \varepsilon$, e veja que $uwv \in \mathcal{B}(W)$, uma vez

que o bloco proibido 11 não ocorre em nenhuma parte de uwv. Caso $u_n v_1 = 11$, escolha $w = 0$, e novamente temos que $uwv \in \mathcal{B}(W)$. Assim, esse subshift é irredutível. ◁

1.5 n-Blocos Adjacentes

Dado um alfabeto $\mathcal{A}$, podemos criar um novo alfabeto considerando os n-blocos sobre $\mathcal{A}$ como seus novos símbolos. Assim, faz sentido falar do shift completo $(\mathcal{A}^n)^{\mathbb{Z}}$. Além disso, há uma maneira de incluir os pontos de $\mathcal{A}^{\mathbb{Z}}$ em pontos de $(\mathcal{A}^n)^{\mathbb{Z}}$.

Definição 1.44. Seja $n \in \mathbb{N}^*$. Defina o *código n-bloco adjacente* $\beta_n : \mathcal{A}^{\mathbb{Z}} \to (\mathcal{A}^n)^{\mathbb{Z}}$ da seguinte maneira:

$$(\beta_n(x))_i = x_{[i,i+n-1]}. \tag{1.34}$$

Exemplo 1.45. Seja $\mathcal{A} = \{0, 1, a, b\}$. Vejamos qual é a imagem de um mesmo ponto por diferentes β_n.

$$\begin{aligned}
\beta_2(\ldots 11ab0.ab10\ldots) &= \ldots[11][1a][ab][b0][0a].[ab][b1][10]\ldots \\
\beta_3(\ldots 11ab0.ab10\ldots) &= \ldots[11a][1ab][ab0][b0a][0ab].[ab1][b10]\ldots \\
\beta_4(\ldots 11ab0.ab10\ldots) &= \ldots[11ab][1ab0][ab0a][b0ab][0ab1].[ab10]\ldots
\end{aligned}$$

Veja que o bloco na posição i em $\beta_n(x)$ é o n-bloco que começa na posição i de x. ◁

Seja $W \subseteq \mathcal{A}^{\mathbb{Z}}$ um subshift. Podemos restringir β_n de maneira que W seja seu domínio. Veja que a imagem de um ponto $x \in W$ por β_n é formada por n-blocos que ocorrem em x, logo pertencentes a $\mathcal{B}_n(W)$. Podemos então restringir o alfabeto do contra-domínio, e tomá-lo como sendo $\mathcal{B}_n(W)$. Assim, podemos definir o *código n-bloco adjacente com domínio no subshift* W, $\beta_n : W \to (\mathcal{B}_n(W))^{\mathbb{Z}}$, determinado pela mesma equação (1.34) anterior.

Definição 1.46. Seja W um subshift e $n \in \mathbb{N}^*$. O *shift n-bloco adjacente* ou a *representação em bloco adjacente* de W é a imagem $W^{[n]} = \beta_n(W)$.

Na próxima seção, demonstraremos o seguinte resultado.

Proposição 1.47. *Os shifts de blocos adjacentes de um subshift também são subshifts.*

Demonstração. Segue do Teorema 1.60. □

1.6 Morfismos

Nessa seção definiremos as funções que preservam a estrutura de um subshift.

Seja W um subshift sobre o alfabeto $\mathcal{A}$ e seja $\mathcal{D}$ um outro alfabeto. Sejam $m, n \in \mathbb{Z}$ com $-m \leq n$ e suponha que temos uma função $\Phi : \mathcal{B}_{m+n+1}(W) \to \mathcal{D}$, que chamaremos

de *mapa de blocos*, que faz corresponder a cada $(m+n+1)$-bloco permitido em W um símbolo em $\mathcal{D}$. Podemos então definir uma função $\phi : W \to \mathcal{D}^{\mathbb{Z}}$ que leva um ponto $x \in W$ em um ponto $y = \phi(x) \in \mathcal{D}^{\mathbb{Z}}$ seguindo a regra definida por Φ:

$$y_i = \phi(x)_i = \Phi(x_{[i-m,i+n]}). \tag{1.35}$$

Um esquema representando a transformação ϕ é dado na Figura 1.

Definição 1.48. Seja W um subshift sobre $\mathcal{A}$ e $\Phi : \mathcal{B}_{m+n+1}(W) \to \mathcal{D}$ um mapa de blocos. A função $\phi : W \to \mathcal{D}^{\mathbb{Z}}$ definida por (1.35) é chamada de *código de blocos com memória m e antecipação n induzido por* Φ. Denotamos $\phi = \Phi_{\infty}^{[-m,n]}$ para especificar a dependência com relação a Φ, a memória m e antecipação n.

Quando ocorrer de a imagem $\phi(X)$ estar contida em um subshift $Y \subseteq \mathcal{D}^{\mathbb{Z}}$, abusamos da notação e escrevemos $\phi : X \to Y$.

Exemplo 1.49. Seja $W \subseteq \mathcal{A}^{\mathbb{Z}}$ um subshift, e o mapa $\Phi : \mathcal{B}_2(W) \to \mathcal{A}$ dado por $\Phi(a_0a_1) = a_1$. Tome $m = 0$ e $n = 1$ e temos que $\phi = \Phi_{\infty}^{[0,1]}$ é o operador de shift σ. Outra maneira de descrevê-lo seria usando $\Psi : \mathcal{B}_1(W) \to \mathcal{A}$ dada por $\Psi(a_0) = a_0$, definir a memória $m = -1$ e antecipação $n = 1$ e teríamos $\sigma = \Psi_{\infty}^{[1,1]}$. Se, ao invés disso, definirmos a memória como sendo $m = 1$ e antecipação $n = -1$, temos a função inversa do operador de shift $\sigma^{-1} = \Psi_{\infty}^{[-1,-1]}$. ◁

Os exemplos acima mostram que um código de blocos pode ser representado de várias maneiras. Além disso, dois códigos de blocos podem ser induzidos pelo mesmo mapa de blocos, porém serem diferentes devido as suas memórias e antecipações.

Exemplo 1.50. Seja W o shift da razão áurea do Exemplo 1.28, $\mathcal{D} = \{a, b, c\}$ e $\Theta : \mathcal{B}_2(W) \to \mathcal{D}$ dado por $\Theta(00) = a, \Theta(01) = b, \Theta(10) = c$. Veja que não precisamos definir $\Theta(11)$ uma vez que $11 \notin \mathcal{B}_2(W)$. Podemos definir $\theta_1 = \Theta_{\infty}^{[0,1]}$ e temos por exemplo

$$\theta_1(\ldots 00101010.0000100\ldots) = \ldots abcbcbca.aaabca\ldots \tag{1.36}$$

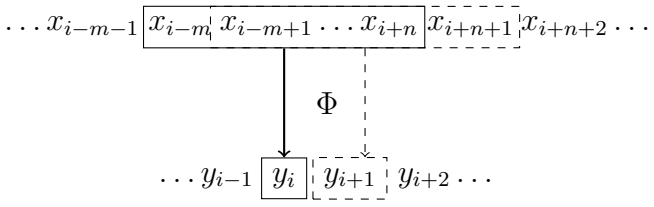
◁

Exemplo 1.51. Seja $W \subseteq \mathcal{A}^{\mathbb{Z}}$ um subshift, $\mathcal{D} = \mathcal{B}_n(W)$ e $\Phi : \mathcal{B}_n(W) \to \mathcal{D}$ dado por

$$\Phi(a_0 \ldots a_{n-1}) = [a_0 \ldots a_{n-1}] \in \mathcal{B}_n(W). \tag{1.37}$$

Então $\beta_n = \Phi_{\infty}^{[0,n-1]}$ é o código n-bloco adjacente da seção anterior. ◁

Figura 1 – Código de blocos $\phi = \Phi_{\infty}^{[-m,n]}$.



Exemplo 1.52. Seja $\phi = \Phi_\infty^{[-m,n]} : X \to Y$ um código de blocos induzido pelo mapa de blocos $\Phi : \mathcal{B}_{m+n+1}(X) \to \mathcal{D}$. Por puro formalismo, pode ser conveniente pensarmos que $\phi(x)_i$ depende de um sub-bloco maior do que $x_{[i-m,i+n]}$, porém seguindo a mesma regra Φ. Dados $M \geq m$ e $N \geq n$, definimos $\hat{\Phi} : \mathcal{B}_{M+N+1}(X) \to \mathcal{D}$ por

$$\hat{\Phi}(x_{[i-M,i+N]}) = \Phi(x_{[i-m,i+n]}). \tag{1.38}$$

Veja que $\hat{\Phi}_\infty^{[-M,N]} = \Phi_\infty^{[-m,n]}$. Esse processo de passar de Φ para $\hat{\Phi}$ é chamado de "aumentar a janela de Φ". ◁

Exemplo 1.53. Seja $\Phi : \mathcal{B}_{m+n+1}(X) \to \mathcal{D}$ um mapa de blocos. Abusando da notação, podemos definir $\Phi(v)$ para blocos v com comprimento maior que $m+n+1$. Seja $v = x_{[-m,n+k-1]}$ um $(m+n+k)$-bloco em $\mathcal{B}(X)$. Sua imagem por Φ é um k-bloco sobre $\mathcal{D}$ definida por

$$\Phi(v) = \omega_0\omega_1 \ldots \omega_{k-1} \tag{1.39}$$

em que $\omega_j = \Phi(x_{[-m+j,n+j]})$ para $j = 0, \ldots, k-1$. Essencialmente, o que fazemos é uma versão finita do processo de calcular $\phi(x)$ através de Φ. ◁

Proposição 1.54. *Sejam X e Y subshifts e σ_X, σ_Y seus operadores de shift. Se $\phi : X \to Y$ é um código de blocos, então $\phi \circ \sigma_X = \sigma_Y \circ \phi$, isto é, o seguinte diagrama comuta.*

$$\begin{array}{ccc} X & \xrightarrow{\phi} & Y \\ {\scriptstyle \sigma_X}\downarrow & & \downarrow{\scriptstyle \sigma_Y} \\ X & \xrightarrow{\phi} & Y \end{array}$$

Demonstração. Basta analisar a igualdade $\phi \circ \sigma_X = \sigma_Y \circ \phi$ coordenada a coordenada. Suponha $\phi = \Phi_\infty^{[-m,n]}$. Seja $x \in X$ qualquer, então

$$\begin{aligned} \phi \circ \sigma_X(x)_i &= \phi(\sigma_X(x))_i = \\ &= \Phi(\sigma_X(x)_{[i-m,i+n]}) = \Phi(x_{[i+1-m,i+1+n]}) = \\ &= \phi(x)_{i+1} = \sigma_Y \circ \phi(x)_i \end{aligned} \tag{1.40}$$

A igualdade vale para qualquer coordenada, então segue a identidade desejada. □

Proposição 1.55. *Códigos de blocos são funções contínuas.*

Demonstração. Aumentando a janela se necessário, seja $\phi = \Phi_\infty^{[-N,N]} : X \to \mathcal{D}^{\mathbb{Z}}$ um código de blocos. Tome $x \in X$ e seja $k \in \mathbb{N}$ qualquer. Veja que se $y \in X$ é tal que $y_{[-N-k,k+N]} = x_{[-N-k,k+N]}$ então para $-k \leq j \leq k$ temos

$$\phi(y)_j = \Phi(y_{[j-N,j+N]}) = \Phi(x_{[j-N,j+N]}) = \phi(x)_j \tag{1.41}$$

o que mostra que $y_{[-k,k]} = x_{[-k,k]}$. Pelo Lema 1.21, ϕ é contínua.

□

Exemplo 1.56. Vejamos que os períodos de um ponto são preservados por um código de blocos. Se $\phi : X \to Y$ é um código de blocos e $x \in X$ tem período n, então

$$\phi(x) = \phi(\sigma^n(x)) = \sigma^n(\phi(x)), \tag{1.42}$$

o que mostra que $\phi(x)$ também tem período n. ◁

Teorema 1.57. *Sejam X e Y subshifts. Uma função $\phi : X \to Y$ é um código de blocos se e somente se ϕ é contínua e comuta com o operador de shift, isto é, $\phi \circ \sigma_X = \sigma_Y \circ \phi$.*

Demonstração. ($\Rightarrow$) Se $\phi = \Phi_\infty^{[-m,n]}$ é um código de blocos, então é contínuo e comuta com o operador de shift pelas Proposições 1.54 e 1.55.

($\Leftarrow$) Seja $\phi : X \to Y$ uma função contínua e que comuta com o operador de shift. Do Lema 1.21, sabemos que existe um N tal que $\phi(x)_0$ depende apenas de $x_{[-N,N]}$. Defina o mapa de blocos $\Phi : \mathcal{B}_{2N+1}(X) \to \mathcal{B}_1(Y)$ por

$$\Phi(x_{[-N,N]}) = \phi(x)_0. \tag{1.43}$$

Veja que $\phi = \Phi_\infty^{[-N,N]}$, uma vez que

$$\begin{aligned} \phi(x)_i = \phi(x)_{0+i} = \sigma_Y^i(\phi(x))_0 = \phi(\sigma_X^i(x))_0 = \\ = \Phi(\sigma_X^i(x)_{[-N,N]}) = \Phi(x_{[-N+i,N+i]}). \end{aligned} \tag{1.44}$$

□

Um código de blocos $\phi : X \to Y$ que é sobrejetor é chamado de um *código fator de X para Y*; nesse caso dizemos que Y é um *fator* de X. Um código de blocos $\phi : X \to Y$ que é injetor é chamado de *imersão de X em Y*. Um caso interessante ocorre quando ϕ é *inversível*, isto é, existe um código de blocos $\psi : Y \to X$ tal que $\phi \circ \psi = \mathrm{id}_X$ e $\psi \circ \phi = \mathrm{id}_Y$.

Definição 1.58. Um código de blocos $\phi : X \to Y$ que é inversível é chamado de uma *conjugação*. Nesse caso, dizemos que X e Y são *conjugados*, e denotamos por $X \cong Y$.

Exemplo 1.59. Seja W um subshift e $W^{[n]}$ o shift n-bloco adjacente. Sabemos pelo Exemplo 1.51 que $\beta_n : W \to W^{[n]}$ é um código de blocos. Defina então o mapa de blocos $\Psi : \mathcal{A}_W^{[n]} \to \mathcal{A}$ por $\Psi([a_0 \dots a_{n-1}]) = a_0$, e considere o código de 1-bloco $\psi = \Psi_\infty^{[0,n-1]} : W^{[n]} \to W$. Veja que $\forall i \in \mathbb{Z}$,

$$\psi(\beta_n(x))_i = \psi(\beta_n(x)_i) = \psi([x_i \dots x_{i+n-1}]) = x_i, \tag{1.45}$$

ou seja, $\psi \circ \beta_n = \mathrm{id}_W$. De maneira análoga, mostra-se que $\beta_n \circ \psi = \mathrm{id}_{W^{[n]}}$, e obtemos $W \cong W^{[n]}$. ◁

Teorema 1.60. *Seja X um subshift e $\phi : X \to \mathcal{D}^{\mathbb{Z}}$ um código de blocos. Então $\phi(X)$ é um subshift.*

Demonstração. Conforme [4, p.213], a imagem de um conjunto compacto por uma função contínua é um conjunto compacto. Pela Proposição 1.35, X é compacto, e a Proposição 1.55 garante que ϕ é uma função contínua. Logo, $\phi(X)$ é compacto.

Denote por $\sigma : X \to X$ o operador de shift em X e $\sigma_{\mathcal{D}}$ o operador de shift em $\mathcal{D}^{\mathbb{Z}}$. Pela Proposição 1.54, temos que $\phi \circ \sigma = \sigma_{\mathcal{D}} \circ \phi$. Pela Proposição 1.33, temos que $X = \sigma(X)$. Logo, vale que

$$\sigma_{\mathcal{D}}(\phi(X)) = \phi(\sigma(X)) = \phi(X), \tag{1.46}$$

isto é, $\phi(X)$ é shift-invariante. Logo, pelo Teorema 1.36, $\phi(X)$ é um subshift. □

2 Subshifts do Tipo Finito

Um dos tipos mais "simples" de subshift é aquele que pode ser descrito por um conjunto finito de restrições.

2.1 Restrições finitas

Definição 2.1. Um subshift $W \subseteq \mathcal{A}^{\mathbb{Z}}$ é dito ser um *subshift do tipo finito*, abreviadamente *STF*, se existe algum conjunto finito $\mathcal{F}$ de blocos sobre $\mathcal{A}$ tal que $W = \mathsf{X}_{\mathcal{F}}$.

Exemplo 2.2. O shift da razão áurea (Exemplo 1.28) é um STF, uma vez que ele pode ser descrito por $\mathsf{X}_{\{11\}}$. $\triangleleft$

Veja que pelo Teorema 1.39, vale que $W = \mathsf{X}_{\mathcal{B}(W)^c}$ para qualquer subshift W. Exceto quando $W = \mathcal{A}^{\mathbb{Z}}$, $\mathcal{B}(W)^c$ é um conjunto infinito. Por exemplo, tomando W como o shift da razão áurea, temos

$$\mathcal{B}(W)^c = \{11, 110, 011, 111, 1100, 1101, 1110, \ldots\}. \tag{2.1}$$

Entretanto, isso não contradiz o fato de o mesmo é um STF, uma vez que pela definição, só precisa existir *algum* conjunto finito de palavras proibidas que descreva o shift.

Lema 2.3. *Se W é um STF, então existe um $N \in \mathbb{N}^*$ e um conjunto de N-blocos $\mathcal{F}$ tal que $W = \mathsf{X}_{\mathcal{F}}$.*

Demonstração. Se W é um STF sobre o alfabeto $\mathcal{A}$, então existe um conjunto finito F tal que $W = \mathsf{X}_F$. Escolha um $N \in \mathbb{N}^*$ tal que $N \geq \max\{|w| : w \in F\}$. Veja que dado $w \in \mathcal{A}^N$, ou w ocorre em W – e temos $w \in \mathcal{B}_N(W)$ – ou w não ocorre em W. Neste último caso, temos que w deve conter algum sub-bloco que pertence a F, uma vez que todo bloco de F tem comprimento menor ou igual a N. Assim, seja $\mathcal{F} = \mathcal{A}^N \backslash \mathcal{B}_N(W)$. Vejamos que $(\mathsf{X}_{\mathcal{F}})^c = W^c$.

($\supseteq$) Tome $x \in W^c = (\mathsf{X}_F)^c$. Então existe um $w \in F$ tal que $w \triangleleft x$, isto é, $w = x_{[i,i+|w|-1]}$ para algum $i \in \mathbb{Z}$. Então veja que $w \subset x_{[i,i+N-1]} \in \mathcal{A}^N$. Ocorre que $x_{[i,i+N-1]} \notin \mathcal{B}_N(W)$, uma vez que o mesmo contém o bloco proibido w como sub-bloco. Logo, $x_{[i,i+N-1]} \in \mathcal{A}^N \backslash \mathcal{B}_N(W) = \mathcal{F}$, e temos que $x \in (\mathsf{X}_{\mathcal{F}})^c$.

($\subseteq$) Tome $x \in (\mathsf{X}_{\mathcal{F}})^c$. Então existe um $w \in \mathcal{F} = \mathcal{A}^N \backslash \mathcal{B}_N(W)$ tal que $w = x_{[i,i+N-1]}$ para algum $i \in \mathbb{Z}$. Como $w \notin \mathcal{B}_N(W)$, então $w \in \mathcal{B}(W)^c$. Assim $x \in (\mathsf{X}_{B(W)^c})^c = W^c$.

Agora basta tomar o complementar da igualdade acima e temos o resultado desejado. $\square$

Assim, se $W = \mathsf{X}_{\mathcal{F}}$ é um STF, podemos supor que todos os blocos de $\mathcal{F}$ tem comprimento N. Por definição, $x \in \mathcal{A}^{\mathbb{Z}}$ pertence a W exatamente quando nenhum de deus sub-blocos ocorre em $\mathcal{F}$, ou seja, exatamente quando $x_{[i,i+N-1]} \notin \mathcal{F}, \forall i \in \mathbb{Z}$. Pelo Lema anterior, isso equivale a dizer que $x \in W$ se $x_{[i,i+N-1]} \in \mathcal{B}_N(W)$ para todo i.

Exemplo 2.4. *(O shift par não é um STF).* Suponha que o shift par do Exemplo 1.29 seja um STF. Então existiria um $N \in \mathbb{N}^*$ tal que $W = \mathsf{X}_{\mathcal{F}}$ em que $\mathcal{F} = \mathcal{A}^N \backslash \mathcal{B}_N(W)$. Seja então $x = 0^{\infty}.10^{2N+1}10^{\infty}$. Veja que vale que $x_{[i,i+N-1]} \in \mathcal{B}_N(W)$ para todo i, e logo teríamos $x \in \mathsf{X}_{\mathcal{F}}$, contradizendo a definição do shift par. ◁

Definição 2.5. Dizemos que um STF *tem memória M* se ele pode ser descrito por um conjunto de blocos proibidos com comprimento $M + 1$.

Vamos explicar essa nomenclatura. Suponha que $W = \mathsf{X}_{\mathcal{F}}$ é um STF com memória M e temos uma máquina que lê palavras e diz se elas contém ou não um sub-bloco que é proibido em W. Além disso, digamos que nossa máquina lê os símbolos sequencialmente da esquerda pra direita, um por vez. Ao ler o símbolo atual, ela só precisa relembrar os últimos M símbolos para decidir se algum bloco de F ocorreu na palavra dada ou não.

Veja que se um STF tem memória M, ele também tem memória K para qualquer $K \geq M$, basta olhar para a demonstração do Lema 2.3.

Proposição 2.6. *Se W é um STF, então existe um $M \in \mathbb{N}$ tal que W tem memória M.*

Demonstração. Seja $W = \mathsf{X}_{\mathcal{F}}$, com $\mathcal{F}$ um conjunto finito de blocos. Se $\mathcal{F} = \emptyset$, coloque $M = 0$. Caso contrário, pelo Lema 2.3, podemos supor que todos os blocos de $\mathcal{F}$ tem tamanho N, para algum $N \in \mathbb{N}^*$. Faça $M = N - 1$ e o resultado segue. □

Teorema 2.7. *Um subshift W é um STF com memória M se e somente se sempre que $uv, vw \in \mathcal{B}(W)$ com $|v| \geq M$, então $uvw \in \mathcal{B}(W)$.*

Demonstração. ($\Rightarrow$) Seja $W = \mathsf{X}_{\mathcal{F}}$ um STF com memória M, com $\mathcal{F} \subseteq \mathcal{A}^{M+1}$. Suponha que $uv, vw \in \mathcal{B}(W)$ com $|v| = n \geq M$. Então existem $x, y \in W$ tais que $x_{[-k,n-1]} = uv, y_{[0,l]} = vw$, ou seja, $x_{[0,n-1]} = y_{[0,n-1]} = v$. Seja $z = x_{(-\infty,-1]}.vy_{[n,\infty)}$, e mostremos que $z \in W$.

Se $z \notin W$, então existiria um bloco $s \in \mathcal{F}$ tal que $s = z_{[i,i+M]}$ para algum $i \in \mathbb{Z}$. Se $i < 0$, teríamos que $i + M < M \leq n$, logo $s = z_{[i,i+M]} = x_{[i,i+M]}$ o que implicaria $x \notin W$. Caso $i \geq 0$, teríamos que $s = z_{[i,i+M]} = y_{[i,i+M]}$, o que implicaria $y \notin W$. Em ambos os casos, chegaríamos em uma contradição.

Finalmente, observe que $uvw = x_{[-k,-1]}vy_{[n,l]} = z_{[-k,l]}$, logo $uvw \in \mathcal{B}(W)$.

($\Leftarrow$) Agora suponha que $W \subseteq \mathcal{A}^{\mathbb{Z}}$ é um subshift com a propriedade de que se $uv, vw \in \mathcal{B}(W)$ e $|v| \geq M$, então $uvw \in \mathcal{B}(W)$. Seja $\mathcal{F} = \mathcal{A}^{M+1} \backslash \mathcal{B}_{M+1}(W)$. Mostraremos que $W = \mathsf{X}_{\mathcal{F}}$.

($\subseteq$) Se $x \in W$, então nenhum bloco de $\mathcal{F}$ pode ocorrer em W, caso contrário teríamos um bloco fora da linguagem de W ocorrendo em W. Assim, $x \in \mathsf{X}_{\mathcal{F}}$.

($\supseteq$) Tome $x \in \mathsf{X}_{\mathcal{F}}$. Veja que pela definição de $\mathcal{F}$, $x_{[0,M]}, x_{[1,M+1]} \in \mathcal{B}(W)$. Faça $u = x_0, v = x_{[1,M]}, w = x_{M+1}$, e pela propriedade de W temos que $uvw = x_{[0,M+1]} \in \mathcal{B}(W)$, pois $|v| \geq M$. Da mesma forma, $x_{[2,M+2]} \in \mathcal{B}(W)$. Usando novamente a propriedade de W, temos que $x_{[0,1]}x_{[2,M+1]}x_{M+2} = x_{[0,M+2]} \in \mathcal{B}(W)$. Usando indução, podemos mostrar que $x_{[i,j]} \in \mathcal{B}(W)$ para todo $i, j \in \mathbb{Z}$, e pelo Corolário 1.40, $x \in W$. □

Exemplo 2.8. *(S-gap shifts que não são STF).* Seja $W = \mathsf{X}(S)$ um S-gap shift do Exemplo 1.30. Se $S \subseteq \mathbb{N}$ é um conjunto infinito tal que S^c também é infinito, então W não é um STF.

Se W fosse um STF, ele teria memória M para algum $M \in \mathbb{N}^*$. Então, escolheríamos $n \in S^c$ com $n \geq M$, que existe pois S^c é infinito. Como S também é infinito, poderíamos tomar $k \in S$ com $k > n$. Pela definição de $\mathsf{X}(S)$, $10^k1 \in \mathcal{B}(W)$. Veja que $10^n, 0^n1$ são sub-blocos de 10^k1, logo também pertencem a linguagem de W. Mas pelo Teorema anterior, teríamos que $10^n1 \in \mathcal{B}(W)$, o que seria uma contradição.

Veja que o shift par é um caso particular de subshifts desse tipo. ◁

Dado um código de blocos $\phi : X \to Y$, uma questão que surge é: se X é um STF, seria $\phi(X)$ um STF? O próximo exemplo mostra que não.

Exemplo 2.9. Seja X o shift da razão áurea e Y o shift par. Defina $\Phi : \mathcal{B}_2(X) \to \mathcal{B}_1(Y)$ por $\Phi(01) = \Phi(10) = 0, \Phi(00) = 1$ e seja $\phi = \Phi_{\infty}^{[-1,0]}$. Mostraremos que ϕ é um código fator, isto é, $\phi(X) = Y$.

($\subseteq$) Seja $10^k1 \in \mathcal{B}(\phi(X))$. Esse bloco deve ser a imagem de algum bloco da forma $0(01)^r00$. Por indução, mostra-se que $\Phi(0(01)^r00) = 10^{2r}1$. Logo, nenhum bloco proibido de Y ocorre na linguagem de $\phi(X)$, ou em outras palavras, $\mathcal{B}(Y)^c \subseteq \mathcal{B}(\phi(X))^c$. Como visto no Exemplo 1.27, temos

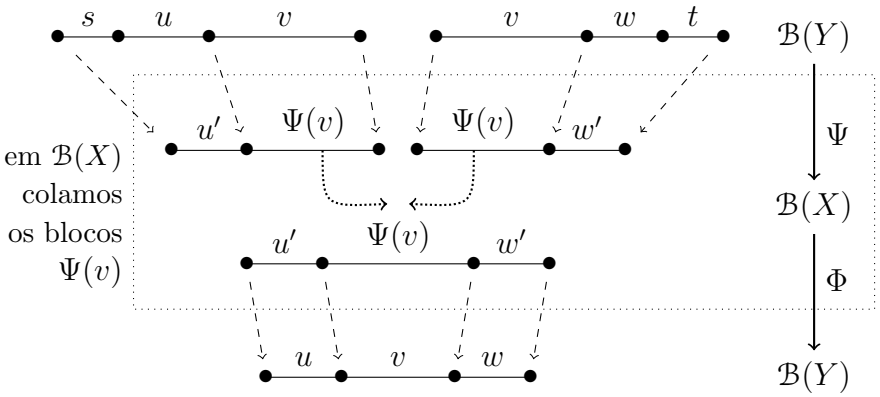
$$\phi(X) = \mathsf{X}_{\mathcal{B}(\phi(X))^c} \subseteq \mathsf{X}_{\mathcal{B}(Y)^c} = Y. \tag{2.2}$$

($\supseteq$) Seja $y \in Y$. Vamos construir $x \in X$ tal que $\phi(x) = y$.

Primeiramente, se $y = 0^\infty$, tome $x = (10)^\infty$. Caso contrário, podemos escolher $n \in \mathbb{Z}$ tal que $z = \sigma^n(y)$ e $z_0 = 1$. Construiremos x tal que $z = \phi(x)$, e logo $y = \sigma^{-n}(z) = \sigma^{-n}(\phi(x)) = \phi(\sigma^{-n}(x))$.

Como $z_0 = 1$, defina $x_{-1}x_0 = 00$. Considere o menor $n_1 \in \mathbb{N}^*$ tal que $z_{n_1} = 1$. Veja que $z_{[0,n_1]} = 10^{2k}1$, para algum $k \in \mathbb{N}$, e basta definir $x_{[0,n_1]} = (01)^k00$. Se tal n_1 não existir, defina $x_{[0,\infty)} = (01)^\infty$ e o processo termina. Se o processo não terminar, considere o menor $n_2 > n_1$ tal que $z_{n_2} = 1$. Então $z_{[n1,n_2]} = 10^{2k}1$, para algum $k \in \mathbb{N}$, e fazemos $x_{[n_1,n_2]} = (01)^k00$. Se tal n_2 não existir, fazemos $x_{[n_1,\infty)} = (01)^\infty$. O procedimento segue indutivamente.

Figura 2 – Diagrama para a demonstração do Teorema 2.10.



Para as coordenadas negativas, o processo é análogo. Considere o menor $m_1 \in \mathbb{N}^*$ tal que $z_{-m_1} = 1$. Então $z_{[-m_1,0]} = 10^{2k}1$ para algum $k \in \mathbb{N}$, e fazemos $x_{[-m_1-1,-1]} = 00(10)^k$. Caso tal m_1 não exista, fazemos $x_{(-\infty,-1]} = (10)^\infty$ e o processo termina. Se o processo não terminar, considere o menor $m_2 > m_1$ tal que $z_{-m_2} = 1$. Então $z_{[-m_2,-m_1]} = 10^{2k}1$, para algum $k \in \mathbb{N}$, e fazemos $x_{[-m_2-1,-m_1-1]} = 00(10)^k$. Se tal m_2 não existir, definimos $x_{(-\infty,-m_1-1]} = (10)^\infty$. O procedimento segue por indução. ◁

Também não podemos garantir nada se Y for um STF. Considere $\mathcal{A}$ um alfabeto com apenas uma letra a, de modo que $\mathcal{A}^\mathbb{Z} = \{a^\infty\}$. Note que, para qualquer subshift X (inclusive um X que não seja um STF), podemos construir um código de blocos $\phi : X \to \mathcal{A}^\mathbb{Z}$, tomando $\phi(x) = a^\infty$ para todo $x \in X$. O próximo teorema mostra que a propriedade de ser do tipo finito é invariante por conjugações.

Teorema 2.10. *Se $\phi : X \to Y$ é uma conjugação e X é um STF, então Y é um STF.*

Demonstração. Segundo o Teorema 2.7, precisamos encontrar um $M \in \mathbb{N}^*$ tal que se $uv, vw \in \mathcal{B}(Y)$ com $|v| \geq M$ então $uvw \in \mathcal{B}(Y)$.

Sejam $\phi = \Phi_\infty^{[-N,N]} : X \to Y$ a conjugação e $\psi = \Psi_\infty^{[-N,N]} : Y \to X$ a sua inversa. Supomos que ambas têm a mesma antecipação e memória N, pois caso contrário poderíamos aumentar a janela de ambas. Como $\phi(\psi(y)) = y$ para todo $y \in Y$, temos que $\Phi \circ \Psi : \mathcal{B}_{4N+1}(Y) \to \mathcal{B}_1(Y)$ é dada por $\Phi \circ \Psi(y_{i-2N} \dots y_{i+2N}) = y_i$.

Como X é um subshift do tipo finito, então ele tem memória $K \in \mathbb{N}^*$. Mostraremos que o subshift Y tem memória $M = K + 4N$, usando a caracterização dada pelo Teorema 2.7: dados $uv, vw \in \mathcal{B}(Y)$ com $|v| \geq M$, temos que mostrar que $uvw \in \mathcal{B}(Y)$.

Como $\mathcal{B}(Y)$ é extensível, então existem $s, t \in \mathcal{B}_{2N}(Y)$ tais que $suv, vwt \in \mathcal{B}(Y)$. Embora não saibamos se $suvwt \in \mathcal{B}(Y)$, considere z um sub-bloco de $suvwt$ com comprimento $4N + 1$. Como $|v| \geq M = 4N + K \geq 4N + 1$, então ou $z \subset suv$ ou $z \subset vwt$. Em ambos os casos, como z é um sub-bloco de um bloco da linguagem, então $z \in \mathcal{B}(Y)$. Logo, faz sentido calcular $\Phi \circ \Psi$ em qualquer sub-bloco de comprimento $4N+1$ de $suvwt$.

Denotemos $suvwt = a_{-2N} \dots a_{2N+|uvw|-1}$, de maneira que $a_0 \dots a_{|uvw|-1} = uvw$, por exemplo. Aplicando $\Phi \circ \Psi$ a $suvwt$ temos

$$\Phi \circ \Psi(suvwt) = b_0 \dots b_{|uvw|-1}, \tag{2.3}$$

em que $b_j = \Phi \circ \Psi(a_{-2N+j} \dots a_{2N+j}) = a_j$ para $j = 0, \dots, |uvw| - 1$. Portanto, temos $\Phi \circ \Psi(suvwt) = uvw$.

Conforme o esquema da Figura 2, seja $\Psi(suv) = u'\Psi(v)$, $\Psi(vwt) = \Psi(v)w'$, com $u', w' \in \mathcal{B}(X)$ e $|\Psi(v)| = |v| - 2N \geq K$, de maneira que podemos colar os dois blocos e obter que $u'\Psi(v)w' \in \mathcal{B}(X)$. Finalmente

$$uvw = \Phi \circ \Psi(suvwt) = \Phi(u'\Psi(v)w') \in \mathcal{B}(Y), \tag{2.4}$$

provando que Y é um subshift do tipo finito. □

2.2 Shifts de Grafos

Dado um grafo direcionado, podemos considerar o conjunto de todos os passeios infinitos no mesmo. Veremos que isso forma um subshift do tipo finito, conhecido como *shift aresta*. Além disso, veremos que todo shift do tipo finito pode ser codificado em um shift aresta.

Definição 2.11. Um *grafo (direcionado)* $G = (V, E, i, t)$ consiste em um conjunto finito V de *vértices* ou *estados* juntamente com um conjunto finito E de *arestas* e um par de funções $i, t : E \to V$. Cada aresta $e \in E$ tem um *estado inicial* $i(e)$ e um *estado terminal* $t(e)$. Um conjunto de arestas no qual todas possuem mesmo estado inicial e mesmo estado terminal é chamado de *arestas múltiplas*. Uma aresta e com $i(e) = t(e)$ é chamada de *laço*.

Em geral, deixamos subentendidas as funções i e t e denotamos um grafo por $G = (V, E)$. Quando estivermos trabalhando com mais de um grafo ao mesmo tempo, usaremos a notação $V(G)$ e $E(G)$ para denotar o conjunto de vértices e arestas do grafo G, bem como i_G e t_G para as funções que indicam o estado inicial e final de cada aresta.

Dado um vértice $I \in V$, denotamos $\mathbf{i}(I) = \{e \in E : i(e) = I\}$ o conjunto das arestas que saem de I, e por $\mathbf{t}(I) = \{e \in E : t(e) = I\}$ o conjunto das arestas que entram em I. O número $|\mathbf{i}(I)|$ é chamado *grau de saída* e o número $|\mathbf{t}(I)|$ é chamado *grau de entrada*.

Exemplo 2.12. No grafo da Figura 3 podemos enumerar as seguintes propriedades: $i(e) = t(e) = K$, logo e é um laço. $\mathbf{i}(J) = \{f, g, h\}$, logo o grau de saída de J é 3. Também temos os graus de entrada e saída de K iguais a $|\mathbf{t}(K)| = 2$ e $|\mathbf{i}(K)| = 5$, respectivamente. ◁

Definição 2.13. Sejam G e H dois grafos. Um *homomorfismo de G em H* consiste em duas funções $\Phi : E(G) \to E(H)$ e $\partial\Phi : V(G) \to V(H)$ tais que $\forall e \in E(G)$ vale que $i_H(\Phi(e)) = \partial\Phi(i_G(e))$ e $t_H(\Phi(e)) = \partial\Phi(t_G(e))$, isto é, o seguinte diagrama comuta

$$\begin{array}{ccccc} V(G) & \xleftarrow{i_G} & E(G) & \xrightarrow{t_G} & V(G) \\ \partial\Phi\downarrow & & \Phi\downarrow & & \downarrow\partial\Phi \\ V(H) & \xleftarrow{i_H} & E(H) & \xrightarrow{t_H} & V(H) \end{array}$$

Denotamos por $(\partial\Phi, \Phi) : G \to H$.

Quando $\partial\Phi$ e Φ são bijetoras, dizemos que $(\partial\Phi, \Phi) : G \to H$ é um *isomorfismo de grafos*, e denotamos por $G \cong H$. Nesse caso, G e H são ditos *isomorfos*. Se H é um grafo isomorfo a G, então ele é "igual" a G, exceto por uma renomeação dos vértices e arestas.

A seguir, veremos que podemos associar a cada grafo uma matriz. Para isso, vamos usar matrizes indexadas não por números naturais, mas por elementos de um conjunto qualquer. Suponha que A e B são conjuntos finitos quaisquer, cujos elementos aparecem em alguma ordem fixada. Dizemos que uma matriz M é $A \times B$ se M for uma matriz de tamanho $|A| \times |B|$ cujas entradas são representadas por $M[a, b]$, com $a \in A$ e $b \in B$. A entrada $M[a, b]$ é a entrada da i-ésima linha e j-ésima coluna da matriz M se a é o i-ésimo elemento de A e b é o j-ésimo elemento de B,

Definição 2.14. Seja G um grafo com o conjunto de vértices V. A *matriz de adjacência de G* é a matriz $V \times V$ denotada por $\mathsf{A}(G)$, e dada por

$$\mathsf{A}(G)[I, J] = |\mathbf{i}(I) \cap \mathbf{t}(J)|. \tag{2.5}$$

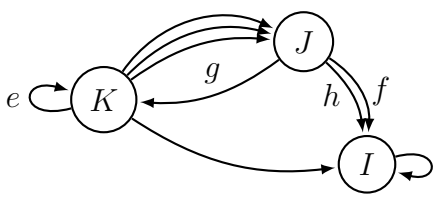
Isto é, a entrada (I, J) da matriz $\mathsf{A}(G)$ é o número de arestas que iniciam no vértice I e terminam no vértice J.

Exemplo 2.15. Considerando os vértices do grafo G da Figura 3 em ordem alfabética, temos que sua matriz de adjacência é

$$\mathsf{A}(G) = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 1 \\ 1 & 3 & 1 \end{bmatrix}. \tag{2.6}$$

◁

Figura 3 – Exemplo de grafo.



Exemplo 2.16. Suponha que A seja a matriz de adjacência de G quando listamos os vértices na ordem $V = \{I_1, I_2, \ldots, I_r\}$. O que aconteceria com A se alterássemos a ordem dos vértices?

Seja $\lambda : \{1, \ldots, r\} \to \{1, \ldots, r\}$ uma bijeção. Suponha que listamos os vértices de G na ordem $\tilde{V} = \{\tilde{I}_1, \ldots, \tilde{I}_r\}$, com $\tilde{I}_k = I_{\lambda(k)}, k = 1, \ldots, r$, e obtemos a matriz de adjacência $\tilde{A}$. Qual a relação que existe entre A e $\tilde{A}$?

Note que, por definição,

$$\tilde{A}[\tilde{I}_i, \tilde{I}_j] = \left|\mathbf{i}(\tilde{I}_i) \cap \mathbf{t}(\tilde{I}_j)\right| = \left|\mathbf{i}(I_{\lambda(i)}) \cap \mathbf{t}(I_{\lambda(j)})\right| = A[I_{\lambda(i)}, I_{\lambda(j)}]. \tag{2.7}$$

Seja então P a matriz de permutação $\tilde{V} \times V$ definida por

$$P[\tilde{I}_i, I_j] = \begin{cases} 1, & \text{se } \lambda(i) = j \\ 0, & \text{caso contrário} \end{cases} \tag{2.8}$$

Note que PA é uma matriz $\tilde{V} \times V$, cuja entrada (i, j) é $(PA)[\tilde{I}_i, I_j]$. Lembrando que os únicos elementos não nulos de P são os da forma $P[\tilde{I}_i, I_{\lambda(i)}]$, temos

$$(PA)[\tilde{I}_i, I_j] = \sum_{I_k \in V} P[\tilde{I}_i, I_k]A[I_k, I_j] = P[\tilde{I}_i, I_{\lambda(i)}]A[I_{\lambda(i)}, I_j] = A[I_{\lambda(i)}, I_j]. \tag{2.9}$$

Por outro lado, $\tilde{A}P$ é uma matriz $\tilde{V} \times V$, cuja (i, j)-ésima entrada é $(\tilde{A}P)[\tilde{I}_i, I_j]$. Denotando por $j^* = \lambda^{-1}(j)$, temos

$$\begin{aligned}(\tilde{A}P)[\tilde{I}_i, I_j] = \sum_{\tilde{I}_k \in \tilde{V}} \tilde{A}[\tilde{I}_i, \tilde{I}_k]P[\tilde{I}_k, I_j] = \tilde{A}[\tilde{I}_i, \tilde{I}_{j^*}]P[\tilde{I}_{j^*}, I_j] = \tilde{A}[\tilde{I}_i, \tilde{I}_{j^*}] \\ = A[I_{\lambda(i)}, I_{\lambda(j^*)}] = A[I_{\lambda(i)}, I_j],\end{aligned} \tag{2.10}$$

provando a igualdade entre as matrizes. ◁

Pelo exemplo anterior, se A e $\tilde{A}$ são matrizes de adjacência de grafos isomorfos, então elas são matrizes semelhantes, no sentido de que existe uma matriz P inversível tal que $\tilde{A} = PAP^{-1}$.

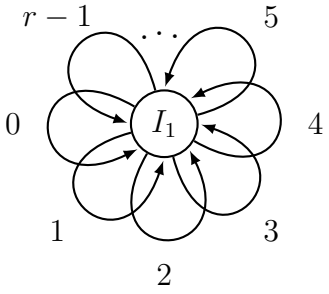
Definição 2.17. Seja $A = (a_{i,j})$ uma matriz $r \times r$ com entradas em $\mathbb{N}$. O *grafo de* A é o grafo $\mathsf{G}(A)$ com conjunto de vértices $V = \{I_1, \ldots, I_r\}$ com $a_{i,j}$ arestas com estado inicial I_i e estado final I_j.

Exemplo 2.18. Seja A a matriz 4×4 dada por

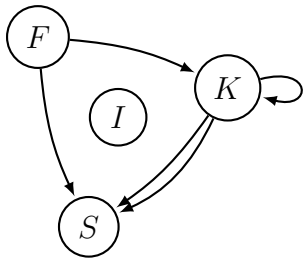
$$A = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \tag{2.11}$$

Se tomarmos o conjunto de vértices como sendo $V = \{I, S, K, F\}$, o grafo $\mathsf{G}(A)$ pode ser visto na Figura 4b. ◁

Figura 4 – Exemplos de grafos construídos a partir das matrizes de adjacência.



(a) Grafo do r-shift completo.



(b) Grafo do Exemplo 2.18.

Definição 2.19. Seja $G = (V, E)$ um grafo com matriz de adjacência A. O *shift-aresta* X_G (ou X_A) é o subshift sobre o alfabeto E especificado por

$$\mathsf{X}_G = \mathsf{X}_A := \{x \in E^{\mathbb{Z}} : t(x_i) = i(x_{i+1}) \ \forall \ i \in \mathbb{Z}\} \tag{2.12}$$

Podemos pensar um ponto $x \in \mathsf{X}_G$ como descrevendo um passeio infinito em ambas as direções no grafo G.

Exemplo 2.20. Seja $r \in \mathbb{N}^*$ e $A = [r]$ uma matriz 1×1. Então $\mathsf{G}(A)$ tem um vértice e r laços nesse vértice (veja a Figura 4a). Se nomearmos as arestas de $\mathsf{G}(A)$ como $0, 1, \ldots, r-1$, então X_G é o r-shift completo. ◁

Proposição 2.21. *Se $G = (V, E)$ é um grafo, seu shift-aresta X_G associado é um subshift do tipo finito com memória 1.*

Demonstração. Considere a coleção finita de 2-blocos sobre E

$$\mathcal{F} = \{ef \in E^2 : t(e) \neq i(f)\}. \tag{2.13}$$

De acordo com a definição 2.19, um ponto $x \in E^{\mathbb{Z}}$ está em X_G sempre que nenhum bloco de $\mathcal{F}$ ocorre em x. Logo, $\mathsf{X}_G = \mathsf{X}_{\mathcal{F}}$, e X_G é um STF. Como todos os blocos de $\mathcal{F}$ tem comprimento 2, $\mathsf{X}_{\mathcal{F}}$ tem memória 1. □

Uma palavra na linguagem do shift-aresta de um grafo G pode ser pensada como um passeio finito sobre esse grafo, o que motiva a seguinte definição.

Definição 2.22. Um *passeio* ou *caminho* $\pi = e_1 e_2 \ldots e_m$ em um grafo G é uma sequência finita de arestas $e_i \in E(G)$ tais que $t(e_i) = i(e_{i+1})$ para $i = 1, \ldots, m-1$. O *comprimento* do passeio é o número de arestas que o compõe, denotado por $|\pi|$. O caminho $\pi = e_1 \ldots e_m$ *começa* no vértice $i(\pi) = i(e_1)$ e *termina* no vértice $t(\pi) = t(e_m)$. Também dizemos que π é um caminho *de* $i(\pi)$ *até* $t(\pi)$, e que o caminho *percorre* os vértices $i(e_1), i(e_2), \ldots, i(e_m), t(e_m)$.

Um *caminho fechado* é um passeio que começa e termina no mesmo vértice. Um *ciclo* é um caminho fechado $\pi = e_1 \dots e_m$ em que $i(e_1), \dots, i(e_m)$ são vértices distintos. Também, para cada $I \in V(G)$, existe um *caminho vazio* ε_I que inicia e termina em I e tem comprimento zero.

Podemos estender a definição acima para um passeio *bi-infinito* em G: uma sequência infinita de arestas $e_i \in E(G)$ tais que $t(e_i) = i(e_{i+1})$ para $i \in \mathbb{Z}$, que é essencialmente um elemento do shift-aresta de G.

Definição 2.23. Seja $G = (V, E)$ um grafo. Dizemos que um vértice $I \in V$ é

(1) uma *fonte*, se $\mathbf{t}(I) = \emptyset$;

(2) um *sumidouro*, se $\mathbf{i}(I) = \emptyset$;

(3) um *vértice isolado*, se $\mathbf{t}(I) = \mathbf{i}(I) = \emptyset$.

Um vértice que satisfaça alguma das condições acima é chamado genericamente de *vértice encalhado*.

Veja que no grafo da Figura 4b, temos que os vértices I, S e F são, respectivamente, um vértice isolado, um sumidouro e uma fonte, e portanto, o grafo possui três vértices encalhados.

Arestas que iniciam ou terminam em um vértice encalhado nunca aparecem em X_G. Se $e \in E(G)$ ocorrem em X_G, então existe um $x \in \mathsf{X}_G$ tal que $x_1 = e$. Vejamos que $I = i(e)$ não é um vértice encalhado. Trivialmente, $e \in \mathbf{i}(I)$, logo $\mathbf{i}(I) \neq \emptyset$. Pela definição do shift-aresta, $t(x_0) = i(e) = I$, logo $\mathbf{t}(I) \neq \emptyset$. Analogamente, $J = t(e)$ não é um vértice encalhado. Vemos então que vértices encalhados não acrescentam nenhuma informação ao shift-aresta X_G.

Definição 2.24. Um grafo é dito ser *essencial* se ele não possui nenhum vértice encalhado.

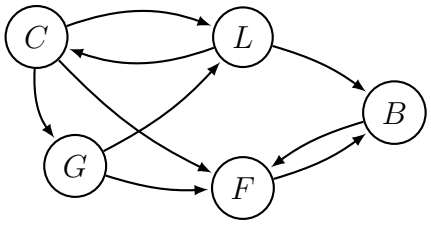
Para remover os vértices encalhados e ficar somente com a parte de G que contribui para o shift-aresta, introduzimos o conceito de *subgrafo*.

Definição 2.25. Um grafo H é dito ser *subgrafo* de G se $V(H) \subseteq V(G)$ e $E(H) \subseteq E(G)$. Nesse caso, denotamos $H \subseteq G$.

Lema 2.26. *Se G é um grafo, então existe um único subgrafo $H \subseteq G$ tal que H é essencial e $\mathsf{X}_G = \mathsf{X}_H$.*

Demonstração. Defina o subgrafo H por $E(H) = \{e \in \mathcal{B}_1(\mathsf{X}_G)\}$ e $V(H) = \{i(e) : e \in E(H)\}$. Note que para cada $e \in \mathcal{B}_1(\mathsf{X}_G)$, temos que existe um $f \in \mathcal{B}_1(\mathsf{X}_G)$ tal que $ef \in \mathcal{B}_2(\mathsf{X}_G)$. Assim, ef é um caminho em G, de maneira que $i(f) = t(e)$, o que mostra que $t(e) \in E(H)$ e, de fato, H é subgrafo de G. Como para qualquer $I \in V(H)$, I é o estado inicial de uma aresta que ocorre em X_G, então I não é um vértice encalhado. Logo, H é essencial. Resta mostrar que $\mathsf{X}_H = \mathsf{X}_G$.

Figura 5 – Exemplo de grafo redutível.



($\subseteq$) Como qualquer passeio bi-infinito em H é um passeio bi-infinito em G, então $\mathsf{X}_H \subseteq \mathsf{X}_G$.

($\supseteq$) Seja $x = (x_i)_{i\in\mathbb{Z}} \in \mathsf{X}_G$. Então, para todo $i \in \mathbb{Z}$, $x_i \in E(G)$ e $t(x_i) = i(x_{i+1})$. Isso implica que $x_i \in \mathcal{B}_1(\mathsf{X}_G)$, logo, $x_i \in E(H)$. Assim, $(x_i)_{i\in\mathbb{Z}} \in (E(H))^{\mathbb{Z}}$, com $t(x_i) = i(x_{i+1}) \forall i \in \mathbb{Z}$, isto é, $x \in \mathsf{X}_H$.

Para demonstrar a unicidade de H, suponha que exista um $H' \subseteq G$ essencial com $\mathsf{X}_{H'} = \mathsf{X}_G$. Como H' é essencial, então $E(H') = \mathcal{B}_1(\mathsf{X}_{H'})$, uma vez que toda aresta de H' ocorre em um passeio bi-infinito. Com isso, temos

$$E(H') = \mathcal{B}_1(\mathsf{X}_{H'}) = \mathcal{B}_1(\mathsf{X}_G) = E(H). \tag{2.14}$$

Portanto, se tivermos $H' \neq H$, deve ocorrer $V(H') \neq V(H)$. Mas então H' necessariamente teria que conter um vértice no qual nenhuma aresta incide, o que contradiz o fato de que H' é essencial. □

Esse resultado permite, a partir de agora, supor que todo grafo é um grafo essencial. Assim, por exemplo, um caminho π em um grafo G corresponde a uma palavra $\pi \in \mathcal{B}(\mathsf{X}_G)$.

Definição 2.27. Um grafo G é dito *irredutível* se para cada par de vértices I e J existe um caminho em G começando em I e terminando em J.

Grafos irredutíveis são chamados de *fortemente conexos* na Teoria dos Grafos. A Figura 5 mostra um exemplo de grafo redutível: não existe nenhum caminho começando em B e terminando em L.

Proposição 2.28. *Um grafo essencial é irredutível se e somente se seu shift-aresta é irredutível.*

Demonstração. ($\Rightarrow$) Considere o grafo G e π, $\tau \in \mathcal{B}(\mathsf{X}_G)$. Suponha que π termina no vértice I e τ começa no vértice J. Como G é irredutível, então existe um caminho $\omega \in \mathcal{B}(\mathsf{X}_G)$ começando em I e terminando em J. Assim, temos que $\pi\omega\tau$ é um caminho em G e logo $\pi\omega\tau \in \mathcal{B}(\mathsf{X}_G)$.

($\Leftarrow$) Sejam I e J vértices de G. Como G é essencial, então existe uma aresta e que termina em I e uma aresta f que começa em J. Como X_G é irredutível, existe um bloco ω tal que $e\omega f \in \mathcal{B}(\mathsf{X}_G)$. Assim, ω é um caminho começando em I e terminando em J o que mostra que G é irredutível. □

2.3 Representação de STF em grafos

Vimos que os shifts-aresta são um conjunto bem específico de subshift do tipo finito, uma vez que possuem memória 1. Nesta seção, veremos que usando a representação em blocos adjacentes, podemos codificar um STF de maneira a obter um shift-aresta.

Exemplo 2.29. Afirmamos que não existe grafo G tal que X_G é o shift da razão áurea. Se existisse, poderíamos supor que G é essencial. O conjunto de arestas de G seria então $E(G) = \{0, 1\}$. Caso G possuísse apenas um vértice, as arestas 0 e 1 seriam laços e X_G seria o 2-shift completo. Se G possuísse dois vértices, teríamos $t(0) = i(1)$ e $t(1) = i(0)$ o que implicaria que $\mathsf{X}_G = \{(01)^\infty, (10)^\infty\}$. ◁

Teorema 2.30. *Se X é um STF com memória $M \in \mathbb{N}$, então existe um grafo G tal que $\mathsf{X}_G = X^{[M+1]}$.*

Demonstração. Se $M = 0$, então X é um shift completo e podemos tomar o grafo do Exemplo 2.20. Caso contrário, construa o grafo G tomando o conjunto de vértices como sendo $V = \mathcal{B}_M(X)$ e o conjunto de arestas como $E = \mathcal{B}_{M+1}(X)$. A aresta $w_0 \dots w_M$ inicia no vértice $w_0 \dots w_{M-1}$ e termina no vértice $w_1 \dots w_M$. Vejamos que $\mathsf{X}_G = X^{[M+1]}$.

($\subseteq$) Veja que se $x \in \mathsf{X}_G$, então $x = (e_i)_{i\in\mathbb{Z}} \in E^{\mathbb{Z}}$ com $t(e_i) = i(e_{i+1})\forall i \in \mathbb{Z}$. Se denotarmos $e_i = w_0^{(i)} \dots w_M^{(i)}$ então a condição $t(e_i) = i(e_{i+1})$ se traduz em $w_1^{(i)} \dots w_M^{(i)} = w_0^{(i+1)} \dots w_{M-1}^{(i+1)}$. Aplicando essa igualdade para todo i, encontramos símbolos $w_i \in \mathcal{B}_1(X)$ tais que

$$x = \dots [w_{-1} \dots w_{M-1}].[w_0 \dots w_M][w_1 \dots w_{M+1}] \dots = \beta_{M+1}((w_i)_{i\in\mathbb{Z}}), \tag{2.15}$$

em que β_{M+1} é o código bloco adjacente definido na seção 1.5. Portanto, $x \in X^{[M+1]}$.

($\supseteq$) Se $x \in X^{[M+1]}$, então existe um $w = (w_i)_{i\in\mathbb{Z}} \in X$ tal que $x = \beta_{M+1}(w)$. Faça $e_i = w_i \dots w_{i+M}$ para todo $i \in \mathbb{Z}$, e veja que $x = (e_i)_{i\in\mathbb{Z}} \in E^{\mathbb{Z}}$. Observando que x satisfaz a condição $t(e_i) = i(e_{i+1})\forall i \in \mathbb{Z}$, obtemos que $x \in \mathsf{X}_G$. □

Exemplo 2.31. Sendo X o shift da razão áurea, então o grafo G tal que $\mathsf{X}_G = X^{[2]}$ obtido pelo processo descrito no Teorema 2.30 é o grafo da Figura 6. ◁

Da mesma maneira que podemos recodificar X e formar $X^{[N]}$, podemos tomar um grafo G e formar um grafo $G^{[N]}$.

Figura 6 – Grafo obtido do shift da razão áurea.

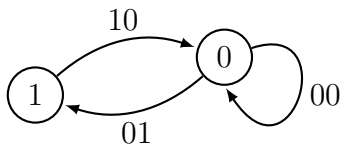
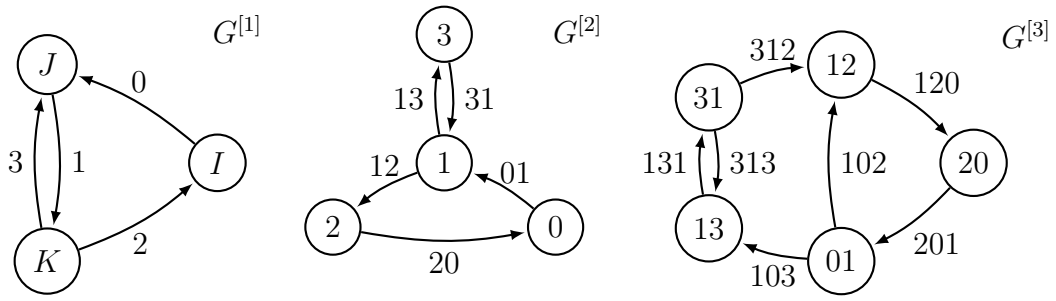


Figura 7 – Exemplo de grafo N-aresta adjacente.



Definição 2.32. Seja $G = (V, E)$ um grafo. Para $N \geq 2$, definimos o *grafo N-aresta adjacente* $G^{[N]}$ tendo o conjunto de vértices $V(G^{[N]}) = \mathcal{B}_{N-1}(\mathsf{X}_G)$ e conjunto de arestas $E(G^{[N]}) = \mathcal{B}_N(\mathsf{X}_G)$. Uma aresta $e = e_1 \dots e_N$ inicia no vértice $i(e) = e_1 \dots e_{N-1}$ e termina no vértice $t(e) = e_2 \dots e_N$. Para $N = 1$, definimos $G^{[1]} = G$.

Na Figura 7, temos um grafo G e seus grafos N-aresta adjacente, para $N = 1, 2, 3$.

Proposição 2.33. *Para qualquer grafo G vale que $(\mathsf{X}_G)^{[N]} = \mathsf{X}_{G^{[N]}}$.*

Demonstração. Veja que um símbolo de $(\mathsf{X}_G)^{[N]}$ é um caminho de tamanho N em G, e estes últimos são exatamente os símbolos de $\mathsf{X}_{G^{[N]}}$. Um passeio bi-infinito em $\mathsf{X}_{G^{[N]}}$ equivale a um ponto de $(\mathsf{X}_G)^{[N]}$ e vice-versa. □

Vejamos agora que, para $N \geq 2$, a matriz de adjacência de $G^{[N]}$ apresenta uma propriedade interessante, que dá origem a uma nova maneira de descrever um subshift.

Proposição 2.34. *Seja G um grafo, $N \geq 2$, e $B = \mathsf{A}(G^{[N]})$. Então as entradas de B são apenas 0 e 1.*

Demonstração. Sejam $I = a_1 \dots a_{N=1}$ e $J = b_1 \dots b_{N-1}$ vértices quaisquer de $G^{[N]}$.

Se $a_2 \dots a_{N-1} = b_1 \dots b_{N-2}$, então existe uma aresta $a_1 \dots a_{N-1} b_{N-1}$ com estado inicial I e estado terminal J. Veja que se existisse outra aresta $c_1 \dots c_N$ iniciando e terminando nesses mesmos vértices, teríamos que $c_1 \dots c_{N-1} = a_1 \dots a_{N-1}$ e $c_2 \dots c_N = b_1 \dots b_{N-1}$, ou seja, $c_1 \dots c_N = a_1 \dots a_{N-1} b_{N-1}$. Isso prova a unicidade da aresta de I para J, e nesse caso temos $B_{I,J} = 1$.

Se $a_2 \dots a_{N-1} \neq b_1 \dots b_{N-2}$, então não existe nenhuma aresta de I para J, o que implica que $B[I, J] = 0$. □

Uma matriz cujas entradas pertencem a $\{0, 1\}$ é chamada de *matriz binária*. Um grafo G cuja matriz de adjacência é uma matriz binária contém no máximo 1 aresta entre cada par de vértices. Assim, um passeio infinito em G pode ser descrito pela sequência de vértices visitados. Isso permite uma nova construção de um subshift, usando matrizes binárias.

Definição 2.35. Seja B uma matriz binária $r \times r$. O *shift-vértice* $\widehat{\mathsf{X}}_B$ é o subshift sobre o alfabeto $\mathcal{A} = \{1, \ldots, r\}$ definido por

$$\widehat{\mathsf{X}}_B = \{(x_i)_{i\in\mathbb{Z}} \in \mathcal{A}^{\mathbb{Z}} : B[x_i, x_{i+1}] = 1 \forall i \in \mathbb{Z}\} \tag{2.16}$$

Se G é um grafo cuja matriz de adjacência é uma matriz binária, definimos $\widehat{\mathsf{X}}_G := \widehat{\mathsf{X}}_{\mathsf{A}(G)}$.

Exemplo 2.36. Se tomamos

$$B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}, \tag{2.17}$$

o shift-vértice obtido é o shift da razão áurea. ◁

Proposição 2.37. *Shifts-vértice são subshifts do tipo finito com memória 1.*

Demonstração. Seja $X = \widehat{\mathsf{X}}_B$ um shift-vértice sobre o alfabeto $\mathcal{A} = \{1, \ldots, r\}$, definido pela matriz binária B. Considere $\mathcal{F} = \{ij \in \mathcal{A}^2 : B[i, j] = 0\}$. É trivial notar que $X = \mathsf{X}_{\mathcal{F}}$. □

Teorema 2.38. *Considerando uma renomeação dos símbolos quando necessário, temos que*

(1) *Todo subshift do tipo finito com memória* 1 *é um shifts-vértice.*
(2) *Todo shift-aresta é um shift-vértice.*
(3) *Se X é um shift do tipo finito com memória M, então $X^{[M]}$ é um shift-vértice. De fato, existe um grafo G tal que $X^{[M]} = \widehat{\mathsf{X}}_G$ e $X^{[M+1]} = \mathsf{X}_G$.*

Demonstração.

(1) Pela Proposição 2.37, temos que shifts-vértice são STF com memória 1. Da mesma forma, seja $\mathsf{X}_{\mathcal{F}}$ um STF com memória 1 sobre o alfabeto $\mathcal{A} = \{1, \ldots, r\}$. Considere a matriz binária $B = (b_{i,j})_{r\times r}$, definida por

$$b_{i,j} = \begin{cases} 0, & \text{se } ij \in \mathcal{F} \\ 1, & \text{se } ij \notin \mathcal{F} \end{cases} \tag{2.18}$$

Vejamos que $\mathsf{X}_{\mathcal{F}} = \widehat{\mathsf{X}}_B$.
($\subseteq$) Seja $x \in \mathsf{X}_{\mathcal{F}}$. Veja que para qualquer $i \in \mathbb{Z}$, temos que $x_i x_{i+1} \notin \mathcal{F}$, logo $B[x_i, x_{i+1}] = 1$, e portanto $x \in \widehat{\mathsf{X}}_B$.
($\supseteq$) Se $x \in \widehat{\mathsf{X}}_B$, então para todo $i \in \mathbb{Z}$ temos que $B[x_i, x_{i+1}] = 1$. Logo $x_i x_{i+1} \notin \mathcal{F}, \forall i \in \mathbb{Z}$, o que mostra que $x \in \mathsf{X}_{\mathcal{F}}$.

(2) Pela Proposição 2.21, temos que todo shift-aresta é um STF com memória 1. O resultado segue do item (1).

(3) Do Teorema 2.30, temos que existe um grafo G tal que $X^{[M+1]} = \mathsf{X}_G$. Veja que $B = \mathsf{A}(G)$ é uma matriz binária, por um argumento análogo à demonstração da Proposição 2.34. Mostraremos então que $\widehat{\mathsf{X}}_B = X^{[M]}$.

($\supseteq$) Seja $x = (u_i)_{i\in\mathbb{Z}} \in X^{[M]}$, em que $u_i = [x_i \dots x_{i+M-1}], \forall i \in \mathbb{Z}$, e cada x_j é um símbolo do alfabeto de X. Note que cada u_i representa um vértice de G, e também vale que $B[u_i, u_{i+1}] = 1$, pois existe a aresta $x_i \dots x_{i+M}$ que inicia em u_i e termina em u_{i+1}. Assim, $x \in \widehat{\mathsf{X}}_B$.
($\subseteq$) Seja $x = (u_i)_{i\in\mathbb{Z}} \in \widehat{\mathsf{X}}_B$. Sabemos que, para qualquer i, $u_i \in V(G) = \mathcal{B}_M(X)$ e que existe exatamente uma aresta em G que começa em u_i e termina em u_{i+1}. Isso significa que se $u_i = x_i \dots x_{i+M-1}$, então temos que $u_{i+1} = x_{i+1} \dots x_{i+M}$, para um único x_{i+M} no alfabeto. Repetindo o argumento para todo i, temos que $x = \beta_M(x_i) \in X^{[M]}$.

□

2.4 Separação de estados

Dado um grafo G qualquer, podemos aplicar um processo conhecido como separação de estados, e obter um novo grafo H tal que $\mathsf{X}_H \cong \mathsf{X}_G$.

Seja $G = (V, E)$ um grafo, que por hora vamos supor sem laços, e $I \in V$ um estado fixado. Particione o conjunto das arestas que iniciam em I em dois subconjuntos disjuntos não-vazios, chamados $\mathbf{i}(I)_1$ e $\mathbf{i}(I)_2$.

Construímos o grafo H da seguinte maneira. Os vértices de H são os mesmos de G, exceto que o vértice I é substituído por dois novos vértices, que chamaremos de I^1 e I^2, isto é,

$$V(H) = (V \backslash \{I\}) \cup \{I^1, I^2\}. \tag{2.19}$$

Agora note que, uma vez que G não tem laços, as arestas de G podem ser particionadas como

$$E = \mathbf{i}(I)_1 \dot\cup \mathbf{i}(I)_2 \dot\cup \mathbf{t}(I) \dot\cup W, \tag{2.20}$$

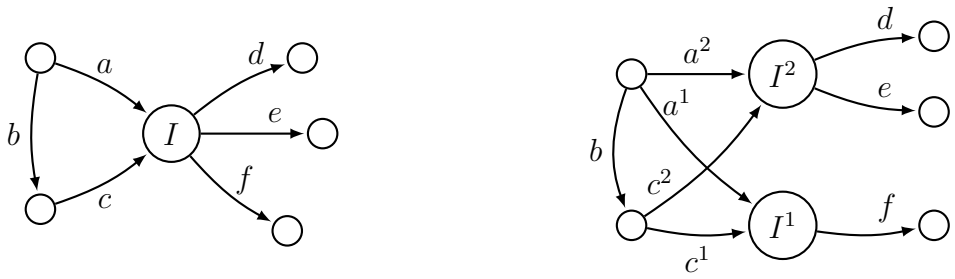
em que W são as arestas que não estão nos outros conjuntos. Então, construímos as arestas de H da seguinte maneira:

(1) Para cada aresta $e \in W$, inclua uma aresta $e \in E(H)$ com mesmo nome, iniciando e terminando nos mesmos estados.
(2) Para cada aresta $e \in \mathbf{i}(I)_j$, inclua uma aresta $e \in E(H)$ com mesmo nome, iniciando em I^j e terminando em $t(e)$, para $j = 1, 2$.
(3) Para cada aresta $e \in \mathbf{t}(I)$, inclua duas arestas $e^1, e^2 \in E(H)$ tais que e^j inicia em $i(e)$ e termina em I^j, para $j = 1, 2$.

Esse processo é chamado de *separação de estados elementar de G no vértice I*.

Exemplo 2.39. Considere o grafo da figura 8a. Escolhemos o vértice I em destaque, e particionamos $i(I)$ em $i(I)_1 = \{f\}$, $i(I)_2 = \{d, e\}$. Aplicando a separação de estados elementar de G em I, obtemos o grafo H da Figura 8b. ◁

Figura 8 – Separação de estados elementar de um grafo no vértice I.



(a) Grafo G com o estado I em destaque. (b) Grafo H obtido pela da separação de estados.

Agora, construímos uma conjugação de maneira que $\mathsf{X}_G \cong \mathsf{X}_H$. Defina o mapa de blocos $\Psi : \mathcal{B}_1(\mathsf{X}_H) \to \mathcal{B}_1(\mathsf{X}_G)$ da seguinte forma: $\Psi(e^i) = e$ se $e \in \mathbf{t}(I)$ e $\Psi(f) = f$ se $f \notin \mathbf{t}(I)$. Basicamente, Ψ apaga os índices sobrescritos dos símbolos de X_H. Veremos no Teorema 2.45, que Ψ transforma caminhos de H em caminhos de G, de maneira que induz um código de blocos $\psi = \Psi_\infty : \mathsf{X}_H \to \mathsf{X}_G$.

Também definimos o mapa de blocos $\Phi : \mathcal{B}_2(\mathsf{X}_G) \to \mathcal{B}_1(\mathsf{X}_H)$ por

$$\Phi(ef) = \begin{cases} e & \text{se } e \notin \mathbf{t}(I), \\ e^1 & \text{se } e \in \mathbf{t}(I) \text{ e } f \in \mathbf{i}(I)_1, \\ e^2 & \text{se } e \in \mathbf{t}(I) \text{ e } f \in \mathbf{i}(I)_2, \end{cases} \tag{2.21}$$

Assim, Φ lê o símbolo a direita de e para decidir se coloca ou não índices sobrescritos. Novamente, o Teorema 2.45 mostra que Φ leva caminhos de G em caminhos de H, e assim induz um código de blocos $\phi = \Phi_\infty^{[0,1]} : \mathsf{X}_G \to \mathsf{X}_H$.

Além disso, provaremos que ϕ é o código inverso de ψ.

O procedimento de separação de estados geral generaliza o caso elementar que acabamos de apresentar: podemos particionar o conjunto $\mathbf{i}(I)$ em mais do que somente dois subconjuntos disjuntos, e podemos fazer esse processo ocorrer em vários estados diferentes. Além disso, podemos aplicar o processo para grafos contendo laços.

Definição 2.40. Seja $G = (V, E)$ um grafo. Para cada estado $I \in V$, particione o conjunto $\mathbf{i}(I)$ em $m(I)$ subconjuntos distintos não-vazios, e chame-os de $\mathbf{i}(I)_1, \ldots, \mathbf{i}(I)_{m(I)}$, com $m(I) \geq 1$. Denote essa partição por $\mathcal{P}_I = \{\mathbf{i}(I)_1, \ldots, \mathbf{i}(I)_{m(I)}\}$ e seja $\mathcal{P} = \bigcup_{I \in V} \mathcal{P}_I$ a partição de E determinada por essas partições de cada estado. A *partição de G usando* $\mathcal{P}$ é o grafo $G^{[\mathcal{P}]}$ que tem estados

$$V(G^{[\mathcal{P}]}) = \bigcup_{I \in V} \{I^1, \ldots, I^{m(I)}\} \tag{2.22}$$

e arestas

$$E(G^{[\mathcal{P}]}) = \bigcup_{e \in E} \{e^1, \ldots, e^{m(t(e))}\} \tag{2.23}$$

Para cada aresta e^j de $G^{[\mathcal{P}]}$, determinamos os vértices em que e^j incide da seguinte maneira. Se $I = i(e)$ e $J = t(e)$ são os vértices inicial e final de e em G, então existe um único k tal que $e \in \mathbf{i}(I)_k$. Definimos então $i(e^j) = I^k$ e $t(e^j) = J^j$.

Esse processo é chamado de *separação de estados de G*.

Veja que no caso da separação de estados elementar, temos $m(I) = 2$ e $m(J) = 1$ para todo $J \neq I$, e além disso, omitimos os sobrescritos 1 desnecessários.

Exemplo 2.41. No grafo da Figura 9a, considere as seguintes partições: $\mathbf{i}(I)_1 = \{h\}$, $\mathbf{i}(J)_1 = \{a\}$, $\mathbf{i}(J)_2 = \{b\}$, $\mathbf{i}(J)_3 = \{c\}$, $\mathbf{i}(K)_1 = \{d\}$, $\mathbf{i}(K)_2 = \{e, f\}$ e $\mathbf{i}(L)_1 = \{g\}$. O grafo particionado $G^{[P]}$ é visto na Figura 9b.

Veremos como definir uma conjugação $\phi : \mathsf{X}_G \to \mathsf{X}_{G^{[\mathcal{P}]}}$ de maneira que temos, por exemplo, $\phi(\ldots egha.cfg\ldots) = \ldots e^1g^1h^1a^3.c^2f^1g^1\ldots$ ◁

Como usamos partições dos conjuntos de arestas que saem de cada estado para construir $G^{[\mathcal{P}]}$, diremos que $G^{[\mathcal{P}]}$ é uma *partição exterior* de G. Analogamente, existe uma definição de *partição interior* na qual particionamos os conjuntos de arestas que entram em cada vértice.

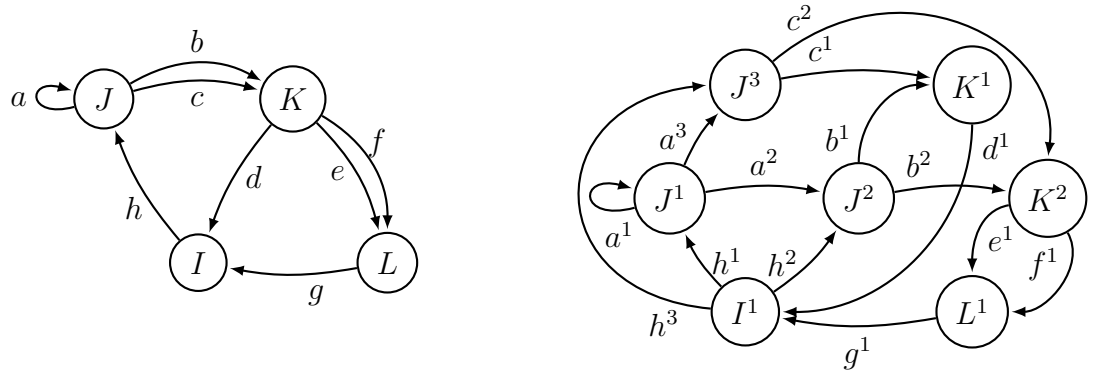
Definição 2.42. Seja $G = (V, E)$ um grafo. Para cada estado $I \in V$, particione o conjunto $\mathbf{t}(I)$ em $m(I)$ subconjuntos distintos não-vazios, e chame-os de $\mathbf{t}(I)_1, \ldots, \mathbf{t}(I)_{m(I)}$, com $m(I) \geq 1$. Denote essa partição por $\mathcal{P}_I = \{\mathbf{i}(I)_1, \ldots, \mathbf{i}(I)_{m(I)}\}$ e seja $\mathcal{P} = \bigcup_{I \in V} \mathcal{P}_I$ a partição de E determinada por essas partições de cada estado. A *partição interior de G usando* $\mathcal{P}$ é o grafo $G_{[\mathcal{P}]}$ que tem estados

$$V(G_{[\mathcal{P}]}) = \bigcup_{I \in V} \{I_1, \ldots, I_{m(I)}\} \tag{2.24}$$

e arestas

$$E(G_{[\mathcal{P}]}) = \bigcup_{e \in E} \{e_1, \ldots, e_{m(i(e))}\} \tag{2.25}$$

Figura 9 – Partição exterior, Exemplo 2.41.



(a) Grafo G.

(b) Grafo $G^{[\mathcal{P}]}$.

Para cada aresta e_k de $G_{[\mathcal{P}]}$, determinamos os vértices em que e_k incide da seguinte maneira. Se $I = i(e)$ e $J = t(e)$ são os vértices inicial e final de e em G, então existe um único j tal que $e \in \mathbf{t}(I)_j$. Definimos então $i(e_k) = I^k$ e $t(e_i) = J^j$.

Exemplo 2.43. No grafo da Figura 10a, considere a partição: $\mathbf{t}(I)_1 = \{d, g, c\}$, $\mathbf{t}(K)_1 = \{a\}$, $\mathbf{t}(K)_2 = \{f\}$, $\mathbf{t}(J)_1 = \{b, h\}$, $\mathbf{t}(J)_2 = \{e\}$. A partição interior $G_{[\mathcal{P}]}$ é visto na Figura 10b. ◁

Definição 2.44. Dizemos que um grafo H é um *particionamento* de um grafo G, e G é uma *amalgamento* de H, se H é isomorfo a alguma partição interior ou exterior de G.

Teorema 2.45. *Se um grafo H é um particionamento de um grafo G, então os shift-aresta X_G e X_H são conjugados.*

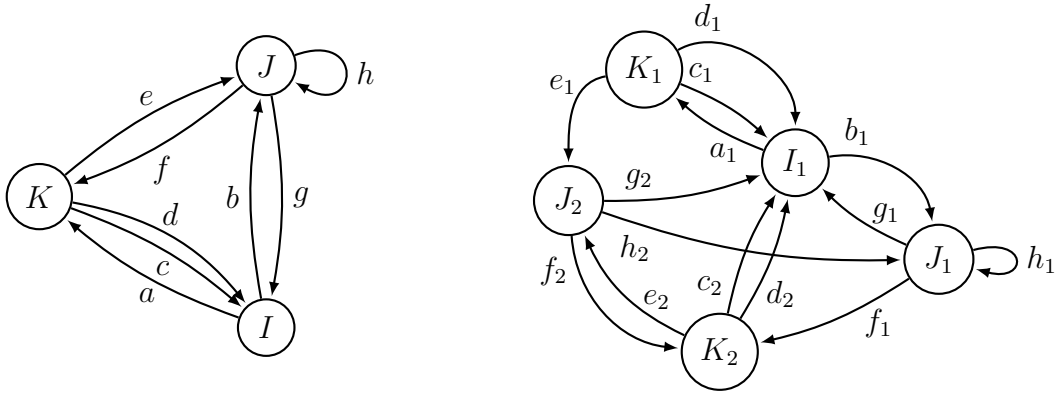
Demonstração. Provaremos para partições exteriores, já que o caso de partições interiores é análogo. Usando a notação da Definição 2.40, podemos supor que $H = G^{[\mathcal{P}]}$.

Defina o mapa de blocos $\Psi : \mathcal{B}_1(\mathsf{X}_H) \to \mathcal{B}_1(\mathsf{X}_G)$ por $\Psi(e^j) = e$. Note que se $e^j f^k \in \mathcal{B}_2(\mathsf{X}_H)$, então $t(e^j) = J^j = i(f^k)$. Com isso, sabemos que $t(e) = J = i(f)$, logo $ef \in \mathcal{B}_2(\mathsf{X}_G)$. Como X_G tem memória 1, então provamos que Ψ leva caminhos de H em caminhos de G. Logo, definindo $\psi = \Psi_\infty$, temos que $\psi(\mathsf{X}_H) \subseteq \mathsf{X}_G$, de maneira que $\psi : \mathsf{X}_H \to \mathsf{X}_G$ é um código de blocos.

Agora, defina o mapa de blocos $\Phi : \mathcal{B}_2(\mathsf{X}_G) \to \mathcal{B}_1(\mathsf{X}_H)$ por $\Phi(fe) = f^j$, em que j é tal que $e \in \mathbf{i}(I)_j$. Perceba que se $efg \in \mathcal{B}_3(\mathsf{X}_G)$ temos que existem vértices I, J e números j, k tais que $f \in \mathbf{i}(I)_j$ e $g \in \mathbf{i}(J)_k$. Assim, $\Phi(efg) = e^j f^k \in \mathcal{B}_2(\mathsf{X}_H)$, uma vez que $t(e^j) = I^j = i(f^k)$. Temos então que Φ leva caminhos de G em caminhos de H, pois X_H tem memória 1. Definindo $\phi = \Phi_\infty^{[0,1]}$, temos que $\phi(\mathsf{X}_G) \subseteq \mathsf{X}_H$, de modo que $\phi : \mathsf{X}_G \to \mathsf{X}_H$ é um código de blocos.

Sendo $x = \ldots e_{-1}.e_0 e_1 \ldots \in \mathsf{X}_G$, então $\phi(x)$ é da forma $\phi(x) = \ldots e_{-1}^{j_{-1}}.e_0^{j_0} e_1^{j_1} \ldots$, e logo $\psi(\phi(x)) = \ldots e_{-1}.e_0 e_1 \ldots = x$. Reciprocamente, se $y = \ldots e_{-1}^{j_{-1}}.e_0^{j_0} e_1^{j_1} \ldots \in \mathsf{X}_H$, então

Figura 10 – Partição interior, Exemplo 2.43.



(a) Grafo G.

(b) Grafo $G_{[\mathcal{P}]}$.

$\psi(y) = \ldots e_{-1}.e_0e_1\ldots$. Como $e_i^{j_i}e_{i+1}^{j_{i+1}} \in \mathcal{B}_2(\mathsf{X}_H)$, então $i(e_{i+1}^{j_{i+1}}) = t(e_i^{j_i}) = J^{j_i}$, em que $J = t(e_i)$. Logo, $e_{i+1} \in \mathbf{i}(J)_{j_i}$, e logo $\Phi(e_ie_{i+1}) = e_i^{j_i}$, para qualquer i. Assim, $\phi(\psi(y)) = y$ e concluímos que $\mathsf{X}_G \cong \mathsf{X}_H$. □

Os códigos de blocos da demonstração anterior recebem nomes especiais: $\phi : \mathsf{X}_G \to \mathsf{X}_H$ é chamado de *código de particionamento externo* e $\psi : \mathsf{X}_H \to \mathsf{X}_G$ é chamado de *código de amalgamento externo.*

Como a separação de estados modifica a matriz de adjacência de um grafo?

Definição 2.46. Seja G um grafo e $H = G^{[\mathcal{P}]}$ uma partição exterior de G usando $\mathcal{P}$. Sejam $U = V(G)$ e $W = V(H)$. A *matriz divisão D para $\mathcal{P}$* é a matriz $U \times W$ dadas por

$$D[I, J^k] = \begin{cases} 1, & \text{se } I = J \\ 0, & \text{se } I \neq J \end{cases} \tag{2.26}$$

A *matriz aresta E para $\mathcal{P}$* é a matriz $W \times U$ definida por

$$E[I^k, J] = |\mathbf{i}(I)_k \cap \mathbf{t}(J)| \tag{2.27}$$

Exemplo 2.47. Considere a partição do Exemplo 2.41. Temos então

$$D = \begin{array}{c} \\ I \\ J \\ K \\ L \end{array} \begin{array}{c} \begin{array}{ccccccc} I^1 & J^1 & J^2 & J^3 & K^1 & K^2 & L^1 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \qquad E = \begin{array}{c} \\ I^1 \\ J^1 \\ J^2 \\ J^3 \\ K^1 \\ K^2 \\ L^1 \end{array} \begin{array}{c} \begin{array}{cccc} I & J & K & L \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array} \tag{2.28}$$

em que indicamos explicitamente a ordem dos vértices. Note que calculando o produto DE, temos

$$DE = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 0 \\ 1 & 0 & 0 & 2 \\ 1 & 0 & 0 & 0 \end{bmatrix} \tag{2.29}$$

que é exatamente a matriz de adjacência de G. Não por acaso, se calcularmos o produto ED obteremos a matriz de adjacência de H. Esse é o conteúdo do próximo teorema. ◁

Teorema 2.48. *Seja G um grafo e $H = G^{[\mathcal{P}]}$ uma partição exterior de G usando a partição $\mathcal{P}$. Se D e E são as matrizes divisão e aresta de $\mathcal{P}$, então $DE = \mathsf{A}(G)$ e $ED = \mathsf{A}(H)$.*

Demonstração. Usando a mesma notação que a Definição 2.46, sabemos que DE é uma matriz $U \times U$, e temos

$$\begin{aligned}(DE)[I,J] &= \sum_{K^k \in W} D[I,K^k]E[K^k,J] = \sum_{k=1}^{m(I)} D[I,I^k]E[I^k,J] = \sum_{k=1}^{m(I)} E[I^k,J] \\ &= \sum_{k=1}^{m(I)} |\mathbf{i}(I)_k \cap \mathbf{t}(J)| = \left| \left(\bigcup_{k=1}^{m(I)} \mathbf{i}(I)_k \right) \cap \mathbf{t}(J) \right| \\ &= |\mathbf{i}(I) \cap \mathbf{t}(J)| = \mathsf{A}(G)[I,J], \end{aligned} \tag{2.30}$$

no qual usamos que $D[I,K^k] = 0$ quando $K \neq I$ e que os conjuntos $\mathbf{i}(I)_j$ são disjuntos. Logo, $DE = \mathsf{A}(G)$. Da mesma forma, ED é uma matriz $W \times W$ e calculando suas entradas

$$\begin{aligned}(ED)[I^\ell,J^m] &= \sum_{K \in U} E[I^\ell,K]D[K,J^m] = E[I^\ell,J]D[J,J^m] = E[I^\ell,J] \\ &= |\mathbf{i}(I)_\ell \cap \mathbf{t}(J)| = |\mathbf{i}(I^\ell) \cap \mathbf{t}(J^m)| = \mathsf{A}(H)[I^\ell,J^m]. \end{aligned} \tag{2.31}$$

Assim, $ED = \mathsf{A}(H)$, como desejado. □

2.5 Teorema da Decomposição

O objetivo dessa seção é provar que toda conjugação entre shifts-aresta é a composição de códigos de particionamento e códigos de amalgamento.

Definição 2.49. Um código de blocos $\phi : X \to Y$ é dito um *1-código* se possuir memória e antecipação 0. Uma conjugação que é um 1-código é dita uma *1-conjugação*.

A definição acima deve-se ao fato de que um 1-código lê apenas um símbolo do alfabeto por vez.

Proposição 2.50. *Dois grafos essenciais G e H são isomorfos se e somente se existe uma 1-conjugação $\phi : \mathsf{X}_G \to \mathsf{X}_H$ cuja inversa $\phi^{-1} : \mathsf{X}_H \to \mathsf{X}_G$ também é uma 1-conjugação.*

Demonstração. Lembrando da notação da Definição 2.13, suponha que $(\partial\Phi, \Phi) : G \to H$ seja o isomorfismo entre G e H, em que $\Phi : E(G) \to E(H)$ e $\partial\Phi : V(G) \to V(H)$ são bijeções.

($\Rightarrow$) Defina $\phi = \Phi_\infty^{[0,0]}$, e perceba que é um 1-código. Sua inversa é $\phi^{-1} = (\Phi^{-1})_\infty^{[0,0]}$, que também é um 1-código.

($\Leftarrow$) Seja $\phi : \mathsf{X}_G \to \mathsf{X}_H$ uma 1-conjugação cuja inversa é uma 1-conjugação. Isso significa que existe um mapa de blocos $\Phi : \mathcal{B}_1(\mathsf{X}_G) \to \mathcal{B}_1(\mathsf{X}_H)$ tal que $\phi = \Phi_\infty^{[0,0]}$. Como os grafos são essenciais, então $\mathcal{B}_1(\mathsf{X}_G) = E(G)$ e $\mathcal{B}_1(\mathsf{X}_H) = E(H)$. Logo, temos que $\Phi : E(G) \to E(H)$. Mostraremos que Φ é inversível.

Sendo ϕ^{-1} uma 1-conjugação, sabemos que existe um mapa de blocos $\Psi : E(H) \to E(G)$ tal que $\phi^{-1} = \Psi_\infty^{[0,0]}$. Dado qualquer $x \in \mathsf{X}_G$, da igualdade $\phi^{-1}(\phi(x)) = x$, temos que

$$\phi^{-1}(\phi(x))_i = \Psi(\phi(x)_i) = \Psi(\Phi(x_i)) = x_i, \tag{2.32}$$

para qualquer $i \in \mathbb{Z}$. Logo, $\Psi = \Phi^{-1}$.

Agora, definimos $\partial\Phi : V(G) \to V(H)$ da seguinte maneira: dado um vértice $J \in V(G)$, escolha uma aresta $e \in \mathbf{i}(J)$ e defina $\partial\Phi(J) = i_H(\Phi(e))$. Temos que mostrar que a nossa definição satisfaz as seguintes propriedades: (i) $\partial\Phi(J)$ não depende da escolha da aresta $e \in \mathbf{i}(J)$, (ii) $i_H(\Phi(e)) = \partial\Phi(i_G(e))$ e $t_H(\Phi(e)) = \partial\Phi(t_G(e))$, (iii) $\partial\Phi$ é bijetora.

(i) Suponha que $e, f \in \mathbf{i}(J)$. Como G é essencial, existe uma aresta g que termina em J. Assim, ge e gf são caminhos em G. Isso significa que $\Phi(g)\Phi(e)$ e $\Phi(g)\Phi(f)$ são caminhos em H. Observando os vértices em que ambos os caminhos incidem, temos

$$i_H(\Phi(e)) = t_H(\Phi(g)) = i_H(\Phi(f)), \tag{2.33}$$

e logo, $\partial\Phi(J)$ está bem definida.

(ii) Veja que fazendo $J = i_G(e)$, temos que $\partial\Phi(J) = \partial\Phi(i_G(e)) = i_H(\Phi(e))$. Agora, considere uma aresta e qualquer, e seja $J = t_G(e)$. Tome uma aresta $f \in \mathbf{i}(J)$, e sabemos que $\partial\Phi(J) = i_H(\Phi(f))$. Note que ef é um caminho em G, logo $\Phi(e)\Phi(f)$ é um caminho em H. Assim, temos que

$$t_H(\Phi(e)) = i_H(\Phi(f)) = \partial\Phi(J) = \partial\Phi(t_G(e)), \tag{2.34}$$

o que prova a segunda igualdade desejada.

(iii) Para mostrar que $\partial\Phi$ é sobrejetora, tome um vértice $K \in V(H)$. Como H é essencial, existe uma aresta f que $K = i_H(f)$. Como Φ é bijetora, existe uma aresta e em G tal que $f = \Phi(e)$. Logo, tomando $I = i_G(e)$, temos que $\partial\Phi(I) = i_H(\Phi(e)) = i_H(f) = K$. Como os conjuntos $V(G)$ e $V(H)$ são finitos, então $\partial\Phi$ é injetora, e portanto, bijetora.

Com isso, construímos o isomorfismo $(\partial\Phi, \Phi) : G \to H$, completando a nossa demonstração. □

Seja $G = (V, E)$ um grafo. Considere a partição

$$\mathcal{P} = \bigcup_{e \in E} \{e\}, \tag{2.35}$$

de modo que cada $\mathbf{i}(J)$ é particionado em conjuntos unitários de arestas, $\forall J \in V$. A partição exterior $G^{[\mathcal{P}]}$ contém um vértice para cada aresta de G, então podemos fazer uma correspondência e supor que $V(G^{[\mathcal{P}]}) = E(G) = \mathcal{B}_1(\mathsf{X}_G)$. Para cada aresta $e \in E(G)$, temos que existem $|t(e)|$ arestas em $G^{[\mathcal{P}]}$ da forma e^j, em que j é um sobrescrito qualquer. Podemos usar as próprias arestas de G como sobrescritos, de maneira que para cada

$ef \in \mathcal{B}_2(\mathsf{X}_G)$ temos uma aresta $e^f \in E(G^{[\mathcal{P}]})$. O grafo assim construído é chamado *particionamento exterior completo*.

A construção acima pode ser feita para um particionamento interior $G_{[\mathcal{P}]}$. Nesse caso, temos o *particionamento interior completo*. Note que ambos os particionamentos completos são isomorfos ao grafo 2-aresta adjacente $G^{[2]}$, usando os mapas de bloco $g^f \mapsto gf$ e $g_f \mapsto gf$.

Vimos na demonstração do Teorema 2.45 que se H é um particionamento de G, então existe um código de particionamento externo $\psi_{GH} : \mathsf{X}_G \to \mathsf{X}_H$ com memória 0 e antecipação 1. Sua inversa $\alpha_{HG} : \mathsf{X}_H \to \mathsf{X}_G$ é um 1-código chamada de código de amalgamento externo. O mesmo também pode ser dito para códigos de particionamento interno e códigos de amalgamento interno. Daqui em diante, não faremos referência ao tipo dos mesmo, chamaremos simplesmente de códigos de particionamento e códigos de amalgamento.

Pela Proposição 2.50, temos que se $\phi : \mathsf{X}_G \to \mathsf{X}_H$ é uma 1-conjugação cuja inversa é uma 1-conjugação, então existe um isomorfismo de grafos $(\partial\Phi, \Phi) : G \to H$ tal que $\phi = \Phi_\infty$. Note que ϕ é, ao mesmo tempo, um código de amalgamamento e um código de particionamento. Como ϕ consiste em basicamente trocarmos os nomes das arestas de G, dizemos que ϕ é um *código de rotulação*.

Exemplo 2.51. Seja G um grafo qualquer. Considere o operador de shift $\sigma : \mathsf{X}_G \to \mathsf{X}_G$ como uma conjugação. Vamos representar σ como a composição de códigos de particionamento e códigos de amalgamento.

Seja H o particionamento externo completo de G, de maneira que o código de particionamento $\psi_{GH} = (\Psi_{GH})_\infty^{[0,1]}$ é induzido pelo mapa de blocos tal que $\Psi_{GH}(ef) = e^f$. Seja K o particionamento interior completo de G, de forma que o código de particionamento interno seja $\psi_{GK} = (\Psi_{GK})_\infty^{[-1,0]}$, no qual $\Psi_{GK}(fg) = g_f$. O código de amalgamento interno $\alpha_{KG} = \psi_{GK}^{-1}$ é induzido pelo mapa de blocos $g_f \mapsto g$.

Note que o mapa $\Theta(e^f) = f_e$ é um isomorfismo de H em K, e induz um código de rotulação $\theta : \mathsf{X}_H \to \mathsf{X}_K$. O seguinte diagrama mostra o que acontece quando aplicamos $\alpha_{KG} \circ \theta \circ \psi_{GH}$ a um ponto de X_G.

$$\begin{array}{ccc} \ldots abc.defg \ldots & \xrightarrow{\psi_{GH}} & \ldots a^b b^c c^d.d^e e^f f^g \ldots \\ \sigma \big\downarrow & & \big\downarrow \theta \\ \ldots bcd.efg \ldots & \xleftarrow{\alpha_{KG}} & \ldots b_a c_b d_c.e_d f_e g_f \ldots \end{array}$$

Com isso vemos que $\sigma = \alpha_{KG} \circ \theta \circ \psi_{GH}$, como desejado. ◁

Lema 2.52. *Seja $\phi : \mathsf{X}_G \to \mathsf{X}_H$ uma 1-conjugação cuja inversa tem memória $m \in \mathbb{N}$ e antecipação $n \in \mathbb{N}^*$. Então existem particionamentos externos $\tilde{G}$ de G e $\tilde{H}$ de H tais que o diagrama*

$$\begin{array}{ccc} \mathsf{X}_G & \xrightarrow{\psi_{G\tilde{G}}} & \mathsf{X}_{\tilde{G}} \\ {\scriptstyle\phi}\downarrow & & \downarrow{\scriptstyle\tilde{\phi}} \\ \mathsf{X}_H & \xleftarrow{\alpha_{\tilde{H}H}} & \mathsf{X}_{\tilde{H}} \end{array}$$

comuta, e $\tilde{\phi}$ é uma 1-conjugação cuja inversa tem memória m e antecipação $n-1$.

Demonstração. Digamos que o código ϕ seja induzido por $\Phi : E(G) \to E(H)$. Seja $\tilde{H}$ o particionamento externo completo de H, de maneira que as arestas de $\tilde{H}$ são da forma h^k, para $hk \in \mathcal{B}_2(\mathsf{X}_H)$.

Para cada vértice J de G, particione $\mathbf{i}(J)$ de acordo com suas imagens por Φ:

$$\mathbf{i}(J)_h = \{g \in \mathbf{i}(J) : \Phi(g) = h\}. \tag{2.36}$$

Seja $\tilde{G}$ o particionamento externo resultante, com código de particionamento $\psi_{G\tilde{G}}$ induzido pelo mapa de blocos $fg \mapsto f^{\Phi(g)}$.

Defina $\tilde{\Phi} : E(\tilde{G}) \to E(\tilde{H})$ por $\tilde{\Phi}(g^h) = \Phi(g)^h$. Vejamos que $\tilde{\Phi}$ induz um 1-código $\tilde{\phi} : \mathsf{X}_{\tilde{G}} \to \mathsf{X}_{\tilde{H}}$. Para cada $ef \in \mathcal{B}_2(\mathsf{X}_G)$, temos que existe uma aresta $e^{\Phi(f)}$ em $\tilde{G}$. A imagem dessa aresta por $\tilde{\Phi}$ é $\tilde{\Phi}(e^{\Phi(f)}) = \Phi(e)^{\Phi(f)}$, que é uma aresta de $\tilde{H}$ já que $\Phi(e)\Phi(f)$ é um caminho em H.

O diagrama seguinte mostra a ação desses códigos.

$$\begin{array}{ccc} \ldots g_{-3}g_{-2}g_{-1}.g_0g_1g_2\ldots & \xrightarrow{\psi_{G\tilde{G}}} & \ldots g_{-3}^{h_{-2}}g_{-2}^{h_{-1}}g_{-1}^{h_0}.g_0^{h_1}g_1^{h_2}g_2^{h_3}\ldots \\ {\scriptstyle\phi}\downarrow & & \downarrow{\scriptstyle\tilde{\phi}} \\ \ldots h_{-3}h_{-2}h_{-1}.h_0h_1h_2\ldots & \xleftarrow{\alpha_{\tilde{H}H}} & \ldots h_{-3}^{h_{-2}}h_{-2}^{h_{-1}}h_{-1}^{h_0}.h_0^{h_1}h_1^{h_2}h_2^{h_3}\ldots \end{array}$$

Para mostrar que $\tilde{\phi}^{-1}$ tem a memória e antecipação corretas, vamos mostrar que $\tilde{x}_0 = \tilde{\phi}^{-1}(\tilde{y})_0$ depende apenas de $\tilde{y}_{[-m,n-1]}$, para qualquer $\tilde{y} \in \mathsf{X}_{\tilde{H}}$. Para isso, seja $y = \alpha_{\tilde{H}H}(\tilde{y})$, $x = \phi^{-1}(y)$, e perceba que $\tilde{x}_0 = x_0^{y_1}$. Veja que $y_{[-m,n]}$ está determinado por $\tilde{y}_{[-m,n-1]}$: supondo algum $\tilde{w} \in \mathsf{X}_{\tilde{H}}$ com $\tilde{w}_{[-m,n-1]} = \tilde{y}_{[-m,n-1]}$, teríamos que $\tilde{w}_{n-1} = y_{n-1}^{y_n}$, o que implica que $\tilde{w}_n$ seja da forma y_n^j para algum sobrescrito j, e quando aplicássemos o código de amalgamento teríamos $\alpha_{\tilde{H}H}(\tilde{w})_{[-m,n]} = y_{[-m,n]}$. Como x_0 está determinado por $y_{[-m,n]}$, provamos que $\tilde{\phi}^{-1}$ tem memória m e antecipação $n-1$. □

Lema 2.53. *Seja $\phi : \mathsf{X}_G \to \mathsf{X}_H$ uma 1-conjugação cuja inversa tem memória $m \in \mathbb{N}^*$ e antecipação $n \in \mathbb{N}$. Então existem particionamentos internos $\tilde{G}$ de G e $\tilde{H}$ de H tais que o diagrama*

$$\begin{array}{ccc} \mathsf{X}_G & \xrightarrow{\psi_{G\tilde{G}}} & \mathsf{X}_{\tilde{G}} \\ {\scriptstyle\phi}\downarrow & & \downarrow{\scriptstyle\tilde{\phi}} \\ \mathsf{X}_H & \xleftarrow{\alpha_{\tilde{H}H}} & \mathsf{X}_{\tilde{H}} \end{array}$$

comuta, e $\tilde{\phi}$ é uma 1-conjugação cuja inversa tem memória $m-1$ e antecipação n.

Demonstração. Análoga a demonstração do Lema 2.52. Sendo $\phi = \Phi_\infty$, basta tomar $\tilde{H}$ como sendo o particionamento interno completo de H. Para cada vértice J de G, particione $\mathbf{t}(J)$ da seguinte maneira:

$$\mathbf{t}(J)_h = \{g \in \mathbf{t}(J) : \Phi(g) = h\} \tag{2.37}$$

O grafo $\tilde{G}$ é o particionamento interno resultante, com código de particionamento $\psi_{G\tilde{G}}$ induzido por $fg \mapsto g_{\Phi(f)}$.

Defina $\tilde{\Phi} : E(\tilde{G}) \to E(\tilde{H})$ por $\tilde{\Phi}(g_h) = \Phi(g)_h$, e é fácil ver que $\tilde{\Phi}$ induz um 1-código $\tilde{\phi} : \mathsf{X}_{\tilde{G}} \to \mathsf{X}_{\tilde{H}}$.

Na demonstração de que $\tilde{\phi}^{-1}$ tem memória $m-1$ e antecipação n, basta ver que se $\tilde{y}, \tilde{w} \in \mathsf{X}_{\tilde{H}}$ são tais que $\tilde{y}_{[-m+1,n]} = \tilde{w}_{[-m+1,n]}$, então

$$\alpha_{\tilde{H}H}(\tilde{y})_{[-m,n]} = \alpha_{\tilde{H}H}(\tilde{w})_{[-m,n]}, \tag{2.38}$$

como desejado. □

Proposição 2.54. *Seja X um subshift qualquer. Então $(X^{[N]})^{[2]} \cong X^{[N+1]}$, para qualquer $N \in \mathbb{N}^*$.*

Demonstração. Pelo Exemplo 1.59, temos que $(X^{[N]})^{[2]} \cong X^{[N]} \cong X \cong X^{[N+1]}$. □

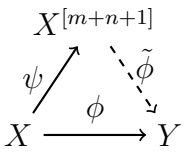
Em particular, a Proposição acima mostra que dado um shift-aresta X_G, temos uma sequência de particionamentos completos que transforma X_G em $(\mathsf{X}_G)^{[N]}$, para qualquer $N \in \mathbb{N}^*$.

Proposição 2.55. *Todo código de blocos pode ser recodificado de maneira que se torne um 1-código.*

Demonstração. Seja $\phi : X \to Y$ um código de blocos da forma $\phi = \Phi_\infty^{[-m,n]}$. Considere $\psi = \sigma^{-m} \circ \beta_{m+n+1}$, em que β_k é o código k-bloco adjacente. Note que

$$\psi(X) = \sigma^{-m} \circ \beta_{m+n+1}(X) = \sigma^{-m}(X^{[m+n+1]}) = X^{[m+n+1]}, \tag{2.39}$$

em que usamos a propriedade da shift-invariância. Logo, ψ é uma conjugação. Observando o diagrama abaixo,



podemos definir o 1-código $\tilde{\phi} = \phi \circ \psi^{-1}$ e a demonstração está completa. □

Teorema 2.56 (Teorema da Decomposição)**.** *Toda conjugação entre shifts-aresta é a composição de códigos de particionamento e códigos de amalgamento.*

Demonstração. Seja $\phi : \mathsf{X}_G \to \mathsf{X}_H$ uma conjugação. Pela Proposição 2.55, podemos supor que ϕ é 1-conjugação. Digamos que ϕ^{-1} tenha memória m e antecipação n, e aumentando a janela se necessário, supomos que $m, n \in \mathbb{N}$. Se $m = n = 0$, então ϕ é uma 1-conjugação com inversa que também é uma 1-conjugação, e pela Proposição 2.50, temos que ϕ é um código de rotulação.

Se $n \in \mathbb{N}^*$, podemos aplicar o Lema 2.52 repetidas vezes e obter uma sequência códigos de particionamento externo ψ_j, códigos de amalgamento externo α_j e 1-conjugações $\tilde{\phi}_j$, com $1 \leq j \leq n$, para os quais $\tilde{\phi}^{-1}$ tem memória m e antecipação $n-j$, como no diagrama abaixo.

$$\begin{array}{ccccccc} \mathsf{X}_G & \xrightarrow{\psi_1} & \mathsf{X}_{G_1} & \longrightarrow \cdots \xrightarrow{\psi_n} & \mathsf{X}_{G_n} \\ \phi\downarrow & & \downarrow\tilde{\phi}_1 & & \downarrow\tilde{\phi}_n \\ \mathsf{X}_H & \xleftarrow{\alpha_1} & \mathsf{X}_{H_1} & \longleftarrow \cdots \xleftarrow{\alpha_n} & \mathsf{X}_{H_n} \end{array}$$

Em particular, a inversa de $\tilde{\phi}_n$ tem memória m e antecipação 0. Se $m \in \mathbb{N}^*$, aplicamos o Lema 2.53 e obtemos uma sequência de códigos de particionamento interno ψ_{n+k}, códigos de amalgamento interno α_{n+k} e 1-conjugações $\tilde{\phi}_{n+k}$, para $1 \leq k \leq m$, tais que $\tilde{\phi}_{n+k}^{-1}$ tem memória $m-k$ e antecipação 0, como mostrado no diagrama a seguir.

$$\begin{array}{ccccccc} \mathsf{X}_{G_n} & \xrightarrow{\psi_{n+1}} & \mathsf{X}_{G_{n+1}} & \longrightarrow \cdots \xrightarrow{\psi_{n+m}} & \mathsf{X}_{G_{n+m}} \\ \tilde{\phi}_n\downarrow & & \downarrow\tilde{\phi}_{n+1} & & \downarrow\tilde{\phi}_{n+m} \\ \mathsf{X}_{H_n} & \xleftarrow{\alpha_{n+1}} & \mathsf{X}_{H_{n+1}} & \longleftarrow \cdots \xleftarrow{\alpha_{n+m}} & \mathsf{X}_{H_{n+m}} \end{array}$$

Note que $\tilde{\phi}_{n+m}^{-1}$ tem memória 0 e antecipação 0, logo é um código de rotulação. Colando esses diagramas, temos o seguinte.

$$\begin{array}{ccccccccc} \mathsf{X}_G & \xrightarrow{\psi_1} & \cdots & \xrightarrow{\psi_n} & \mathsf{X}_{G_n} & \xrightarrow{\psi_{n+1}} & \cdots & \xrightarrow{\psi_{n+m}} & \mathsf{X}_{G_{n+m}} \\ \phi\downarrow & & & & \downarrow\tilde{\phi}_n & & & & \downarrow\tilde{\phi}_{n+m} \\ \mathsf{X}_H & \xleftarrow{\alpha_1} & \cdots & \xleftarrow{\alpha_n} & \mathsf{X}_{H_n} & \xleftarrow{\alpha_{n+1}} & \cdots & \xleftarrow{\alpha_{n+m}} & \mathsf{X}_{H_{n+m}} \end{array}$$

Pelo diagrama, temos que

$$\phi = \alpha_1 \circ \ldots \circ \alpha_{n+m} \circ \tilde{\phi}_{n+m} \circ \psi_{n+m} \circ \ldots \circ \psi_1, \tag{2.40}$$

que é uma composição de códigos de particionamento e códigos de amalgamento, como desejado. □

3 Subshifts Sóficos

Suponha que temos um grafo cujas arestas são rotuladas com símbolos de um alfabeto $\mathcal{A}$. Um caminho bi-infinito nesse grafo dá origem a um ponto do shift completo $\mathcal{A}^{\mathbb{Z}}$ se lermos os rótulos de suas arestas. O conjunto de todos os pontos obtidos dessa maneira é chamado de *subshift sófico*.

3.1 Apresentações de Subshifts Sóficos

Definição 3.1. Um *grafo rotulado* $\mathcal{G}$ é um par $(G, \mathcal{L})$, em que G é um grafo com conjunto de arestas E e $\mathcal{L} : E \to \mathcal{A}$ é uma função chamada de *rotulação*, que faz corresponder a cada aresta $e \in E$ um símbolo $\mathcal{L}(e)$ do alfabeto $\mathcal{A}$. Dizemos que G é o *grafo subjacente* de $\mathcal{G}$. O grafo rotulado é *irredutível* se seu grafo subjacente o for.

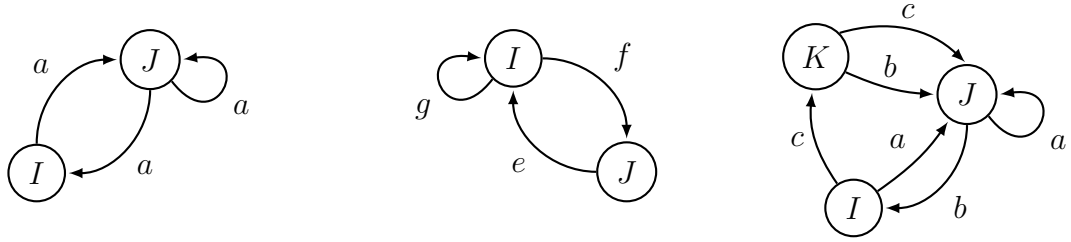
Exemplo 3.2. Na Figura 11a temos o grafo rotulado $\mathcal{G} = (G, \mathcal{L})$, no qual $\mathcal{L}(e) = a$ para toda aresta e de G. No grafo rotulado $\mathcal{H} = (H, \mathcal{L})$ da Figura 11b, temos que $\mathcal{L} : E(H) \to E(H)$ é a função identidade que associa a cada aresta seu próprio nome. O grafo rotulado da Figura 11c mostra o caso mais geral, no qual o mesmo rótulo é dado para várias arestas diferentes. ◁

De maneira análoga ao capítulo anterior, podemos representar um grafo rotulado por uma matriz de adjacência.

Definição 3.3. A *matriz de adjacência simbólica* de um grafo rotulado $\mathcal{G} = (G, \mathcal{L})$, denotada por $\mathsf{A}_{\mathcal{G}}$, é a matriz $E(G) \times E(G)$ tal que a $[I, J]$-ésima entrada é igual a "soma formal" dos rótulos das arestas que iniciam em I e terminam em J, ou $\emptyset$ se nenhuma aresta inicia em I e termina em J. Mais precisamente,

$$\mathsf{A}_{\mathcal{G}}[I, J] = \begin{cases} \mathcal{L}(e_1) + \ldots + \mathcal{L}(e_n), & \text{se } \mathbf{i}(I) \cap \mathbf{t}(J) = \{e_1, \ldots, e_n\} \\ \emptyset, & \text{se } \mathbf{i}(I) \cap \mathbf{t}(J) = \emptyset \end{cases} \tag{3.1}$$

Figura 11 – Exemplos de grafos rotulados.



(a) Grafo rotulado $\mathcal{G}$. (b) Grafo rotulado $\mathcal{H}$. (c) Grafo rotulado $\mathcal{J}$.

Exemplo 3.4. Listamos abaixo as matrizes de adjacência simbólica para os grafos rotulados da Figura 11.

$$\mathsf{A}_{\mathcal{G}} = \begin{bmatrix} \emptyset & a \\ a & a \end{bmatrix} \qquad \mathsf{A}_{\mathcal{H}} = \begin{bmatrix} g & f \\ e & \emptyset \end{bmatrix} \qquad \mathsf{A}_{\mathcal{J}} = \begin{bmatrix} \emptyset & a & c \\ b & a & \emptyset \\ \emptyset & \emptyset & b+c \end{bmatrix} \tag{3.2}$$

◁

Também podemos definir homomorfismos de grafos rotulados, se impusermos uma condição extra.

Definição 3.5. Sejam $\mathcal{G} = (G, \mathcal{L}_G)$ e $\mathcal{H} = (H, \mathcal{L}_H)$ grafos rotulados. Um *homomorfismo de grafos rotulados* de $\mathcal{G}$ em $\mathcal{H}$ é um homomorfismo que grafos $(\partial\Phi, \Phi) : G \to H$ tal que $\mathcal{L}_H(\Phi(e)) = \mathcal{L}_G(e)$ para toda aresta $e \in E(G)$. Nesse caso escrevemos $(\partial\Phi, \Phi) : \mathcal{G} \to \mathcal{H}$. Se $(\partial\Phi, \Phi) : G \to H$ é um isomorfismo de grafos, então temos um *isomorfismo de grafos rotulados*, e escrevemos $(\partial\Phi, \Phi) : \mathcal{G} \cong \mathcal{H}$.

Dois grafos rotulados são isomórficos quando existe um isomorfismo entre eles, e nesse caso podemos pensar que eles são essencialmente o mesmo grafo.

Dado um grafo rotulado $\mathcal{G} = (G, \mathcal{L})$, podemos usar $\mathcal{L}$ para rotular caminhos em G ou até mesmo pontos em X_G. Se $\pi = e_1 e_2 \dots e_n$ é um caminho em G, então definimos

$$\mathcal{L}(\pi) = \mathcal{L}(e_1)\mathcal{L}(e_2)\dots\mathcal{L}(e_n), \tag{3.3}$$

que é uma palavra sobre o alfabeto $\mathcal{A}$, também chamada de *bloco rotulado*. Para cada caminho vazio ε_I de G, definimos $\mathcal{L}(\varepsilon_I) = \varepsilon$. Se $\xi = (e_n)_{n\in\mathbb{N}}$ é um caminho bi-infinito em G, ou equivalentemente $\xi \in \mathsf{X}_G$, definimos

$$\mathcal{L}_\infty(\xi) := \dots\mathcal{L}(e_{-2})\mathcal{L}(e_{-1})\mathcal{L}(e_0)\mathcal{L}(e_1)\dots, \tag{3.4}$$

que é um ponto de $\mathcal{A}^{\mathbb{Z}}$. Vamos denotar o conjunto de todos os pontos de $\mathcal{A}^{\mathbb{Z}}$ obtidos dessa forma por

$$\mathsf{X}_{\mathcal{G}} := \mathcal{L}_\infty(\mathsf{X}_G) = \{\mathcal{L}_\infty(\xi) : \xi \in \mathsf{X}_G\} \tag{3.5}$$

Definição 3.6. Um subconjunto $X \subseteq \mathcal{A}^{\mathbb{Z}}$ é dito um *subshift sófico* se $X = \mathsf{X}_{\mathcal{G}}$ para algum grafo rotulado $\mathcal{G}$. Nesse caso, dizemos que $\mathcal{G}$ é uma *apresentação* do subshift sófico X.

Assim como no Capítulo 2, consideramos apenas grafos rotulados cujos grafos adjacentes são essenciais. Note que a definição ainda não garante que um subshift sófico seja, de fato, um subshift. Esse é o nosso próximo resultado.

Teorema 3.7. *Subshifts sóficos são subshifts.*

Demonstração. Seja $\mathsf{X}_\mathcal{G}$ um subshift sófico, com $\mathcal{G} = (G, \mathcal{L})$. Note que $\mathcal{L}_\infty$ é um código de blocos com memória 0 e antecipação 0 induzido pelo mapa de blocos $\mathcal{L} : E(G) \to \mathcal{A}$. Pela Proposição 2.21, X_G é um subshift, e pelo Teorema 1.60, sua imagem $\mathsf{X}_\mathcal{G} = \mathcal{L}_\infty(\mathsf{X}_G)$ por um código de blocos também é um subshift. □

Note que qualquer grafo G pode ser transformado em um grafo rotulado de maneira trivial como $\mathcal{G} = (G, \mathrm{id}_{E(G)})$. Assim, todo shift-aresta é um subshift sófico. Veremos que o mesmo vale para subshifts do tipo finito em geral.

3.2 Caracterização de Subshifts Sóficos

Relembrando que dizemos que um subshift Y é fator de X se existe um código de blocos $\phi : X \to Y$ sobrejetor, apresentamos o seguinte resultado fundamental.

Teorema 3.8. *Um subshift é sófico se e somente se é um fator de um STF.*

Demonstração. ($\Rightarrow$) Seja $\mathsf{X}_\mathcal{G}$ um subshift sófico, com apresentação $\mathcal{G} = (G, \mathcal{L})$. Por definição, sabemos que $\mathsf{X}_\mathcal{G} = \mathcal{L}_\infty(\mathsf{X}_G)$. Logo, $\mathsf{X}_\mathcal{G}$ é fator de X_G, que é um STF pela Proposição 2.21.

($\Leftarrow$) Suponha agora que Y é um subshift tal que $\phi : X \to Y$ é um código fator, com X um STF. Suponha que ϕ tem memória m e antecipação n, de modo que seja induzido pelo mapa de blocos Φ, e podemos escrever $\phi = \Phi_\infty^{[-m,n]}$. Aumentando m se necessário, podemos supor que X tem memória $m + n$.

Pelo Teorema 2.30, sabemos que existe um grafo G tal que $\mathsf{X}_G = X^{[m+n+1]}$. Tal qual a Proposição 2.55, defina $\psi : X \to X^{[m+n+1]}$ por $\psi = \sigma^{-m} \circ \beta_{m+n+1}$, e com isso obtemos que $\tilde{\phi} = \phi \circ \psi^{-1} : X^{[m+n+1]} \to Y$ é um 1-código de blocos. Como ϕ é código fator e ψ^{-1} é conjugação, então $\tilde{\phi}$ é código fator.

Lembrando que um 1-código é um código de blocos com memória e antecipação 0, sabemos que existe um mapa de blocos $\tilde{\Phi} : \mathcal{B}_1(X^{[m+n+1]}) \to \mathcal{B}_1(Y)$ tal que $\tilde{\phi} = \tilde{\Phi}_\infty$. Agora, note que

$$\mathcal{B}_1(X^{[m+n+1]}) = \mathcal{B}_1(\mathsf{X}_G) = E(G), \tag{3.6}$$

de maneira que podemos usar $\tilde{\Phi}$ como função de rotulação. Assim, $\mathcal{G} = (G, \tilde{\Phi})$ é uma apresentação para Y, uma vez que

$$\mathsf{X}_\mathcal{G} = \tilde{\Phi}_\infty(\mathsf{X}_G) = \tilde{\phi}(X^{[m+n+1]}) = Y, \tag{3.7}$$

provando o resultado desejado. □

Corolário 3.9. *Todo subshift do tipo finito é um subshift sófico.*

Demonstração. Para um STF X, basta tomar o código identidade $\mathrm{id}_X : X \to X$. □

Corolário 3.10. *Um fator de um subshift sófico é um subshift sófico.*

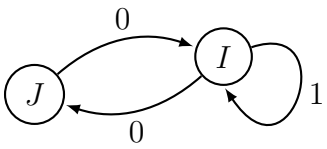
Demonstração. Seja $\phi : Y \to Z$ um código fator, com Y sófico. Pelo Teorema, existe um código fator $\psi : X \to Y$ com X um STF. Basta notar que $\phi \circ \psi : X \to Z$ é um código fator, o que torna Z sófico. □

Corolário 3.11. *Um subshift que é conjugado de um subshift sófico também é sófico.*

Demonstração. Toda conjugação é, em particular, um código fator, e o resultado segue do Corolário anterior. □

Exemplo 3.12. Considere X o shift da razão áurea e Y o shift par. Relembre que no Exemplo 2.9, construímos um código fator $\phi : X \to Y$, induzido por um mapa de blocos Φ. Usando o Teorema 2.30, construímos um grafo G tal que X_G é conjugado a X. Agora, podemos usar Φ como função rotulação e criar a apresentação $\mathsf{G} = (G, \Phi)$ para o shift par. O grafo rotulado pode ser conferido na Figura 12. Note que o shift par é um subshift que é sófico mas não é do tipo finito. ◁

Figura 12 – Apresentação do shift par como grafo rotulado.



No exemplo a seguir, construímos um subshift que não é sófico.

Exemplo 3.13. Considere $\mathcal{A} = \{a, b, c\}$, e o conjunto de blocos proibidos

$$\mathcal{F} = \{ab^n c^k a : n, k \in \mathbb{N}, n \neq k\}. \tag{3.8}$$

O subshift $\mathsf{X}_\mathcal{F}$ é chamado de *subshift livre de contexto*. Vamos mostrar que $\mathsf{X}_\mathcal{F}$ não é um subshift sófico.

Suponha, por absurdo, que exista uma apresentação $(G, \mathcal{L})$ para $\mathsf{X}_\mathcal{F}$. Seja r o número de vértices de G. Como $w = ab^{r+1}c^{r+1}a$ é um bloco permitido na linguagem de $\mathsf{X}_\mathcal{F}$, existe um caminho π tal que $\mathcal{L}(\pi) = w$. Seja τ o subcaminho de π que apresente b^{r+1}. Como G só tem r vértices, então pelo menos dois vértices de τ devem ser iguais, o que garante que τ contém um ciclo. Escreva $\tau = \tau_1\tau_2\tau_3$ com τ_2 sendo o ciclo, e τ_1, τ_3 podem ser possivelmente caminhos vazios. Note que $\tau' = \tau_1\tau_2\tau_2\tau_3$ é um caminho em G, e se trocarmos τ por τ' em π, obteremos um caminho π' tal que $\mathcal{L}(\pi') = ab^{r+s+1}c^{r+1}a$, com $s = |\tau_2|$. Mas esse bloco pertence a $\mathcal{F}$, o que é uma contradição. ◁

3.3 Autômatos

Nessa seção, introduzimos os conceitos vindos da teoria da computação: autômatos finitos e linguagens regulares. Enunciaremos um resultado que garante a equivalência entre autômatos finitos determinísticos e não-determinísticos, bem como mostraremos que um subshift é sófico se e somente se sua linguagem é regular.

Definição 3.14. Um *autômato finito determinístico* (AFD) é uma quíntupla

$$\mathscr{A} = (Q, \Sigma, \delta, q_0, F), \tag{3.9}$$

na qual

(1) Q é um conjunto finito cujos elementos são chamados de *estados*,
(2) Σ é um conjunto finito chamado de *alfabeto*,
(3) $\delta : Q \times \Sigma \to Q$ é a *função de transição*,
(4) $q_0 \in Q$ é o *estado inicial*, e
(5) $F \subseteq Q$ é o conjunto de *estados de aceitação*.

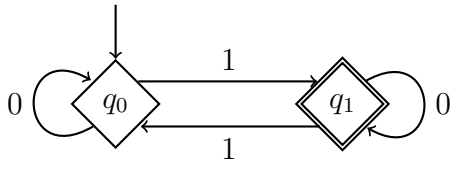
Exemplo 3.15. Podemos representar um AFD $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ por um grafo rotulado, no qual os estados constituem o conjunto dos vértices, e para cada elemento $(q, a) \in Q \times \Sigma$ existe uma aresta com rótulo a que inicia em q e termina em $\delta(q, a)$. Indicamos o estado inicial q_0 com uma aresta que inicia em nenhum vértice e termina em q_0, e representamos os estados finais com traçado duplo.

Na Figura 13 temos um exemplo de AFD com $Q = \{q_0, q_1\}$, $\Sigma = \{0, 1\}$, $F = \{q_1\}$ e δ dada por

$$\delta(q_0, 0) = q_0, \qquad \delta(q_0, 1) = q_1, \qquad \delta(q_1, 0) = q_1, \qquad \delta(q_1, 1) = q_0. \tag{3.10}$$

◁

Figura 13 – Exemplo de autômato finito.



Podemos pensar que um autômato é uma máquina que lê palavras e retorna uma resposta binária: ou ele aceita a palavra ou ele rejeita a palavra. Dada uma palavra $w = w_1 \dots w_n$ no alfabeto Σ, a computação é feita da seguinte forma: o autômato inicia no estado q_0, lê o símbolo w_1, e muda para o estado $e_1 = \delta(q_0, w_1)$. Em seguida, ele lê o símbolo w_2 e muda para o estado $e_2 = \delta(e_1, w_2)$, e segue assim até ler todos os símbolos da palavra w. Se o estado final desse procedimento for um estado de aceitação, a palavra é aceita; caso contrário, a palavra é rejeitada.

Pensando o autômato como um grafo rotulado, dada uma palavra $w = w_1 \dots w_n$, iniciamos no vértice q_0 e seguimos sequencialmente na direção das arestas $w_1, w_2, \dots, w_n$, e chegamos em um vértice r_n. O autômato só aceita a entrada se $r_n \in F$. Isso nos leva a seguinte definição.

Definição 3.16. Denotamos o conjunto de palavras (incluindo a palavra vazia) sobre um alfabeto finito Σ qualquer por Σ^*. Uma *linguagem* é um conjunto de palavras de um alfabeto.

Definição 3.17. Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um AFD e $w = w_1 \dots w_n$ uma palavra no alfabeto Σ. Dizemos que $\mathscr{A}$ *aceita* w se existe uma sequência de estados $r_0, r_1, \dots, r_n$ tal que

(1) $r_0 = q_0$,
(2) $r_i = \delta(r_{i-1}, w_i)$, para $i = 1, \dots, n$, e
(3) $r_n \in F$.

Definimos a *linguagem reconhecida* pelo autômato $\mathscr{A}$ como sendo o conjunto

$$\mathscr{L}(\mathscr{A}) = \{w \in \Sigma^* : \mathscr{A} \text{ aceita } w\}. \tag{3.11}$$

Exemplo 3.18. O autômato do exemplo 3.15 aceita as palavras 1, 10, 010, 100, 111 e rejeita as palavras 00, 11, 011, 101.

Vamos mostrar que a linguagem reconhecida pelo autômato é descrita por

$$A = \{w \in \Sigma^* : w \text{ contém uma quantidade ímpar de 1's}\}. \tag{3.12}$$

Para isso, basta ver que $\delta(q_i, 0) = q_i$, para $i = 0, 1$. Também, $\delta(q_i, 1) = q_{\overline{i+1}}$, no qual $\overline{i+1}$ é o resto da divisão de $i + 1$ por 2. Sendo k o número de símbolos 1 presentes na palavra w, note que o estado final que a computação para é $q_{\overline{k}}$. Então, a palavra é aceita pelo autômato se e somente se $q_{\overline{k}} = q_1$, o que é equivalente a dizer que k é ímpar. ◁

Uma maneira de condensar a computação é através da função de transição estendida.

Definição 3.19. Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um AFD. A *função transição estendida*, denotada por

$$\underline{\delta} : Q \times \Sigma^* \to Q \tag{3.13}$$

é definida recursivamente da seguinte maneira:

(1) $\underline{\delta}(q, \varepsilon) = q$, para qualquer estado q, com ε o bloco vazio,
(2) $\underline{\delta}(q, aw) = \underline{\delta}(\delta(q, a), w)$, para qualquer estado q, símbolo a e palavra w.

Note então que, com a definição acima, dizer que uma palavra $w = w_1 \dots w_n$ é aceita pelo autômato $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ é equivalente a dizer que $\underline{\delta}(q_o, w)$ é um estado de

aceitação. De fato, sejam $r_0, r_1, \ldots, r_n$ os estados dados pela definição de que $\mathscr{A}$ aceita w. Então

$$\begin{aligned}\underline{\delta}(q_o, w) &= \underline{\delta}(\delta(q_o, w_1), w_2 \ldots w_n) = \underline{\delta}(r_1, w_2 \ldots w_n) = \\ &= \underline{\delta}(\delta(r_1, w_2), w_3 \ldots w_n) = \underline{\delta}(r_2, w_3 \ldots w_n) = \ldots \\ &\ldots = \underline{\delta}(r_{n-1}, w_n) = \underline{\delta}(\delta(r_{n-1}, w_n), \varepsilon) = \underline{\delta}(r_n, \varepsilon) = r_n \in F\end{aligned} \tag{3.14}$$

Da mesma forma, se $\underline{\delta}(q_0, w) \in F$, definimos $r_0 = q_0$ e $r_i = \delta(r_{i-1}, w_i)$ para $i = 1, \ldots, n$. Note que $r_n = \underline{\delta}(q_0, w) \in F$, e de fato, essa sequência $r_0, \ldots, r_n$ satisfaz a definição de que $\mathscr{A}$ aceita w.

O adjetivo determinístico dado aos autômatos acima definidos deve-se ao fato de que, em qualquer momento durante a computação, o estado seguinte está completamente determinado pelo estado atual e pelo respectivo símbolo da entrada. Em termos de grafos rotulados, para cada vértice existe exatamente uma aresta saindo desse vértice com rótulo a, para cada símbolo a do alfabeto.

Um autômato finito *não-determinístico* seria um autômato no qual essa propriedade não vale. Para um vértice qualquer do grafo rotulado e um símbolo a do alfabeto, podem existir várias arestas saindo desse vértice com rótulo a – e até mesmo nenhuma.

Como a computação é feita nesse caso? Ao ler um símbolo que possui várias arestas associadas, o autômato cria cópias de si, de maneira que cada cópia segue cada aresta que possua aquele símbolo como rótulo. E cada cópia do autômato segue essa mesma regra, de maneira que podemos imaginar vários ramos de computação. Se em algum momento não existir aresta com o símbolo lido, a cópia do autômato deixa de existir. Finalmente, se alguma das cópias do autômato termina a computação em um estado de aceitação, dizemos que o autômato aceita a entrada.

Da mesma forma que a função de transição estendida pode ser aplicada ao bloco vazio ε, permitimos ε como rótulo de arestas no caso não-determinístico, e denotamos por $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$. Se o autômato estiver em um vértice que contém alguma aresta de saída com rótulo ε, ele cria cópias de si mesmo sem ler nenhuma entrada, e cada cópia segue na direção das arestas que contém o rótulo ε.

Com essa digressão, estamos prontos para dar a definição de autômato finito não-determinístico e a maneira como ele computa. A seguir, $\mathcal{P}(Q)$ representa o conjunto das partes de Q.

Definição 3.20. Um *autômato finito não-determinístico* (AFN) é uma quíntupla

$$\mathscr{A} = (Q, \Sigma, \delta, q_0, F), \tag{3.15}$$

na qual

(1) Q é um conjunto finito cujos elementos são chamados de *estados*,

(2) Σ é um conjunto finito chamado de *alfabeto*,

(3) $\delta : Q \times \Sigma_\varepsilon \to \mathcal{P}(Q)$ é a *função de transição*,

(4) $q_0 \in Q$ é o *estado inicial*, e
(5) $F \subseteq Q$ é o conjunto de *estados de aceitação*.

Note que no caso não-determinístico, para cada estado e cada símbolo, a função de transição nos dá um conjunto de possíveis estados futuros.

Ao representar um AFN como um grafo rotulado, para cada estado q, cada símbolo $a \in \Sigma_\varepsilon$, e cada estado $p \in \delta(q, a)$, temos uma aresta iniciando em q e terminando em p com rótulo a. Um aresta com rótulo ε é chamada de *aresta vazia*.

Definição 3.21. Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um AFN e $w = w_1 \dots w_n$ uma palavra no alfabeto Σ_ε. Dizemos que $\mathscr{A}$ *aceita* w se existe uma sequência de estados $r_0, r_1, \dots, r_n$ tal que

(1) $r_0 = q_0$,
(2) $r_i \in \delta(r_{i-1}, w_i)$, para $i = 1, \dots, n$, e
(3) $r_n \in F$.

De modo geral, estamos focando nossa atenção apenas em palavras sem o símbolo vazio. Então, podemos dizer que $w \in \Sigma^*$ é *aceita* por $\mathscr{A}$ se existem símbolos $w_1, \dots, w_n \in \Sigma_\varepsilon$ não todos vazios tais que w é igual a concatenação $w_1 \dots w_n$, e $\mathscr{A}$ aceita a palavra $w_1 \dots w_n$. A *linguagem reconhecida* por $\mathscr{A}$ é definida como

$$\mathscr{L}(\mathscr{A}) = \{w \in \Sigma^* : \mathscr{A} \text{ aceita } w\}. \tag{3.16}$$

Exemplo 3.22. Seja $\Sigma = \{0, 1\}$, $Q = \{q_0, q_1, q_2\}$, $F = \{q_2\}$ e $\delta : Q \times \Sigma_\varepsilon \to \mathcal{P}(Q)$ dada pela seguinte tabela

δ	ε	0	1
q_0	$\emptyset$	$\{q_0, q_1\}$	$\{q_2\}$
q_1	$\emptyset$	$\emptyset$	$\{q_0\}$
q_2	$\{q_1\}$	$\emptyset$	$\emptyset$

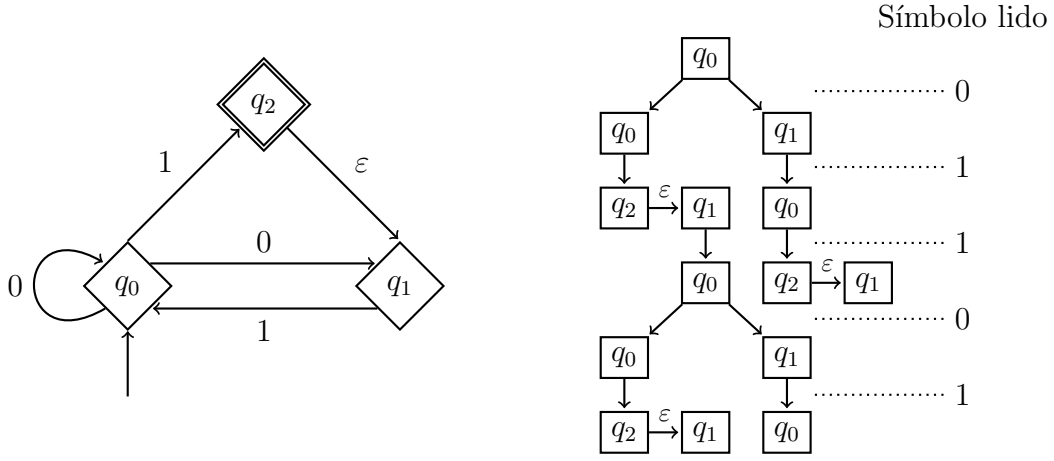
O grafo rotulado que representa o AFN $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ é mostrado na Figura 14a. Na Figura 14b, temos uma representação de como seria o processo de computar a entrada 01101. Essa palavra é aceita, uma vez que ela é a concatenação $01\varepsilon101$, que é uma palavra aceita pelo autômato, pois a sequência $q_0, q_0, q_2, q_1, q_0, q_0, q_2$ satisfaz as condições da Definição 3.21.

Note que em cada passo da computação, o autômato se encontra em vários estados ao mesmo tempo. ◁

Definição 3.23. Dizemos que dois autômatos (determinísticos ou não-determinísticos) $\mathscr{A}$ e $\mathscr{B}$ são *equivalentes* se eles reconhecem a mesma linguagem, i.e.,

$$\mathscr{L}(\mathscr{A}) = \mathscr{L}(\mathscr{B}). \tag{3.17}$$

Figura 14 – Exemplo de computação num AFN.



(a) Autômato finito não-determinístico.

(b) Ramos de computação.

Exemplo 3.24. *(Todo AFD é equivalente a um AFN).* Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um AFD. Vamos construir um AFN que seja equivalente a $\mathscr{A}$. Basta definirmos a função de transição $\delta' : Q \times \Sigma_\varepsilon \to Q$ da seguinte maneira: $\delta'(q, \varepsilon) = \emptyset$, e $\delta'(q, a) = \{\delta(q, a)\}$ para $a \in \Sigma$. Assim, o autômato $\mathscr{A}' = (Q, \Sigma, \delta', q_0, F)$ é não-determinístico. Note que não existe nenhuma aresta vazia em $\mathscr{A}'$. Vamos então provar que $\mathscr{L}(\mathscr{A}) = \mathscr{L}(\mathscr{A}')$.

($\subseteq$) Seja $w = w_1 \dots w_n \in \mathscr{L}(\mathscr{A})$. Então, existem $r_0, \dots, r_n \in Q$ tais que $r_0 = q_0, r_i = \delta(r_{i-1}, w_i), i = 1, \dots, n$ e $r_n \in F$. Esses mesmos r_i satisfazem a Definição 3.21 para $\mathscr{A}'$, uma vez que a condição (2) se traduz como $r_i \in \delta'(r_{i-1}, w_i) \Leftrightarrow r_i \in \{\delta(r_{i-1}, w_i)\}$ para $i = 1, \dots, n$. Logo, $w \in \mathscr{L}(\mathscr{A}')$.

($\supseteq$) Se $w = w_1 \dots w_n \in \mathscr{L}(\mathscr{A}')$ então existem $r_0, \dots, r_n$ que satisfazem a Definição 3.21. Note que a condição (2) só é satisfeita, se $r_i = \delta(r_{i-1}, w_i)$, $i = 1, \dots, n$. Assim, esses mesmos r_i satisfazem a Definição 3.17, de modo que $w \in \mathscr{L}(\mathscr{A})$. ◁

Embora os AFN sejam uma generalização dos AFD, existe um resultado que garante que dado qualquer autômato não-determinístico, existe um autômato determinístico equivalente a ele.

Para apresentarmos a ideia da demonstração desse resultado, iremos construir a função de transição estendida de um autômato não-determinístico. Para isso, precisamos definir a função *fecho vazio*.

Definição 3.25. Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um AFN. O *fecho vazio* de um estado, denotado por $\mathscr{E}(q)$, é o conjunto de todos os estados que podem ser alcançados seguindo arestas com o símbolo vazio, i.e., $p \in \mathscr{E}(q)$ se existe uma sequência de estados $r_0, \dots, r_n$ tal que $r_0 = q$, $r_i \in \delta(r_{i-1}, \varepsilon)$, para $i = 1, \dots, n$ e $r_n = p$. Em particular, para $n = 0$, temos que $q \in \mathscr{E}(q)$ para todo $q \in Q$. Definimos a *função fecho vazio* $\mathscr{E} : Q \to \mathcal{P}(Q)$ que associa a cada $q \in Q$ seu fecho vazio $\mathscr{E}(q) \in \mathcal{P}(Q)$.

A função *fecho vazio estendido* $\underline{\mathscr{E}} : \mathcal{P}(Q) \to \mathcal{P}(Q)$ é definida como

$$\underline{\mathscr{E}}(P) = \bigcup_{q \in P} \mathscr{E}(q). \tag{3.18}$$

Exemplo 3.26. Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um AFN. Note que, ao iniciar uma computação, sem precisar ler nenhuma entrada, podemos nos mover para qualquer estado do fecho vazio do estado inicial q_0. Efetivamente, a computação inicia em todos os estados $\mathscr{E}(q_0)$. Chamaremos os estados em que o autômato e suas cópias se encontram em cada passo da computação de *estados ativos*. Então, suponha que é lido o símbolo w_1 da entrada. Cada cópia do autômato se move seguindo a função δ. Se o conjunto dos estados ativos era $R \subseteq Q$, então ao ler o símbolo w_1, qual será o novo conjunto dos estados ativos?

Se uma cópia do autômato estava em um estado $q \in R$, então o novo conjunto de estados ativos para essa cópia será $\delta(q, w_1)$. Mais que isso, como cada estado pode conter arestas vazias, temos que considerar o fecho vazio de $\delta(q, w_1)$. Assim, o conjunto de novos estados ativos será

$$\bigcup_{q \in R} \underline{\mathscr{E}}(\delta(q, w_1)). \tag{3.19}$$

Podemos definir a função de transição estendida como $\underline{\delta} : \mathcal{P}(Q) \times \Sigma_\varepsilon \to \mathcal{P}(Q)$ por

(1) $\underline{\delta}(R, \varepsilon) = \underline{\mathscr{E}}(R)$,

(2) $\underline{\delta}(R, a) = \bigcup_{q \in R} \underline{\mathscr{E}}(\delta(q, a))$, para $a \in \Sigma$.

◁

Teorema 3.27. *Todo autômato finito não-determinístico é equivalente a um autômato finito determinístico.*

Ideia da demonstração. Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um AFN. Vamos construir um AFD $\mathscr{A}' = (Q', \Sigma, \delta', q_0', F')$ tal que $\mathscr{L}(\mathscr{A}) = \mathscr{L}(\mathscr{A}')$. Definimos os componentes de $\mathscr{A}'$ como:

(1) $Q' = \mathcal{P}(Q)$, pois a cada passo da computação, o autômato $\mathscr{A}$ pode estar em vários estados simultaneamente.

(2) $q_0' = \mathscr{E}(q_0)$, pois ao iniciar a computação, $\mathscr{A}$ pode estar em qualquer estado no fecho-vazio de q_0.

(3) $F' = \{R \in Q' : R \cap F \neq \emptyset\}$, isto é, F' é o conjunto de todos os subconjunto de Q que contém algum estado de aceitação de $\mathscr{A}$.

(4) $\delta' : Q' \times \Sigma \to Q'$, definiremos a função de transição de maneira análoga ao Exemplo 3.26.

$$\delta'(R, a) = \bigcup_{q \in R} \underline{\mathscr{E}}(\delta(q, a)). \tag{3.20}$$

A prova de que $\mathscr{A}$ e $\mathscr{A}'$ reconhecem a mesma linguagem é feita por indução no tamanho da entrada, e pode ser encontrada em [3]. □

Agora, por autômato, entenda-se um autômato finito determinístico.

Definição 3.28. Seja $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$ um autômato. Dizemos que um estado q é *acessível* se existe uma palavra w tal que $\underline{\delta}(w, q_0) = q$. Dizemos que q é *coacessível* se existe uma palavra w tal que $\underline{\delta}(w, q) \in F$. Dizemos que o autômato é *acessível* (respectivamente *coacessível*) se todo estado $q \in Q$ é acessível (respectivamente coacessível). Dizemos que o autômato é *aparado* se ele for acessível e coacessível.

Dizer que um estado é acessível equivale a dizer que existe um caminho no grafo rotulado que começa em q_0 e termina em q, e dizer que o mesmo é coacessível significa dizer que existe um caminho começando em q e terminando em um estado de aceitação. Note que um estado que não é acessível nem coacessível não pode aparecer na computação de uma palavra aceita na linguagem. Por isso, só consideraremos autômatos aparados.

Definição 3.29. Uma linguagem é dita *regular* se é a linguagem reconhecida por algum autômato finito.

Proposição 3.30. *A linguagem de um subshift sófico é uma linguagem regular.*

Demonstração. Seja X um subshift sófico, com apresentação $\mathcal{G} = (G, \mathcal{L})$, em que $G = (V, E)$ é um grafo essencial e $\mathcal{L} : E \to \Sigma$ a rotulação. Vamos construir um AFN que aceite qualquer caminho finito no grafo $\mathcal{G}$.

Seja q_0 um elemento que não está no conjunto dos vértices V. A ideia essencial é adicionar esse vértice como sendo o vértice inicial do autômato, ligá-lo a todos os outros vértices usando arestas vazias, e definir todos os outros como vértices finais.

Seja então $\mathscr{A} = (V \cup \{q_0\}, \Sigma, \delta, q_0, V)$. Para definirmos a função de transição, denote por $V(q, a)$ o conjunto de vértices que podem ser alcançados a partir de q lendo a entrada a, i. e., $V(q, a) = \{t(e) : e \in \mathbf{i}(q), \mathcal{L}(e) = a\}$. Assim, a função de transição pode ser descrita como

δ	ε	$a \in \Sigma$
q_0	V	$\emptyset$
$q \in V$	$\emptyset$	$V(q, a)$

Como G é essencial, qualquer palavra em $w \in \mathcal{B}(X)$ é um caminho finito em $\mathcal{G}$. Note que w é aceita por $\mathscr{A}$ pois εw é aceita. Da mesma forma, qualquer palavra aceita por $\mathscr{A}$ é da forma εw, em que w não contém nenhum bloco vazio. Assim, w é um caminho no grafo $\mathcal{G}$, logo $w \in \mathcal{B}(\mathsf{X}_{\mathcal{G}})$. □

Exemplo 3.31. Considere X o shift completo sobre Σ. Note que X é um STF, e logo, é um subshift sófico. Pela Proposição que acabamos de provar, $\mathcal{B}(X) = \Sigma^*$ é uma linguagem regular. ◁

Definição 3.32. Sejam A e B duas linguagens sobre o mesmo alfabeto Σ. Definimos a *concatenação* de A e B como sendo o conjunto de todas as palavras obtidas pela concatenação de uma palavra em A com uma palavra em B, denotado por

$$AB = \{ab \in \Sigma^* : a \in A, b \in B\}. \tag{3.21}$$

Lema 3.33. *O conjunto das linguagens regulares é fechado por concatenação e complementação.*

Demonstração. Sejam A e B linguagens regulares reconhecidas pelos autômatos finitos determinísticos $\mathscr{A} = (Q_A, \Sigma, \delta_A, q_A, F_A)$ e $\mathscr{B} = (Q_B, \Sigma, \delta_B, q_B, F_B)$, respectivamente.

Primeiramente, vamos construir um autômato que reconheça a linguagem AB. Seja $\mathscr{N} = (Q_A \cup Q_B, \Sigma, \delta_1, q_A, F_B)$, com δ_1 descrita por

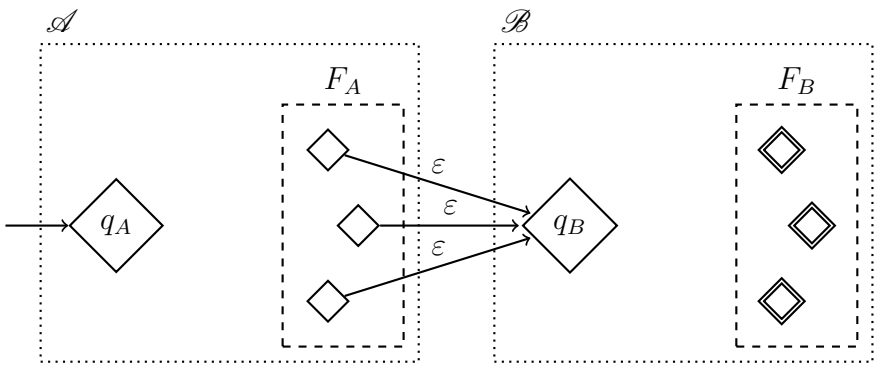
δ_1	ε	$a \in \Sigma$
$q \in Q_A \backslash F_A$	$\emptyset$	$\{\delta_1(q,a)\}$
$q \in F_A$	$\{q_B\}$	$\{\delta_1(q,a)\}$
$q \in Q_B$	$\emptyset$	$\{\delta_2(q,a)\}$

A Figura 15 ilustra essa construção. O que fizemos foi ligar os estados finais de $\mathscr{A}$ com o estado inicial q_B de $\mathscr{B}$ usando arestas vazias.

Agora, mostraremos que $\mathscr{L}(\mathscr{N}) = AB$.

($\subseteq$) Se uma palavra $w \in \Sigma_\varepsilon^*$ é aceita por $\mathscr{N}$, então ela representa um caminho começando em q_A e terminando em algum estado de F_B. Pela nossa construção, um caminho que comece em q_A e chegue em algum estado de F_B deve necessariamente passar por algum estado de F_A, seguir por uma aresta vazia e chegar em q_B. Dessa maneira, podemos escrever $w = a\varepsilon b$, em que a é o rótulo de algum caminho em $\mathscr{A}$ que inicia em q_A e termina em algum estado de F_A, e b é rótulo de um caminho começando em q_B e

Figura 15 – Autômato finito reconhecendo a linguagem AB.



Na figura, ocultamos os vértices de ambos os autômatos que não sofreram alteração com a construção. Note que qualquer caminho que saia de um estado de $\mathscr{A}$ e chega em um estado de $\mathscr{B}$ deve necessariamente passar por uma aresta vazia.

terminando em um estado de F_B. Assim, a é aceito por $\mathscr{A}$ e b é aceito por $\mathscr{B}$, de modo que $ab \in AB$, como desejado.

($\supseteq$) Veja que qualquer palavra $ab \in AB$ é aceita por $\mathscr{N}$. Isso pois, se $a \in A$, então a é o rótulo de um caminho em $\mathscr{A}$ que inicia em q_A e termina em algum estado de F_A. Assim, seguimos por uma aresta vazia até q_B, e como $b \in B$, b é o rótulo de um caminho começando em q_B e terminando em algum estado de F_B. Logo, $a\varepsilon b$ é uma palavra aceita por $\mathscr{N}$, como desejado.

Agora, vamos construir um autômato finito que reconheça a linguagem $\Sigma^*\backslash A$. Note que esse é o conjunto de palavras que são rejeitadas por $\mathscr{A}$. Como $\mathscr{A}$ é determinístico, uma palavra w é rejeitada se e somente se é o rótulo de um caminho começando em q_A e terminando em um estado fora de F_A. Assim, podemos definir $\mathscr{M} = (Q_A, \Sigma, \delta_A, q_A, Q_A\backslash F_A)$, de modo que uma palavra é aceita por $\mathscr{M}$ se e somente se é rejeitada por $\mathscr{A}$, como desejado. □

Proposição 3.34. *Seja $W \subseteq \Sigma^*$ uma linguagem regular e seja $\mathscr{A}$ um AFD aparado reconhecendo a linguagem $\Sigma^*\backslash(\Sigma^* W \Sigma^*)$. Então $\mathscr{L}(\mathscr{A}) = \mathcal{B}(\mathsf{X}_W)$.*

Demonstração. Note que, de fato, $\Sigma^*\backslash(\Sigma^* W \Sigma^*)$ é regular, pois é o complemento da concatenação de linguagens regulares. Denote $\mathscr{A} = (Q, \Sigma, \delta, q_0, F)$. Vamos mostrar que $\mathscr{L}(\mathscr{A}) = \mathcal{B}(\mathsf{X}_W)$.

($\subseteq$) Seja w uma palavra aceita pelo autômato $\mathscr{A}$. Suponha que w contenha algum sub-bloco u que pertence a W. Então existe um caminho em $\mathscr{A}$ com rótulo u. Como $\mathscr{A}$ é aparado, existe um caminho com rótulo a que inicia em q_0 e termina no primeiro estado de u. Da mesma forma, existe um caminho com rótulo b que inicia no último estado de u e termina em um estado final. Assim, a palavra aub seria aceita por $\mathscr{A}$, o que é uma contradição, pois $aub \in \Sigma^* W \Sigma^*$.

($\supseteq$) Se $w \in \mathcal{B}(\mathsf{X}_W)$, então w não contém nenhum sub-bloco que pertença a W. Assim, $w \notin \Sigma^* W \Sigma^*$, ou seja, $w \in \Sigma^*\backslash(\Sigma^* W \Sigma^*)$, de modo que w é aceita por $\mathscr{A}$. □

Corolário 3.35. *Se A é uma linguagem regular então X_A é um subshift sófico.*

Demonstração. Se A é uma linguagem regular, pela Proposição 3.34, existe um autômato finito aparado $\mathscr{A}$ tal que $\mathscr{L}(\mathscr{A}) = \mathcal{B}(\mathsf{X}_A)$. Como $\mathscr{A}$ é aparado, cada palavra em $\mathcal{B}(\mathsf{X}_A)$ é o rótulo de um caminho num grafo rotulado determinado por $\mathscr{A}$. Tal grafo é uma apresentação para o subshift X_A, o que mostra que esse ultimo é um subshift sófico, como desejado. □

Teorema 3.36. *Um subshift é sófico se e somente se sua linguagem é regular.*

Demonstração. ($\Rightarrow$) Segue da Proposição 3.30.

($\Leftarrow$) Seja X um subshift tal que $\mathcal{B}(X)$ é uma linguagem regular. Pelo Lema 3.33, $\mathcal{B}(X)^c$ é regular.

Relembrando o Teorema 1.39, que caracteriza a linguagem de um subshift, temos que para qualquer subshift X vale a igualdade $X = \mathsf{X}_{\mathcal{B}(X)^c}$. Agora, usamos o Corolário 3.35, temos que $\mathsf{X}_{\mathcal{B}(X)^c}$ é sófico, como desejado. □

Considerações Finais

Grande parte dos resultados desse texto se baseiam no fato de que nosso alfabeto é finito, entre eles a compacidade do shift completo e a compacidade dos subshifts. Pesquisas muito recentes nessa área generalizam os resultados desse texto para o caso em que o alfabeto é infinito.

A codificação de subshifts do tipo finito em shifts-aresta estabelece uma relação entre a dinâmica simbólica e a teoria de grafos. Além disso, o Teorema da Decomposição da Seção 2.5 permitiu entender uma conjugação entre dois STF como uma sequência de transformações na estrutura dos grafos que os representam.

Ao permitirmos grafos rotulados, expandimos a classe de subshifts que poderíamos representar, os chamados subshifts sóficos. Em paralelo, mostramos que autômatos finitos também podem ser representados como grafos rotulados, o que nos permitiu estabelecer uma conexão entre a dinâmica simbólica e a teoria da computação.

Referências

[1] Marie-Pierre Béal, Jean Berstel, Soren Eilers, and Dominique Perrin. Symbolic dynamics. 2010. to appear in Handbook of Automata.

[2] Jacques Hadamard. Les surfaces à courbures opposées et leurs lignes géodésique. *J. Math. Pures Appl.*, 4:27–73, 1898.

[3] John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages and Computation (Addison-Wesley series in computer science)*. Addison-Wesley Publishing Company, 1979.

[4] Elon Lages Lima. *Espaços métricos*. Instituto de Matemática Pura e Aplicada, CNPq, Rio de Janeiro, 4.ed., 2009.

[5] Douglas Lind and Brian Marcus. *An introduction to symbolic dynamics and coding*. Cambridge University Press, Cambridge New York, 1995.

[6] Marston Morse and Gustav A. Hedlund. Symbolic dynamics. *American Journal of Mathematics*, 60(4):815, oct 1938.

[7] Michael Sipser. *Introduction to the Theory of Computation*. Course Technology, 2006.