



VICTOR MENDONÇA ORTIZ SIQUEIRA

CURVAS DE PREENCHIMENTO DE ESPAÇO:
ESTUDO E APLICAÇÃO COMPUTACIONAL

Santo André, 2020



UNIVERSIDADE FEDERAL DO ABC

CENTRO DE MATEMÁTICA, COMPUTAÇÃO E COGNIÇÃO

VICTOR MENDONÇA ORTIZ SIQUEIRA

CURVAS DE PREENCHIMENTO DE ESPAÇO:

ESTUDO E APLICAÇÃO COMPUTACIONAL

Orientador: Prof. Dr. Rodrigo Roque Dias

Dissertação de mestrado apresentada ao Centro de
Matemática, Computação e Cognição para
obtenção do título de Mestre Profissional

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO
DEFENDIDA PELO ALUNO VICTOR MENDONÇA ORTIZ SIQUEIRA,
E ORIENTADA PELO PROF. DR. RODRIGO ROQUE DIAS.

SANTO ANDRÉ, 2020

Sistema de Bibliotecas da Universidade Federal do ABC
Elaborada pelo Sistema de Geração de Ficha Catalográfica da UFABC
com os dados fornecidos pelo(a) autor(a).

MENDONÇA ORTIZ SIQUEIRA, VICTOR

Curvas de preenchimento de espaço : estudo e aplicação
computacional / VICTOR MENDONÇA ORTIZ SIQUEIRA. — 2020.

73 fls. : il.

Orientador: RODRIGO ROQUE DIAS

Dissertação (Mestrado) — Universidade Federal do ABC, Mestrado
Profissional em Matemática em Rede Nacional - PROFMAT,
Santo André, 2020.

1. curvas de preenchimento de espaço. I. ROQUE DIAS,
RODRIGO. II. Mestrado Profissional em Matemática em Rede
Nacional - PROFMAT, 2020. III. Título.

Este exemplar foi revisado e alterado em relação à versão original, de acordo com as observações levantadas pela banca no dia da defesa, sob responsabilidade única do(a) autor(a) e com a anuência do(a) orientador(a).

Santo André/ SP

17 de janeiro de 2020

Assinatura do(a) autor(a): Victor Mendonça Ortiz Siqueira

Assinatura do(a) orientador(a):





MINISTÉRIO DA EDUCAÇÃO

Fundação Universidade Federal do ABC

Avenida dos Estados, 5001 – Bairro Santa Terezinha – Santo André – SP
CEP 09210-580 · Fone: (11) 4996-0017

FOLHA DE ASSINATURAS

Assinaturas dos membros da Banca Examinadora que avaliou e aprovou a Defesa de Dissertação de Mestrado do candidato, VICTOR MENDONÇA ORTIZ SIQUEIRA realizada em 16 de Dezembro de 2019:

Prof.(a) LUCIA RENATO JUNQUEIRA
UNIVERSIDADE DE SÃO PAULO

Prof.(a) SINUE DAYAN BARBERO LODOVICI
UNIVERSIDADE FEDERAL DO ABC

Prof.(a) DANIEL MIRANDA MACHADO
UNIVERSIDADE FEDERAL DO ABC

Prof.(a) RENATO MENDES COUTINHO
UNIVERSIDADE FEDERAL DO ABC

Prof.(a) RODRIGO ROQUE DIAS
UNIVERSIDADE FEDERAL DO ABC - Presidente

* Por ausência do membro titular, foi substituído pelo membro suplente descrito acima: nome completo, instituição e assinatura

O presente trabalho foi realizado com apoio da
Coordenação de Aperfeiçoamento de Pessoal de
Nível Superior - Brasil (CAPES) - Código de Fi-
nanciamento 001

“Matemática não é apenas números, e sim envolve letras e toda a capacidade que o ser humano conseguir expressar.”

(François Viète)

RESUMO

Curvas que preenchem o espaço (outrora conhecidas como Curvas de Peano ou Curvas de Hilbert) são representações de funções que estabelecem uma relação entre um espaço euclidiano unidimensional (uma reta) e um bidimensional (uma superfície). A inquietação gerada pelo fato de ser possível preencher uma através da outra, considerando a diferença de dimensões entre elas, foi motivo de muita pesquisa nas áreas de Análise Real e Teoria dos Números. Nesta obra encontra-se uma proposta de análise sob perspectiva geométrica. Um grande entrave no estudo, a visualização dos resultados obtidos pela manipulação da função envolvida, será superado pelo uso de uma ferramenta de uso intuitivo elaborada pelo autor, de modo a levantar conclusões, sugestões, e mesmo aplicabilidade em outras áreas de estudo como a do fractal.

Palavras-chave: Curva de Preenchimento de Espaço, Curva de Peano, Curva de Lebesgue

ABSTRACT

Space-filling curves (sometimes known as Peano Curves or Hilbert Curves) are representations of functions that establish a relation between an one-dimensional euclidean space (a straight line) and a two-dimensional Euclidean space (a surface). The uneasiness brought by the fact that it is possible to fill one into another, considering the difference between their dimensions, has motivated researches in areas such as Real Analysis and Number Theory. In this study, there is a suggestion to analyze it under a geometric perspective. A big obstacle, the preview of results obtained by manipulating the given function, will be surpassed by using a tool of intuitive use, elaborated by the author, in a way readers can raise conclusions, suggestions, or even applicability in other study areas like fractals.

Keywords: Space-Filling Curve, Peano Curve, Lebesgue Curve

CONTEÚDO

INTRODUÇÃO	1
1 TERMINOLOGIA	5
1.1 Notações	5
1.2 Terminologia Básica	6
2 CURVA DE PEANO	13
2.1 Motivação	13
2.2 A Função de Peano	15
2.3 A Curva Geométrica de Peano	19
3 CURVA DE HILBERT	23
3.1 Aritmetização da Curva de Hilbert	26
4 CURVA DE LEBESGUE	33
4.1 Conjunto de Cantor	33
4.2 Definição de Lebesgue para Curva de Preenchimento de Espaço	38
4.3 Aproximação Poligonal para Curva de Lebesgue	40
5 MODELAGEM COMPUTACIONAL	45
5.1 Ambiente de programação	46
5.2 Restrições adotadas no projeto	46
5.3 Implementação do programa	47
6 DEMONSTRAÇÃO DE USO	49
7 ANÁLISE DE RESULTADOS	53
7.1 Continuidade do projeto	53
7.2 Passeio do cavalo	54
7.3 Fractais	54
7.4 Estudo de Caso	55
8 CONSIDERAÇÕES FINAIS	63

A APÊNDICE A	65
B APÊNDICE B	67
Bibliografia	73

INTRODUÇÃO

Curva de preenchimento de espaço é o nome dado a uma função matemática sobrejetora cujo domínio está contido em um conjunto unidimensional e cuja imagem está contida em um conjunto de dimensão maior.

Segundo Byrne [2], Müller [7] e Sagan [10], o estudo de teoria dos conjuntos é um ramo relativamente novo da matemática, ganhando grande impulsão nos últimos séculos. O cuidado do detalhamento dos conhecidos conjuntos numéricos ganhou força com a caracterização dos números naturais através de G. Peano e seus axiomas em 1879, junto com a expansão para os inteiros, racionais e reais.

Grande parte do estudo de conjuntos está no estudo da pertinência de seus elementos e da quantidade de entes que os compõem. No estudo de conjuntos numéricos, o principal motivador dos estudos foi a caracterização da quantidade tomada por “infinito” que caracterizaria a infinidade de números que cada conjunto numérico comporta.

Era sabido que da expansão do conjunto dos números naturais para os inteiros, dos inteiros para os racionais, racionais para reais, e assim sucessivamente, mais e mais elementos iam se agregando aos conjuntos, tornando-os cada vez “maiores”; mas este termo carecia de definição, uma vez que “infinito” não é um número, e portanto não era capaz de descrever uma quantidade para tais conjuntos.

Até que em 1878, G. Cantor apresenta ao mundo o conceito de cardinalidade: era possível “contar” os elementos no conjunto dos números inteiros utilizando os números naturais; logo, havia tantos números inteiros quanto números naturais, tornando ambos os conjuntos igualmente grandes em mesma “ordem”– dizendo-se, portanto, que ambos os conjuntos possuem mesma cardinalidade.

O conceito de cardinalidade se justifica no fato de existir uma bijeção entre os conjuntos: existe como se mapear univocamente cada elemento de um conjunto sobre outro através de uma função que desempenha o papel de “contagem de elementos”. Através deste conceito, nasce também o conceito de conjuntos finitos, infinitos enumeráveis e infinitos não enumeráveis (caso em que recai nos números reais).

Neste ambiente da descoberta da possibilidade de se mapear funções de um conjunto sobre outro aparentemente maior, surge a pesquisa da possibilidade de se mapear funções em dimensões diferentes. Seria possível criar uma bijeção entre um conjunto de dimensão x em um conjunto de dimensão y , sendo $x \neq y$ (unidimensional em bidimensional, como no exemplo)? Com isto, a curva de preenchimento de espaço começa a ganhar forma.

A esta curva, deu-se o nome de “preencher espaço” por causa da primeira curva investigada ter todos os pontos da imagem (de domínio bidimensional) relacionados a pontos pertencentes a um intervalo de números reais, sendo hoje mais correto nomear esta primeira curva de “curva que preenche superfície”, limitando o termo espaço para quando se preenche três dimensões. Como há de se mostrar adiante, tal curva é necessariamente não injetora, o que conseqüentemente implica não haver bijeção entre tais conjuntos.

Nascem aqui os primeiros estudos do ramo hoje conhecido por Topologia como uma expansão da Geometria Euclidiana tradicional, investigando o comportamento dessas funções e conjuntos quanto à sua continuidade, compacidade e conexidade. Nossa curva de preenchimento de espaço, hoje, pode ser vista de maneira mais geral como a existência de uma relação, nesta obra tratada por funções, entre espaços de dimensões não necessariamente iguais.

Mas um dos grandes entraves no estudo destas curvas era a análise dos resultados obtidos. O primeiro estudo desenvolvido no tema, por Peano, era meramente algorítmico e de difícil interpretação de como a mudança nas variáveis acarretaria o preenchimento de uma região plana. D. Hilbert, em 1891, foi o primeiro a sugerir tal curva com uma visualização de como cada ponto num espaço plano era imagem de um ponto do segmento de reta através de sucessivas iterações. Finalmente, em 1904, H. Lebesgue estende os conceitos existentes até então para um modelo ainda mais geral de curva criada por interpolação e relação com conjuntos particionados, trabalho possível graças ao notável conjunto de Cantor, descrito por G. Cantor em 1883.

O propósito desta obra, além de gerar interesse pelo estudo do tema sobre curvas e dimensões espaciais, é o de também criar uma ferramenta de uso bastante básico capaz de gerar tais curvas. Com ela, será capaz de se analisar quais parâmetros que são necessários para a criação da curva; como a alteração destes parâmetros implica (ou não) a criação de uma nova curva; e como, dada uma curva, é possível intuir seus aspectos básicos e, com eles, obter os parâmetros.

Os capítulos subsequentes tratarão das definições matemáticas básicas do tema (curvas de Peano, Hilbert e Lebesgue, a última na qual se baseia esta obra); definirá como construir um software computacional capaz de visualizar sua curva, baseada nos algoritmos de construção das curvas propostos pelos autores estudados; mostrará como o software criado é capaz de auxiliar cognitivamente na construção e parametrização das curvas; e, por fim, mostrará alguns exemplos de abordagens do mundo real capazes de serem modelados pelo software (e, conseqüentemente, explorados como tais curvas).

Esta obra não tem a pretensão de servir como um curso completo sobre o tema, mas sim servir de motivação e fomento para mais pesquisas no tema, bem como instigar a criação de plataformas computacionais mais sofisticadas capazes de gerar visualizações para níveis multidimensionais (como, por exemplo, para o cubo).

TERMINOLOGIA

Começemos, então, a definir a terminologia básica que será utilizada ao longo deste trabalho. Introduziremos as notações adotadas, bem como conceitos tomados por básicos para a correta interpretação dos temas. O conteúdo das definições pode ser consultado em Kaplansky [4]; e Lima [5, 6].

Por fim, sempre que útil, adotaremos proposições e demonstrações que reforcem temas específicos, utilizando os axiomas básicos para desenvolver um raciocínio que complete asserções primitivas em mais complexas.

Note que, de forma a fomentar a construção do raciocínio que acompanha o estudo histórico do tema, alguns conceitos e definições deixarão para ser abordados no capítulo correspondente da curva estudada, para que se prove que, dadas as formas de concepção iniciais, o nível de abstração foi sendo construído com o tempo de forma a poder abranger e caracterizar curvas do tipo de preenchimento de espaço com maior precisão.

1.1 NOTAÇÕES

Começemos então introduzindo as notações utilizadas neste trabalho:

Definição 1.1. O **corpo ordenado** completo dos números reais será denotado por \mathbb{R} ; o espaço cartesiano n -dimensional por \mathbb{R}^n e o corpo dos números complexos por \mathbb{C} . Usaremos letra cursiva maiúscula para denotar **subconjuntos** de \mathbb{R}^n como:

$\mathcal{I} = [0, 1]$ para o intervalo fechado unitário;

$\mathcal{Q} = [0, 1]^2$ para o quadrado fechado unitário; e

$\mathcal{C} = [0, 1]^3$ para o cubo fechado unitário.

Definição 1.2. O **complementar** de um conjunto \mathcal{S} será denotado por \mathcal{S}^C quando for claro o contexto de qual conjunto universo o complementar será tomado. Caso contrário, escreveremos $\mathcal{U} \setminus \mathcal{S}$ para denotar o complementar de \mathcal{S} em relação ao conjunto \mathcal{U} .

Definição 1.3. A **norma euclidiana** de um vetor $x = (\xi_1, \dots, \xi_n)$ será denotada por $\|x\| = \sqrt{\sum_{k=1}^n \xi_k^2}$; e o espaço euclidiano n -dimensional (que consiste de \mathbb{R}^n com a norma euclidiana definindo a métrica) denotado por \mathbb{E}^n .

Definição 1.4. **Matrizes** serão denotadas por letras góticas maiúsculas: $\mathfrak{H}, \mathfrak{B}, \mathfrak{S}$.

Definição 1.5. Uma **injeção** de \mathcal{A} em \mathcal{B} será denotada por $f : \mathcal{A} \xrightarrow{1-1} \mathcal{B}$; uma **sobrejeção** de \mathcal{A} em \mathcal{B} será denotada por $f : \mathcal{A} \xrightarrow{em} \mathcal{B}$; e uma **bijeção** de \mathcal{A} em \mathcal{B} será denotada por $f : \mathcal{A} \leftrightarrow \mathcal{B}$.

Definição 1.6. Seja $n > 1$ um número inteiro. A **representação na base n** de um número inteiro será dada por:

$$[x_0 x_1 x_2 \dots x_j]_n = x_0 n^j + x_1 n^{j-1} + x_2 n^{j-2} + \dots + x_j, \text{ com } x_j \in \{0, 1, 2, \dots, n-1\}.$$

A representação na base n de um número em \mathcal{I} será dada por:

$$[0, x_1 x_2 x_3 \dots]_n = \frac{x_1}{n} + \frac{x_2}{n^2} + \frac{x_3}{n^3} + \dots, \text{ com } x_j \in \{0, 1, 2, \dots, n-1\}.$$

Usaremos barras superiores para indicar **períodos** como em:

$$0, a_1 a_2 a_3 \dots a_n \overline{b_1 b_2 b_3 \dots b_n} = 0, a_1 a_2 a_3 \dots a_n b_1 b_2 b_3 \dots b_n b_1 b_2 b_3 \dots b_n b_1 \dots$$

1.2 TERMINOLOGIA BÁSICA

A seguir, introduziremos alguns conceitos-chave bastante empregados na concepção histórica de curvas de preenchimento de espaço, que culminarão formalmente em sua caracterização:

Definição 1.7. Chamaremos de **vizinhança δ de $a \in \mathbb{E}^n$** o subconjunto de \mathbb{E}^n $N_\delta(a) = \{x \in \mathbb{E}^n; \|x - a\| < \delta\}$; e chamaremos de **vizinhança δ estrita de $a \in \mathbb{E}^n$** o subconjunto de \mathbb{E}^n $N'_\delta(a) = \{x \in \mathbb{E}^n; 0 < \|x - a\| < \delta\} = N_\delta(a) - \{a\}$.

Definição 1.8. Chamaremos um ponto $a \in \mathcal{A} \subseteq \mathbb{E}^n$ de **ponto interior de \mathcal{A}** se existe $\delta > 0$ tal que $N_\delta(a) \subseteq \mathcal{A}$; e denotaremos o conjunto de todos os pontos interiores de \mathcal{A} por $\text{int}(\mathcal{A})$, também chamado de **interior de \mathcal{A}** .

Definição 1.9. Diremos que o conjunto \mathcal{A} é **aberto** se $\text{int}(\mathcal{A}) = \mathcal{A}$; e diremos que o conjunto \mathcal{A} é **fechado** se \mathcal{A}^C é aberto.

Definição 1.10. Diremos que um ponto $a \in \mathbb{E}^n$ é **ponto de acumulação** de um conjunto $\mathcal{A} \subseteq \mathbb{E}^n$ se, para qualquer $\delta > 0$, tivermos $N'_\delta(a) \cap \mathcal{A} \neq \emptyset$.

Diremos que $a \in \mathbb{R}$ é **ponto de acumulação à esquerda** de \mathcal{A} se, para todo $\delta > 0$, $(a - \delta, a) \cap \mathcal{A} \neq \emptyset$; e é **ponto de acumulação à direita** de \mathcal{A} se, para todo $\delta > 0$, $(a, a + \delta) \cap \mathcal{A} \neq \emptyset$.

O **fecho** de \mathcal{A} , denotado por $\overline{\mathcal{A}}$, é definido como a união de \mathcal{A} com o conjunto de todos os seus pontos de acumulação.

Lema 1.11. Um conjunto é fechado se, e somente se, contém todos os seus pontos de acumulação; ou seja, $\overline{\mathcal{A}} = \mathcal{A}$.

Demonstração. Suponha \mathcal{A} fechado e a um ponto de acumulação com $a \notin \mathcal{A}$. Então, $a \in \mathcal{A}^C$. Como \mathcal{A}^C é aberto, então existe $\delta > 0$ tal que $N_\delta(a) \subseteq \mathcal{A}^C$. Mas então a não pode ser ponto de acumulação de \mathcal{A} pois $N'_\delta(a)$ é disjunto de \mathcal{A} .

Suponha agora que $\overline{\mathcal{A}} = \mathcal{A}$. Se $x \notin \mathcal{A}$, então $x \notin \overline{\mathcal{A}}$. Logo, x não é ponto de acumulação de \mathcal{A} e, portanto, x possui uma vizinhança que é disjunta de \mathcal{A} . Assim, x é ponto interior de \mathcal{A}^C . Isto é verdadeiro para todo $x \notin \mathcal{A}$, portanto \mathcal{A}^C é aberto; logo, \mathcal{A} é fechado. \square

Definição 1.12. Se f é uma função de um subconjunto de \mathbb{E}^m em \mathbb{E}^n , então $f[\mathcal{A}] = \{f(x); x \in \mathcal{A} \cap D(f)\}$, onde $D(f)$ denota o **domínio** de f e $\mathcal{A} \subseteq \mathbb{E}^m$, é chamada de **imagem direta** de \mathcal{A} por f .

Definição 1.13. $f^{-1}[\mathcal{B}] = \{x \in D(f); f(x) \in \mathcal{B}\}$ é chamado de **imagem inversa** de \mathcal{B} sobre f . $CD(f)$ denota o **contradomínio** de f .

Destas últimas duas, notemos algumas propriedades:

$$f^{-1}[f[\mathcal{A}]] \supseteq \mathcal{A} \cap D(f), f[f^{-1}[\mathcal{B}]] = \mathcal{B} \cap \text{Im}(f) \quad (1.1)$$

$$f[\mathcal{A}_1 \cup \mathcal{A}_2] = f[\mathcal{A}_1] \cup f[\mathcal{A}_2], f^{-1}[\mathcal{B}_1 \cup \mathcal{B}_2] = f^{-1}[\mathcal{B}_1] \cup f^{-1}[\mathcal{B}_2] \quad (1.2)$$

$$f[\mathcal{A}_1 \cap \mathcal{A}_2] \subseteq f[\mathcal{A}_1] \cap f[\mathcal{A}_2], f^{-1}[\mathcal{B}_1 \cap \mathcal{B}_2] = f^{-1}[\mathcal{B}_1] \cap f^{-1}[\mathcal{B}_2] \quad (1.3)$$

$$f[\mathcal{A}] = \emptyset \Rightarrow \mathcal{A} \cap D(f) = \emptyset, f^{-1}[\mathcal{B}] = \emptyset \Rightarrow \mathcal{B} \cap \text{Im}(f) = \emptyset \quad (1.4)$$

$$(f \circ g)^{-1}[\mathcal{A}] = g^{-1}[f^{-1}[\mathcal{A}]] \quad (1.5)$$

Definição 1.14. Sejam $\mathcal{A} \subseteq \mathbb{E}^m$ e $\mathcal{B} \subseteq \mathbb{E}^n$. Uma função $f : \mathcal{A} \rightarrow \mathcal{B}$ é dita **contínua** em $a \in \mathcal{A}$ se para todo $\epsilon > 0$ existir um $\delta(\epsilon, a) > 0$ tal que $f[N_{\delta(\epsilon, a)}(a)] \subseteq N_\epsilon(f(a))$. Uma função é contínua em \mathcal{A} se é contínua em todo ponto de \mathcal{A} .

Definição 1.15. Uma função $f : \mathbb{R} \rightarrow \mathcal{A} \subseteq \mathbb{E}^n$ é dita **contínua à esquerda** de $a \in \mathbb{R}$ se para todo $\epsilon > 0$ existir um $\delta(\epsilon, a) > 0$ tal que para todo $x \in]a - \delta(\epsilon, a), a]$ temos $\|f(a) - f(x)\| < \epsilon$.

Definição 1.16. Uma função $f : \mathbb{R} \rightarrow \mathcal{A} \subseteq \mathbb{E}^n$ é dita **contínua à direita** de $a \in \mathbb{R}$ se para todo $\epsilon > 0$ existir um $\delta(\epsilon, a) > 0$ tal que para todo $x \in [a, a + \delta(\epsilon, a)[$ temos $\|f(x) - f(a)\| < \epsilon$.

Definição 1.17. Se $f : \mathcal{I} \rightarrow \mathbb{E}^n$ é contínua, então f é chamada de **curva**. $f(0)$ é chamado de ponto inicial da curva e $f(1)$ o ponto final.

Definição 1.18. Se a curva $f : \mathcal{I} \rightarrow \mathbb{E}^n$, $n \geq 2$ é sobrejetora, então f é chamada de **curva de preenchimento de espaço**.

Demonstraremos nos capítulos subsequentes alguns casos particulares que se enquadram como curvas de preenchimento de espaço.

Definição 1.19. Dados $\mathcal{A} \neq \emptyset$ e $\mathcal{B} \neq \emptyset$, a notação $|\mathcal{A}| \leq |\mathcal{B}|$ indica existir $f : \mathcal{A} \rightarrow \mathcal{B}$ injetora (ou, equivalentemente, indica existir $g : \mathcal{B} \rightarrow \mathcal{A}$ sobrejetora). A notação $|\mathcal{A}| = |\mathcal{B}|$ indica existir $h : \mathcal{A} \rightarrow \mathcal{B}$ bijetora (caso em que dizemos que \mathcal{A} e \mathcal{B} possuem mesma cardinalidade).

Teorema 1.20 (Cantor-Bernstein). *Se $|\mathcal{A}| \leq |\mathcal{B}|$ e $|\mathcal{B}| \leq |\mathcal{A}|$, então $|\mathcal{A}| = |\mathcal{B}|$.*

A demonstração do teorema acima foge do escopo deste trabalho. Como G. Cantor demonstrou em 1878, quaisquer dois espaços euclidianos de dimensão finita μ_0, μ_1 , não importa suas dimensões, têm a mesma cardinalidade; conseqüentemente, podem ser colocados em correspondência um para um. Isto significa existir uma bijeção $f : \mu_0 \rightarrow \mu_1$. Em particular, existem as bijeções $g : \mathcal{I} \rightarrow \mathcal{Q}$, $h : \mathcal{I} \rightarrow \mathcal{C}$, etc.

Entretanto, para que uma curva seja de preenchimento de espaço, como definida acima, ela é necessariamente uma função não injetora, conforme propõe o Teorema de Netto a seguir (cuja demonstração para o caso $m = 1$ pode ser encontrada em [11, Teorema 6.4]):

Teorema 1.21 (Netto). *Se f representa uma bijeção de $[0, 1]^m$ em $[0, 1]^n$ e $m \neq n$, então f é necessariamente descontínua.*

O Teorema de Netto implica não haver curvas bijetoras de preenchimento de espaço. Vamos, então, nos capítulos seguintes, procurar demonstrar que os casos particulares de curvas de preenchimento de espaço estudados são funções contínuas e sobrejetoras de domínio no segmento de reta e imagem em uma superfície.

As seguintes definições e teoremas também se farão úteis no desenvolvimento de definições utilizados nos capítulos a seguir:

Definição 1.22. Uma **sequência de funções** é uma sequência $(f_n)_{n \in \mathbb{N}}$, onde cada f_n é uma função.

Definição 1.23. Seja \mathcal{A} um conjunto qualquer. Diz-se que uma sequência de funções $f_n : \mathcal{A} \rightarrow \mathbb{E}^n$ **converge pontualmente** para uma função $f : \mathcal{A} \rightarrow \mathbb{E}^n$ quando, para cada $a \in \mathcal{A}$, a sequência $(f_1(a), f_2(a), \dots, f_n(a), \dots)$ de pontos $f_n(a) \in \mathbb{E}^n$ converge para o ponto $f(a) \in \mathbb{E}^n$.

Definição 1.24. Diz-se que a sequência de funções $f_n : \mathcal{A} \subseteq \mathbb{E}^m \rightarrow \mathbb{E}^n$ **converge uniformemente** em \mathcal{A} para a função $f : \mathcal{A} \subseteq \mathbb{E}^m \rightarrow \mathbb{E}^n$ quando, para todo número real $\epsilon > 0$ dado, for possível obter $n_0 \in \mathbb{N}$ tal que, qualquer que seja $a \in \mathcal{A}$, $n > n_0 \Rightarrow \|f_n(a) - f(a)\| < \epsilon$.

Teorema 1.25. Se uma sequência de funções $f_n : \mathcal{A} \rightarrow \mathcal{B}$, contínuas no ponto $a \in \mathcal{A}$, converge uniformemente em \mathcal{A} para uma função $f : \mathcal{A} \rightarrow \mathcal{B}$, então f é contínua no ponto a .

Demonstração. Para provar que f é contínua no ponto a , supomos dado $\epsilon > 0$ e escolhemos um número natural n_0 tal que $n > n_0$ implica $\|f_n(x) - f(x)\| < \frac{\epsilon}{3}$ para todo $x \in \mathcal{A}$. Como f_{n_0+1} é contínua no ponto a , existe $\delta > 0$ tal que $\|x - a\| < \delta$ em \mathcal{A} implica $\|f_{n_0+1}(x) - f_{n_0+1}(a)\| < \frac{\epsilon}{3}$. Então, para todo $x \in \mathcal{A}$ com $\|x - a\| < \delta$, temos:

$$\begin{aligned} \|f(x) - f(a)\| &\leq \|f(x) - f_{n_0+1}(x)\| + \|f_{n_0+1}(x) - f_{n_0+1}(a)\| + \|f_{n_0+1}(a) - f(a)\| < \\ &< \frac{\epsilon}{3} + \frac{\epsilon}{3} + \frac{\epsilon}{3} = \epsilon. \end{aligned}$$

□

Para o seguinte teorema, faz-se necessário introduzir as seguintes nomenclaturas:

Definição 1.26. Dados um conjunto $\mathcal{A} \subseteq \mathbb{R}$ e $x \in \mathbb{R}$, dizemos que x é **limitante superior** de \mathcal{A} se, para todo $a \in \mathcal{A}$, $a \leq x$. Um conjunto que admite um limitante superior é dito **limitado superiormente**.

Definição 1.27. Dado um conjunto $\mathcal{A} \subseteq \mathbb{R}$ não vazio e limitado superiormente, seja $\mathcal{B} \subseteq \mathbb{R}$ o conjunto de todos os limitantes superiores de \mathcal{A} . Definimos o **supremo** de \mathcal{A} como sendo o menor elemento de \mathcal{B} . (O Axioma da Completude garante que o supremo de \mathcal{A} sempre existe nestas condições.)

Teorema 1.28. *Seja $\mathcal{I} \supseteq \mathcal{I}_1 \supseteq \dots \supseteq \mathcal{I}_{n-1} \supseteq \mathcal{I}_n \supseteq \dots$ uma seqüência de intervalos fechados encaixantes, de extremidades $a_j < b_j$, com $\mathcal{I}_j \neq \emptyset$ e $|b_j - a_j| \rightarrow 0$ quando $n \rightarrow \infty$. Então $\bigcap_{n \in \mathbb{N}} \mathcal{I}_n \neq \emptyset$; além disso, existe um único ponto $a \in \mathcal{I}$ pertencente a todos os intervalos \mathcal{I}_j .*

Demonstração. Tomemos $\mathcal{I}_n = [a_n, b_n]$. Como $\mathcal{I}_{n+1} \subseteq \mathcal{I}_n$, temos que $a_{n+1} \geq a_n$ e $b_{n+1} \leq b_n$ para todo $n \in \mathbb{N}$. Mais do que isso, por indução podemos mostrar que, se $n > m$, $a_n \geq a_m$ e $b_n \leq b_m$. Disso segue, para todos $m, n \in \mathbb{N}$, que $a_n \leq b_m$. De fato, tomemos, k o máximo entre m e n . Temos então que $a_n \leq a_k < b_k \leq b_m$. Logo, o conjunto $\{a_n : n \in \mathbb{N}\}$ é limitado superiormente, e claramente não vazio. Tomemos a o supremo deste conjunto. Temos que $a_n \leq a$ para todo $n \in \mathbb{N}$. Por outro lado, se existisse n tal que $b_n < a$, por definição de supremo teríamos que $b_n < a_m$ para algum $m \in \mathbb{N}$, que contradiz nossa hipótese inicial. Logo, $a < b_n$ para todo $n \in \mathbb{N}$. Concluimos, então, que $a \in [a_n, b_n]$ para todo n , provando que $a \in \bigcap_{n \in \mathbb{N}} \mathcal{I}_n$, ou seja, a interseção dos intervalos encaixantes é não vazia.

Além disso, no caso em que $\lim_{n \rightarrow \infty} |b_n - a_n| = 0$, podemos ver facilmente que a será o único elemento pertencente à interseção: de fato, se $b \in \bigcap_{n \in \mathbb{N}} \mathcal{I}_n$, teremos $|b - a| \leq |b_n - a_n|$ para todo $n \in \mathbb{N}$. Fazendo n crescer indefinidamente, concluimos que $|b - a|$ deve ser zero, donde $b = a$. \square

Os próximos dois resultados fazem parte da problematização inicial de como se construir uma curva de preenchimento de espaço, problema este analisado e superado por ser precursor:

Teorema 1.29. *Seja o número em base b escrito em representação finita de dígitos $x = [0, d_1 d_2 d_3 \dots d_n d]_b$, com $d \neq 0$, e o número também em base b escrito em representação infinita de dígitos $x' = [0, d_1 d_2 d_3 \dots d_n (d-1) \overline{(b-1)}]_b$. Temos que $x = x'$.*

Demonstração. Note que:

$$x = [0, d_1 d_2 d_3 \dots d_n d]_b = \frac{d_1}{b^1} + \frac{d_2}{b^2} + \frac{d_3}{b^3} + \dots + \frac{d_n}{b^n} + \frac{d}{b^{n+1}} + \frac{0}{b^{n+2}} + \frac{0}{b^{n+3}} + \dots = \sum_{k=1}^{\infty} \frac{d_k}{b^k}.$$

Podemos, então, escrever x' segundo mesmo somatório, como segue:

$$x' = \sum_{k=1}^n \frac{d_k}{b^k} + \frac{d-1}{b^{n+1}} + \sum_{k=n+2}^{\infty} \frac{b-1}{b^k}.$$

Note que a última parcela é a soma de uma progressão geométrica infinita de primeiro termo igual a $\frac{b-1}{b^{n+2}}$ e razão $\frac{1}{b}$. Como $S = a_1 \cdot (1-q)^{-1}$:

$$S = \frac{b-1}{b^{n+2}} \cdot \left(1 - \frac{1}{b}\right)^{-1} = \frac{b-1}{b^{n+2}} \cdot \frac{b}{b-1} = \frac{1}{b^{n+1}}.$$

$$\therefore x' = \sum_{k=1}^n \frac{d_k}{b^k} + \frac{d-1}{b^{n+1}} + \frac{1}{b^{n+1}} = \sum_{k=1}^n \frac{d_k}{b^k} + \frac{d}{b^{n+1}} + \frac{0}{b^{n+2}} + \frac{0}{b^{n+3}} + \dots = x.$$

□

Proposição 1.30. *Sejam $[0, d_1 d_2 d_3 \dots]_b$ e $[0, t_1 t_2 t_3 \dots]_b$ duas representações distintas em base b para um mesmo número. Então uma delas é uma representação finita (ou seja, terminada por uma sequência infinita de zeros).*

Demonstração. Seja $n \in \mathbb{N}$ o menor natural com $d_n \neq t_n$. Supondo, sem perda de generalidade, que $d_n < t_n$, temos que $d_n + 1 \leq t_n \Rightarrow t_n - d_n \geq 1$.

Como $[0, d_1 d_2 d_3 \dots]_b = \sum_{i=1}^{\infty} \frac{d_i}{b^i}$ e $[0, t_1 t_2 t_3 \dots]_b = \sum_{i=1}^{\infty} \frac{t_i}{b^i}$, temos que:

$$\sum_{i=1}^{\infty} \frac{d_i}{b^i} = \sum_{i=1}^{\infty} \frac{t_i}{b^i} \Rightarrow \sum_{i=1}^{n-1} \frac{d_i}{b^i} + \frac{d_n}{b^n} + \sum_{i=n+1}^{\infty} \frac{d_i}{b^i} = \sum_{i=1}^{n-1} \frac{t_i}{b^i} + \frac{t_n}{b^n} + \sum_{i=n+1}^{\infty} \frac{t_i}{b^i}.$$

Como até a $(n-1)$ -ésima posição os dígitos são iguais:

$$\begin{aligned} \frac{d_n}{b^n} + \sum_{i=n+1}^{\infty} \frac{d_i}{b^i} &= \frac{t_n}{b^n} + \sum_{i=n+1}^{\infty} \frac{t_i}{b^i} \Rightarrow \\ \sum_{i=n+1}^{\infty} \frac{d_i}{b^i} &= \frac{t_n - d_n}{b^n} + \sum_{i=n+1}^{\infty} \frac{t_i}{b^i} \Rightarrow \\ \sum_{i=n+1}^{\infty} \frac{d_i}{b^i} &\geq \frac{1}{b^n} + \sum_{i=n+1}^{\infty} \frac{t_i}{b^i}. \end{aligned}$$

Como $d_i \leq b-1$ para todo $i \in \mathbb{N}$, temos que:

$$\sum_{i=n+1}^{\infty} \frac{d_i}{b^i} \leq \sum_{i=n+1}^{\infty} \frac{b-1}{b^i} = (b-1) \cdot \sum_{i=n+1}^{\infty} \frac{1}{b^i} = (b-1) \cdot \frac{1}{(b-1)b^n} = \frac{1}{b^n}.$$

Logo:

$$\frac{1}{b^n} \geq \sum_{i=n+1}^{\infty} \frac{d_i}{b^i} \geq \frac{1}{b^n} + \sum_{i=n+1}^{\infty} \frac{t_i}{b^i} \Rightarrow \sum_{i=n+1}^{\infty} \frac{t_i}{b^i} \leq 0.$$

Mas $\sum_{i=n+1}^{\infty} \frac{t_i}{b^i} \geq 0$, donde conclui-se que $\sum_{i=n+1}^{\infty} \frac{t_i}{b^i} = 0$. Portanto, cada t_i é zero após a n -ésima posição. \square

Para mostrarmos a curva que será adotada no desenvolvimento do software proposto para visualização, mostraremos os passos formais que criaram a primeira curva a ser dita de preenchimento de espaço, a curva de Peano; o método geométrico que possibilitou o desenho de tais curvas, gerando a curva de Hilbert; até a generalização que permite à curva adotar qualquer percurso e reproduzi-lo em todas as demais iterações: a curva de Lebesgue.

Segue-se então como se deu a origem da primeira curva de preenchimento de espaço documentada: a curva de Peano.

CURVA DE PEANO

Segundo Sagan [11], a primeira curva de preenchimento de espaço criada na história se deve a G. Peano, em 1890, construída de maneira analítica [8]. A função projetada por Peano foi a primeira a se ter ciência de fazer uso de um algoritmo capaz de fornecer uma relação entre coordenadas de um intervalo unitário e todos os pontos de uma região plana unitária. Para tanto, ele faz uso da representação ternária para t : $t = [0, t_1 t_2 t_3 \dots]_3, t_j \in \{0, 1, 2\}$ (representação adotada para todos os números com dígitos para este capítulo); e também faz uso de um operador que definiremos a seguir.

2.1 MOTIVAÇÃO

Segundo nossa definição em 1.18, para criarmos uma função com a característica de preenchimento de espaço, devemos nos preocupar com que esta seja sobrejetora em relação à imagem, que aqui ilustraremos como a porção do plano \mathcal{Q} . Um esforço mais imediato de imaginar uma função com tal característica é dada pela função f como segue:

$$f(0, d_1 d_2 d_3 d_4 d_5 d_6 \dots) = \begin{pmatrix} 0, d_1 d_3 d_5 \dots \\ 0, d_2 d_4 d_6 \dots \end{pmatrix}$$

Notemos que esta função transforma os dígitos d_i de qualquer $t \in \mathcal{I}$, escritos em notação, por exemplo, de base 10, em duas sequências de dígitos que irão representar o par de coordenadas (x, y) referente a um ponto de \mathcal{Q} . Esta função é sobrejetora, pois dado qualquer ponto de \mathcal{Q} , podemos reorganizar suas coordenadas de forma a determinar um ponto de \mathcal{I} .

Como um número pode possuir mais de uma representação decimal, para que f seja antes de mais nada uma função, devemos escolher para qual das duas representações a imagem do plano estará associada.

Por exemplo, seja o ponto de \mathcal{Q} imagem do ponto de \mathcal{I} de coordenada $\frac{1}{2}$. Como este número possui como representações decimais $0,499999\dots$ e $0,500000\dots$, supomos a princípio que a função leve esta imagem no primeiro caso (números com representação decimal infinita).

Agora, para que nossa função seja chamada de **curva** de preenchimento de espaço, precisamos, segundo definição 1.17, que nossa função f seja também contínua. Note-mos pelo exemplo que ao aplicar f , obtemos:

$$f(0,499999\dots) = \begin{pmatrix} 0,499\dots \\ 0,999\dots \end{pmatrix} = \begin{pmatrix} 0,5 \\ 1 \end{pmatrix}$$

Mas repare que ao tomarmos, por exemplo, a imagem de $0,50\bar{1}\dots$, teremos a segunda coordenada em $0,011\dots$; ao tomarmos a imagem de $0,5000\bar{0}1\dots$, teremos a segunda coordenada em $0,001\dots$; a de $0,500000\bar{0}01\dots$ terá coordenada em $0,0001\dots$; o que intuitivamente nos mostra que existem números arbitrariamente próximos de $\frac{1}{2}$ à direita cuja segunda coordenada da imagem se aproxima de 0 e não de 1. Logo, f não pode ser contínua à direita.

Caso similar ocorre quando supomos que f tenha sido definida como levar a imagem à coordenada $0,500000\dots$ de \mathcal{I} . Ao aplicar f obtemos:

$$f(0,500000\dots) = \begin{pmatrix} 0,500\dots \\ 0,000\dots \end{pmatrix} = \begin{pmatrix} 0,5 \\ 0 \end{pmatrix}$$

Ao tomarmos, por exemplo, a imagem de $0,49\bar{8}\dots$, teremos a segunda coordenada em $0,999\dots$; ao tomarmos a imagem de $0,4999\bar{8}\dots$, teremos a segunda coordenada em $0,999\dots$; e assim sucessivamente, mostrando que existem números arbitrariamente próximos de $\frac{1}{2}$ à esquerda cuja segunda coordenada da imagem se aproxima de 1 e não de 0. Logo, f não pode ser contínua à esquerda.

Portanto, para qualquer forma de representar a imagem de $\frac{1}{2}$, não existe limite para este ponto no exemplo, o que torna a função descontínua. Logo, f não pode ser descrita como curva de preenchimento de espaço, apesar da sobrejetividade.

De fato, os números com mais de uma representação ocasionarão problema semelhante, gerando inúmeros pontos de descontinuidade. Em geral não se é tão simples

atender a estes critérios, como ilustrado acima. Vejamos como Peano, o precursor de curvas do gênero, conseguiu contornar este problema a seguir, através de um operador. Por seu pioneirismo, curvas de preenchimento de espaço possuem sinonímia com curvas de Peano.

2.2 A FUNÇÃO DE PEANO

Definição 2.1. O operador **complementar sobre dígito** k será definido junto a um dígito ternário t_j da seguinte forma: $kt_j = 2 - t_j$, $t_j \in \{0, 1, 2\}$. Logo, temos que:

$$k0 = 2; \quad k1 = 1; \quad k2 = 0.$$

Definição 2.2. A função $f_P : \mathcal{I} \rightarrow \mathcal{Q}$ definida por $f_P(t) = (\varphi_P(t), \psi_P(t))$, onde:

$$\begin{aligned} \varphi_P(0, t_1 t_2 t_3 t_4 t_5 \dots) &= 0, t_1 (k^{t_2} t_3) (k^{t_2+t_4} t_5) \dots \\ \psi_P(0, t_1 t_2 t_3 t_4 t_5 \dots) &= 0, (k^{t_1} t_2) (k^{t_1+t_3} t_4) \dots \end{aligned}$$

é chamada de **curva de Peano**, onde k^n é a n -ésima iteração do operador k sobre o dígito t_j .

A seguir, vamos explorar algumas propriedades interessantes decorrentes das definições dadas por Peano:

Proposição 2.3. *Temos que o operador k é o seu próprio inverso; isto é, $(k \cdot k)(t) = t$, para todo $t \in \mathcal{I}$.*

Demonstração. Basta notar que $(k \cdot k)(t) = (k)(kt) = (k)(2 - t) = [2 - (2 - t)] = t$. □

Isto implica que k operado um número ímpar de vezes irá alterar o dígito t como se tivesse sido aplicado uma só vez; enquanto que k operado um número par de vezes resulta no mesmo dígito t .

Proposição 2.4. *Números com mais de uma forma de representação possuem imagens iguais.*

Demonstração. Conforme teorema 1.29, alguns ternários possuem mais de uma forma de representação: todo número de representação finita $0, t_1 t_2 t_3 \dots t_n t$, com $t \neq 0$, pode ser reescrito na representação infinita $0, t_1 t_2 t_3 \dots t_n (t - 1)\bar{2}$.

Para provar que ambas as formas possuem a mesma imagem, considere $n = 2m$, $r = t_2 + t_4 + \dots + t_{2m}$ e $s = t_1 + t_3 + \dots + t_{2m-1}$. Note que:

$$\begin{aligned} \varphi_P(0, t_1 t_2 t_3 \dots t_{2m} t) &= 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (k^r t)(\overline{k^r 0}) = \\ \begin{cases} 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t\bar{0} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t & \text{se } r \text{ é par;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-t)\bar{2} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (3-t) & \text{se } r \text{ é ímpar.} \end{cases} \end{aligned}$$

Para a representação infinita:

$$\begin{aligned} \varphi_P(0, t_1 t_2 t_3 \dots t_{2m} (t-1)\bar{2}) &= 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (k^r (t-1))(k^{r+2} 2)(k^{r+4} 2) \dots = \\ \begin{cases} 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (t-1)\bar{2} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t & \text{se } r \text{ é par;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-(t-1))\bar{0} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (3-t) & \text{se } r \text{ é ímpar.} \end{cases} \end{aligned}$$

Um raciocínio análogo é aplicado à outra coordenada da imagem:

$$\begin{aligned} \psi_P(0, t_1 t_2 t_3 \dots t_{2m} t) &= 0, (k^{t_1} t_2)(k^{t_1+t_3} t_4) \dots (k^s t_{2m})(\overline{k^{s+t} 0}) = \\ \begin{cases} 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t_{2m}\bar{0} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t_{2m} & \text{se } s \text{ é par e } t \text{ é par;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t_{2m}\bar{2} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (t_{2m} + 1) & \text{se } s \text{ é par e } t \text{ é ímpar;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-t_{2m})\bar{2} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (3-t_{2m}) & \text{se } s \text{ é ímpar e } t \text{ é par;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-t_{2m})\bar{0} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-t_{2m}) & \text{se } s \text{ é ímpar e } t \text{ é ímpar.} \end{cases} \end{aligned}$$

E para a representação infinita:

$$\begin{aligned} \psi_P(0, t_1 t_2 t_3 \dots t_{2m} (t-1)\bar{2}) &= 0, (k^{t_1} t_2)(k^{t_1+t_3} t_4) \dots (k^s t_{2m})(k^{s+t-1} 2)(k^{s+t+1} 2) \dots = \\ \begin{cases} 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t_{2m}\bar{0} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t_{2m} & \text{se } s \text{ é par e } t \text{ é par;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots t_{2m}\bar{2} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (t_{2m} + 1) & \text{se } s \text{ é par e } t \text{ é ímpar;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-t_{2m})\bar{2} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (3-t_{2m}) & \text{se } s \text{ é ímpar e } t \text{ é par;} \\ 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-t_{2m})\bar{0} & = 0, t_1(k^{t_2} t_3)(k^{t_2+t_4} t_5) \dots (2-t_{2m}) & \text{se } s \text{ é ímpar e } t \text{ é ímpar.} \end{cases} \end{aligned}$$

Procede-se exatamente da mesma forma para o caso em que $n = 2m + 1$, alterando apenas que tem-se 4 possibilidades para φ_P e 2 possibilidades para ψ_P , todas correspondentes entre si.

Nota: na representação finita $0, t_1 t_2 t_3 \dots t_{2m} t$ o dígito t está representando o último algarismo que não é zero. No decorrer das demonstrações, nos deparamos com $(2-t)$ que não altera a posição do dígito t na representação. Mas para qualquer outro t_j , nos deparamos com $(2-t_j)$ e $(3-t_j)$, onde t_j pode ser tanto 0, 1 ou 2; logo, a operação resultante pode resultar em 3, que na representação ternária não é um dígito, mas sim acarreta somar 1 no dígito $(j-1)$ (e em todos os demais subsequentes $(j-2), (j-3), \dots, 1$ caso eles sejam o algarismo 2).

Isto esgota todas as possibilidades das várias representações para um mesmo número, o que prova que números com representações diferentes possuem o mesmo par (φ_P, ψ_P) de imagem. \square

Proposição 2.5. *A curva de Peano é sobrejetora.*

Demonstração. Tomemos $(0, x_1x_2x_3x_4\dots; 0, y_1y_2y_3y_4\dots) \in \mathcal{Q}$. Basta demonstrar que existe $t = 0, t_1t_2t_3t_4\dots \in \mathcal{I}$ tal que $f_P(t) = \begin{cases} 0, x_1x_2x_3x_4\dots \\ 0, y_1y_2y_3y_4\dots \end{cases}$.

Comparando os dígitos correspondentes entre os ternários e convenientemente adotando que $t_0 = 0$, obtemos pela própria definição da função que:

$$x_n = k^{t_0+t_2+t_4+\dots+t_{2n-2}}t_{2n-1} \text{ e } y_n = k^{t_1+t_3+t_5+\dots+t_{2n-1}}t_{2n}.$$

Como k é seu próprio inverso (2.3), temos que:

$$t_{2n-1} = k^{t_0+t_2+t_4+\dots+t_{2n-2}}x_n \text{ e } t_{2n} = k^{t_1+t_3+t_5+\dots+t_{2n-1}}y_n.$$

Com isto, podemos operar dígito a dígito até obter:

$$t_1 = k^{t_0}x_1 = k^0x_1 = x_1 \text{ (funciona pela escolha conveniente de } t_0\text{);}$$

$$t_2 = k^{t_1}y_1 \text{ (funciona pois } t_1 \text{ foi definido na linha anterior);}$$

$$t_3 = k^{t_0+t_2}x_2 \text{ (funciona pois } t_2 \text{ foi definido na linha anterior);}$$

$$t_4 = k^{t_1+t_3}y_2 \text{ (funciona pois } t_3 \text{ foi definido na linha anterior, etc).}$$

...

$$t_{2n-1} = k^{t_0+t_2+t_4+\dots+t_{2n-2}}x_n$$

$$t_{2n} = k^{t_1+t_3+t_5+\dots+t_{2n-1}}y_n.$$

E então obtemos todos os demais infinitos dígitos de t de forma recursiva, o que prova a proposição. \square

Proposição 2.6. *A curva de Peano é contínua.*

Demonstração. Vamos inicialmente provar que φ_P é contínua à esquerda em todo $t_0 \in (0, 1]$. Seja $\epsilon > 0$, e escolha $n \in \mathbb{N}$ tal que $\frac{1}{3^n} < \epsilon$. Tomemos $t_0 = 0, t_1t_2t_3\dots t_{2n}t_{2n+1}\dots$ a representação de t_0 que não é finita e tome $\delta = 0,000\dots 0t_{2n+1}t_{2n+2}\dots$. Já que $t_0 - \delta = 0, t_1t_2t_3\dots t_{2n}$, então a representação ternária de qualquer $t \in (t_0 - \delta, t_0]$ tem de coincidir com a de t_0 nos $2n$ primeiros dígitos, sendo t , com seus demais dígitos,

representado por $t = 0, t_1 t_2 t_3 \dots t_{2n} r_{2n+1} r_{2n+2} \dots$. Considerando a soma $s = t_2 + t_4 + t_6 + \dots + t_{2n}$:

$$\begin{aligned} |\varphi_P(t_0) - \varphi_P(t)| &= |0, t_1(k^{t_2} t_3) \dots (k^s t_{2n+1}) \dots - 0, t_1(k^{t_2} t_3) \dots (k^s r_{2n+1}) \dots| \leq \\ &= \frac{|k^s t_{2n+1} - k^s r_{2n+1}|}{3^{n+1}} + \frac{|k^{s+t_{2n+2}} t_{2n+3} - k^{s+r_{2n+2}} r_{2n+3}|}{3^{n+2}} + \dots \leq \\ &= \frac{2}{3^{n+1}} + \frac{2}{3^{n+2}} + \frac{2}{3^{n+3}} + \dots = \frac{2}{3^{n+1}} \cdot \frac{1}{1 - \frac{1}{3}} = \\ &= \frac{2}{3^{n+1}} \cdot \frac{3}{2} = \frac{1}{3^n} < \epsilon. \end{aligned}$$

Assim, mostramos que φ_P é contínua à esquerda em t_0 .

Para mostrar que φ_P é contínua à direita em $[0, 1)$, seja $\epsilon > 0$, e novamente escolhamos $n \in \mathbb{N}$ tal que $\frac{1}{3^n} < \epsilon$. Escolhendo a representação de t_0 que não termina com uma sequência infinita de dígitos 2 e tomamos $\delta = \frac{1}{3^{2n}} - 0,000\dots 0 t_{2n+1} t_{2n+2} \dots$. Já que

$$\begin{aligned} t_0 + \delta &= 0, t_1 t_2 t_3 \dots t_{2n-1} t_{2n} t_{2n+1} \dots + \frac{1}{3^{2n}} - 0,000\dots 0 t_{2n+1} t_{2n+2} \dots = \\ &= 0, t_1 t_2 t_3 \dots t_{2n-1} (t_{2n} + 1) = 0, t_1 t_2 t_3 \dots t_{2n-1} t_{2n} \bar{2} \end{aligned}$$

então qualquer $t \in [t_0, t_0 + \delta)$ tem de coincidir com t_0 nos $2n$ primeiros dígitos, representando os demais dígitos por $t = 0, t_1 t_2 t_3 \dots t_{2n} r_{2n+1} r_{2n+2} \dots$. Considerando a soma $s = t_2 + t_4 + t_6 + \dots + t_{2n}$:

$$\begin{aligned} |\varphi_P(t) - \varphi_P(t_0)| &= |0, t_1(k^{t_2} t_3) \dots (k^s r_{2n+1}) \dots - 0, t_1(k^{t_2} t_3) \dots (k^s t_{2n+1}) \dots| \leq \\ &= \frac{|k^s r_{2n+1} - k^s t_{2n+1}|}{3^{n+1}} + \frac{|k^{s+r_{2n+2}} r_{2n+3} - k^{s+t_{2n+2}} t_{2n+3}|}{3^{n+2}} + \dots \leq \\ &= \frac{2}{3^{n+1}} + \frac{2}{3^{n+2}} + \frac{2}{3^{n+3}} + \dots = \frac{2}{3^{n+1}} \cdot \frac{1}{1 - \frac{1}{3}} = \\ &= \frac{2}{3^{n+1}} \cdot \frac{3}{2} = \frac{1}{3^n} < \epsilon \end{aligned}$$

Assim, mostramos que φ_P é contínua à direita em t_0 .

Já que φ_P é contínua à esquerda em $(0, 1]$ e à direita em $[0, 1)$, então φ_P é contínua em $[0, 1]$. A continuidade de ψ_P se dá de maneira análoga, provando a proposição. \square

De posse das proposições 2.5 e 2.6, podemos finalmente resumir que:

Proposição 2.7. *A curva de Peano f_P é uma curva de preenchimento de espaço.*

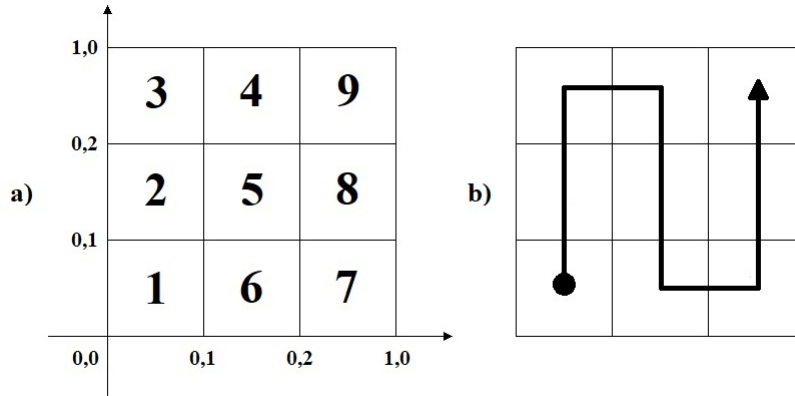


Figura 1: a) Subquadrados de \mathcal{Q} relacionados a $x_1 < x_2 < \dots < x_9$. b) Orientação de preenchimento conforme (a), iniciando em (0; 0) e terminando em (1; 1).

2.3 A CURVA GEOMÉTRICA DE PEANO

Como percebe-se da seção anterior, todas as demonstrações e propriedades exibidas foram dadas sem o uso de ilustrações ou de qualquer conexão geométrica que permitisse o desenho do traço da curva. Seria possível dar a esta curva uma forma de como ela se comporta no plano? Seria possível visualizar esta forma com a qual ela vai percorrendo cada ponto do plano até preenchê-lo por completo?

Essas perguntas não demorariam a ser respondidas. Um ano depois de sua publicação, em 1891, D. Hilbert propôs uma interpretação capaz de desenhá-la de maneira recursiva. Seu raciocínio pode ter sido tão simples quanto se mostra a seguir.

Tome $x_1 = 0,00t_3t_4t_5 \dots \in \mathcal{I}$: aplicando a função de Peano, tem-se as coordenadas:

$$f_P(x_1) = \begin{cases} 0,0(k^0t_3)(k^{0+t_4}t_5) \dots \\ 0,(k^00)(k^{0+t_3}t_4) \dots \end{cases} = \begin{cases} 0,0t_3(k^{t_4}t_5)(k^{t_4+t_6}t_7) \dots \\ 0,0(k^{t_3}t_4)(k^{t_3+t_5}t_6) \dots \end{cases}$$

Particionando \mathcal{Q} em 9 subquadrados congruentes, temos que cada lado mede $[0, 1]_3$ unidades, donde nota-se que $f_P(x_1) \in [0, 0; 0, 1] \times [0, 0; 0, 1]$ pertence ao subquadrado de rótulo 1 da Figura 1a. Note que todo número da forma de x_1 é levado por f_P a este subquadrado independente da escolha de $t_j, j \geq 3$.

Analogamente, aplicando o mesmo processo e fazendo variar os dois primeiros dígitos por todas as demais combinações possíveis, tem-se:

$$\text{Para } x_2 = 0,01t_3t_4t_5 \dots : f_P(x_2) = \begin{cases} 0,0(k^1t_3)(k^{1+t_4}t_5) \dots \\ 0,(k^01)(k^{0+t_3}t_4) \dots \end{cases} = \begin{cases} 0,0(kt_3)(k^{1+t_4}t_5) \dots \\ 0,1(k^{t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_2) \in [0,0;0,1] \times [0,1;0,2]$ (subquadrado de rótulo 2 da Figura 1a.)

$$\text{Para } x_3 = 0,02t_3t_4t_5 \dots : f_P(x_3) = \begin{cases} 0,0(k^2t_3)(k^{2+t_4}t_5) \dots \\ 0,(k^02)(k^{0+t_3}t_4) \dots \end{cases} = \begin{cases} 0,0t_3(k^{2+t_4}t_5) \dots \\ 0,2(k^{t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_3) \in [0,0;0,1] \times [0,2;1,0]$ (subquadrado de rótulo 3 da Figura 1a.)

$$\text{Para } x_4 = 0,10t_3t_4t_5 \dots : f_P(x_4) = \begin{cases} 0,1(k^0t_3)(k^{0+t_4}t_5) \dots \\ 0,(k^10)(k^{1+t_3}t_4) \dots \end{cases} = \begin{cases} 0,1t_3(k^{t_4}t_5) \dots \\ 0,2(k^{1+t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_4) \in [0,1;0,2] \times [0,2;1,0]$ (subquadrado de rótulo 4 da Figura 1a.)

$$\text{Para } x_5 = 0,11t_3t_4t_5 \dots : f_P(x_5) = \begin{cases} 0,1(k^1t_3)(k^{1+t_4}t_5) \dots \\ 0,(k^11)(k^{1+t_3}t_4) \dots \end{cases} = \begin{cases} 0,1(kt_3)(k^{1+t_4}t_5) \dots \\ 0,1(k^{1+t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_5) \in [0,1;0,2] \times [0,1;0,2]$ (subquadrado de rótulo 5 da Figura 1a.)

$$\text{Para } x_6 = 0,12t_3t_4t_5 \dots : f_P(x_6) = \begin{cases} 0,1(k^2t_3)(k^{2+t_4}t_5) \dots \\ 0,(k^12)(k^{1+t_3}t_4) \dots \end{cases} = \begin{cases} 0,1t_3(k^{2+t_4}t_5) \dots \\ 0,0(k^{1+t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_6) \in [0,1;0,2] \times [0,0;0,1]$ (subquadrado de rótulo 6 da Figura 1a.)

$$\text{Para } x_7 = 0,20t_3t_4t_5 \dots : f_P(x_7) = \begin{cases} 0,2(k^0t_3)(k^{0+t_4}t_5) \dots \\ 0,(k^20)(k^{2+t_3}t_4) \dots \end{cases} = \begin{cases} 0,2t_3(k^{t_4}t_5) \dots \\ 0,0(k^{2+t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_7) \in [0,2;1,0] \times [0,0;0,1]$ (subquadrado de rótulo 7 da Figura 1a.)

$$\text{Para } x_8 = 0,21t_3t_4t_5 \dots : f_P(x_8) = \begin{cases} 0,2(k^1t_3)(k^{1+t_4}t_5) \dots \\ 0,(k^21)(k^{2+t_3}t_4) \dots \end{cases} = \begin{cases} 0,2(kt_3)(k^{1+t_4}t_5) \dots \\ 0,1(k^{2+t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_8) \in [0,2;1,0] \times [0,1;0,2]$ (subquadrado de rótulo 8 da Figura 1a.)

$$\text{Para } x_9 = 0,22t_3t_4t_5 \dots : f_P(x_9) = \begin{cases} 0,2(k^2t_3)(k^{2+t_4}t_5) \dots \\ 0,(k^22)(k^{2+t_3}t_4) \dots \end{cases} = \begin{cases} 0,2t_3(k^{2+t_4}t_5) \dots \\ 0,2(k^{2+t_3}t_4) \dots \end{cases} .$$

$\therefore f_P(x_9) \in [0,2;1,0] \times [0,2;1,0]$ (subquadrado de rótulo 9 da Figura 1a.)

Temos então como determinar com precisão em qual região do quadrado a imagem de determinado x irá recair, baseado apenas nos seus dois primeiros dígitos. Mais do que isso: como $x_1 < x_2 < \dots < x_9$, fazendo um segmento orientado que percorre do quadrado 1 ao quadrado 9, temos como entender por quais caminhos a curva vai "preenchendo o espaço", conforme Figura 1b.

Este raciocínio pode ser estendido: tomando os dois próximos dígitos de x (t_3 e t_4), podemos também calcular sua imagem baseado nos valores que podem assumir,

independentemente dos valores de $t_j, j \geq 5$, conforme ilustraremos para os números $x_{11} = 0,0000t_5t_6t_7\dots$ e $x_{12} = 0,0001t_5t_6t_7\dots$:

$$f_P(x_{11}) = \begin{cases} 0,0(k^00)(k^{0+0}t_5)(k^{0+0+t_6}t_7)\dots \\ 0,(k^00)(k^{0+0}0)(k^{0+0+t_5}t_6)\dots \end{cases} = \begin{cases} 0,00t_5(k^{t_6}t_7)\dots \\ 0,00(k^{t_5}t_6)\dots \end{cases}$$

$$\therefore f_P(x_{11}) \in [0,00;0,01] \times [0,00;0,01].$$

$$f_P(x_{12}) = \begin{cases} 0,0(k^00)(k^{0+1}t_5)(k^{0+1+t_6}t_7)\dots \\ 0,(k^00)(k^{0+0}1)(k^{0+0+t_5}t_6)\dots \end{cases} = \begin{cases} 0,00(kt_5)(k^{1+t_6}t_7)\dots \\ 0,01(k^{t_5}t_6)\dots \end{cases}$$

$$\therefore f_P(x_{12}) \in [0,00;0,01] \times [0,01;0,02].$$

Deixamos os demais próximos pontos a cargo do leitor, visto que o processo segue-se por exaustão. Com isso geram-se mais 9 subquadrados para cada um dos quadrados gerados anteriormente, perfazendo um total de $3^2 \times 3^2 = 27$ novos subquadrados em cada região, relacionados a 27 partições de \mathcal{I} , conforme Figura 2a.

Repetindo este processo *ad infinitum*, cada dois dígitos de x ocasiona uma nova iteração de subdivisões de \mathcal{Q} ; portanto, particionando \mathcal{I} em 3^{2n} subintervalos, teremos relacionados 3^{2n} subregiões de \mathcal{Q} , conforme ilustra a Figura 2 para $n = 1$ (a) e $n = 2$ (b).

Como um número cada vez maior de partições de \mathcal{I} fazem os subintervalos se reduzirem a um ponto e as diagonais dos subquadrados se reduzirem a zero (um ponto do espaço), tem-se a correspondência de qualquer ponto do espaço de \mathcal{Q} com um elemento de \mathcal{I} . Isto sugere uma visualização para a curva iteração a iteração, mostrando de que forma seus estágios de construção acabam por preencher mais e mais pedaços do espaço.

Com isso, Hilbert toma nota dos principais elementos da curva: a imagem de x entre uma iteração e a próxima fica contida no subquadrado definido anteriormente; se x_1 e x_2 são números bastante próximos, suas imagens estarão contidas em quadrados próximos; a borda de dois quadrados pertence a dois subintervalos diferentes; e existe sempre um caminho que liga as várias divisões subsequentes, mantendo um padrão de preenchimento. Tais características seriam então generalizadas por Hilbert, fazendo-o propor uma nova curva com tais padrões mantidos, como veremos no capítulo seguinte.

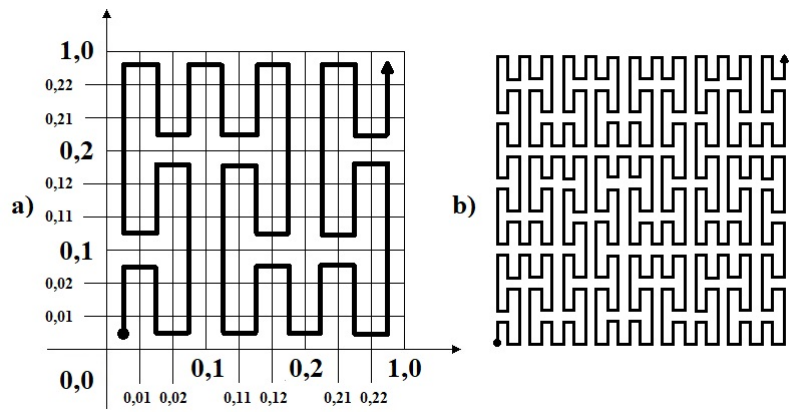


Figura 2: Segunda (a) e terceira (b) iterações na curva de Peano.

CURVA DE HILBERT

De posse das características estudadas na curva de Peano, D. Hilbert propôs em seguida daquele mesmo ano um modelo de curva inspirado no modelo de construção geométrica passo-a-passo através de iterações recursivas, particionando cada vez mais o plano e varrendo estas regiões conforme avançam-se as iterações.

Para tanto, partiu da seguinte heurística: se um intervalo \mathcal{I} pode ser relacionado continuamente em um quadrado \mathcal{Q} , então particionando \mathcal{I} em quatro subintervalos congruentes e \mathcal{Q} em quatro subquadrados congruentes, cada subintervalo pode ser relacionado continuamente em um dos subquadrados. Então, cada subintervalo pode ser particionado em mais 4 partes congruentes, assim como cada subquadrado por ele relacionado pode ser particionado em mais 4 partes congruentes, e o argumento é então repetido *ad infinitum*.

Hilbert provou então haver uma disposição dos subquadrados a cada iteração tal que para cada par de subintervalos particionados adjacentes correspondem subquadrados adjacentes com um lado em comum, de sorte que é preservada a relação de inclusão: se a um intervalo \mathcal{I}_j corresponde um quadrado \mathcal{Q}_j , então a cada partição \mathcal{I}_{jk} de \mathcal{I}_j corresponde um subquadrado \mathcal{Q}_{jk} de \mathcal{Q}_j . A Figura 3 ilustra as três primeiras iterações, com uma linha poligonal partindo do centro de cada subquadrado indicando a ordem em que a sequência de partições de \mathcal{I} é refletida nas partições de \mathcal{Q} .

Podemos então estabelecer, por construção, a seguinte definição:

Definição 3.1. Tomemos o plano \mathcal{Q} particionado em quatro regiões quadradas congruentes $\mathcal{Q}_1^1, \mathcal{Q}_1^2, \mathcal{Q}_1^3, \mathcal{Q}_1^4$ e o intervalo \mathcal{I} particionado em quatro regiões de comprimentos congruentes $\mathcal{I}_1^1, \mathcal{I}_1^2, \mathcal{I}_1^3, \mathcal{I}_1^4$. Seja f_1 a curva que visita estas quatro regiões do plano, fazendo corresponder a cada uma delas uma região do intervalo: $f[\mathcal{I}_1^1] = \mathcal{Q}_1^1, f[\mathcal{I}_1^2] =$

de intervalos define, conforme teorema 1.28, um único ponto $t_0 \in \mathcal{I}$, e então temos $f_H(t_0) = (\xi_0, \eta_0)$. \square

Notemos que se um ponto repousa em um lado de um subquadrado, interior ao quadrado unitário, então ele pertence a dois subquadrados, que podem ou não estarem encaixados em sequência. Caso não estejam, então os pontos deste lado estão relacionados a dois subintervalos diferentes (caso do lado compartilhado pelos subquadrados que parte e que chega a curva na Figura 3a). Porém, se um ponto repousa em um dos vértices de um subquadrado, ponto este interior ao quadrado unitário, então ele obrigatoriamente pertence a quatro subquadrados e, conseqüentemente, a vários subintervalos não necessariamente em sequência. Tal resultado, de que mais de um ponto no intervalo possua mesma imagem na superfície, faz com que a curva não seja injetora, como previmos segundo o Teorema de Netto em 1.21.

Proposição 3.3. *A função $f_H : \mathcal{I} \rightarrow \mathcal{Q}$ é contínua.*

Demonstração. Particionando \mathcal{I} até a n -ésima iteração, teremos um total de 2^{2n} subintervalos de comprimento $\frac{1}{2^{2n}}$. Tomando dois pontos do intervalo $t_1, t_2 \in \mathcal{I}$ cuja distância um do outro seja menor que este comprimento, $|t_1 - t_2| < \frac{1}{2^{2n}}$, temos que estes pontos necessariamente pertencem a um ou a, no máximo, dois subintervalos consecutivos. Como os subintervalos consecutivos estão relacionados a subquadrados consecutivos, temos que a imagem destes pontos $f(t_1), f(t_2) \in \mathcal{Q}$ pertencem a, no máximo, dois subquadrados consecutivos de lado $\frac{1}{2^n}$, com um lado em comum. Estes dois subquadrados formam um retângulo de diagonal $\frac{\sqrt{5}}{2^n}$, conforme Figura 4. Logo, quando $n \rightarrow \infty$, temos que a distância entre as imagens tende a zero, $\|f_H(t_1) - f_H(t_2)\| \leq \frac{\sqrt{5}}{2^n} \rightarrow 0$. Isto conclui que f_H é contínua. \square

Alternativamente, pode-se notar que a sequência de funções contínuas f_n converge uniformemente para f_H ; assim, pelo teorema 1.25 tem-se a continuidade de f_H .

De posse das proposições 3.2 e 3.3, podemos finalmente resumir que:

Proposição 3.4. *A curva de Hilbert f_H é uma curva de preenchimento de espaço.*

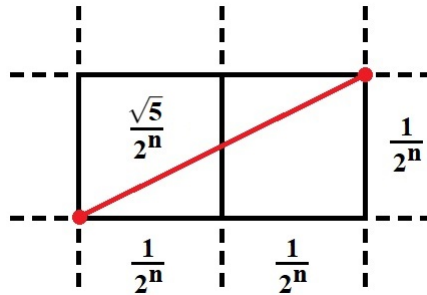


Figura 4: Maior distância possível entre dois pontos pertencentes a dois subquadrados em sequência após a n -ésima iteração.

3.1 ARITMETIZAÇÃO DA CURVA DE HILBERT

Apesar de claramente estabelecido o processo de determinação da curva de Hilbert, percebemos que determinar a imagem de um ponto $t \in \mathcal{I}$ qualquer não consegue ser mais simples que iterando indefinidamente por várias partições de \mathcal{Q} . Determinar a imagem de t é tarefa prática apenas quando t recai como extremidade de algum subintervalo de \mathcal{I} e quando o número de iterações é pequeno; para qualquer outra possibilidade torna-se um processo muito trabalhoso. Hilbert apenas propôs a curva; segundo Sagan [11], foi tarefa de W. Wunderlich mostrar que a construção de tal curva pode ser obtida de maneira mais perceptível.

Ao observarmos a Figura 3, notamos que a curva tem origem no quadrado inferior esquerdo e final no quadrado inferior direito, durante a primeira construção em (a). Nas construções subsequentes em (b) e (c) vemos que esta característica é mantida, sempre iniciando no quadrado inferior mais à esquerda e sempre finalizando no quadrado inferior mais à direita. Como cada um destes quadrados tende para um ponto, considerando \mathcal{Q} no plano cartesiano, temos portanto que a curva de Hilbert se origina na origem do sistema e termina no ponto $(1, 0)$.

Temos de nos preocupar com a orientação de cada subquadrado durante cada processo de iteração de forma que o ponto final do percurso de um subquadrado coincida com o ponto inicial do subquadrado subsequente, sempre respeitando a ordem com que os subquadrados são tomados na iteração anterior. Na Figura 5, podemos perceber, através de vetores, como cada subquadrado é percorrido através da orientação entre pontos inicial e final de cada subquadrado, durante as duas primeiras iterações. Como existe uma correspondência biunívoca entre os pontos do plano \mathbb{E}^2 com os pon-

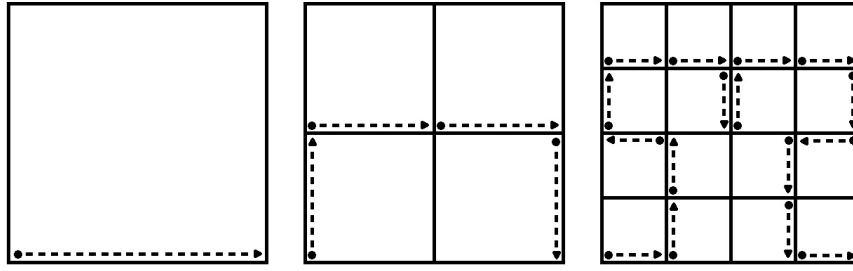


Figura 5: Quadrado unitário percorrido da origem a $(0, 1)$, e orientação dos subquadrados nas duas primeiras iterações.

tos do plano complexo dada por $(a, b) \mapsto (a + bi)$, podemos, para a primeira iteração, pensar em submeter \mathcal{Q} às seguintes transformações lineares, com $z \in \mathbb{C}$:

Para produzirmos o primeiro subquadrado, o qual se origina a curva, temos de contrair \mathcal{Q} à razão $\frac{1}{2}$ em relação à origem: $z' = \frac{1}{2}z$. Isto deixa o vetor ainda apontando para a direita. Para que aponte para cima, aplicamos uma rotação de 90° ao quadrado resultante: $z'' = z'i$. Com isto nosso quadrado muda para o segundo quadrante. Para que volte ao quadrante original, basta refletir em torno do eixo imaginário: $z''' = -\overline{z''}$. Combinando estas três operações em uma, definimos uma transformação composta única \mathfrak{H}_0 dada por: $\mathfrak{H}_0 z = z''' = -\overline{z''} = -\overline{z'i} = \overline{z'i} = \frac{1}{2}\overline{z}i$.

Para produzirmos, na ordem que seguem os vetores, o segundo subquadrado, basta que apliquemos a contração em \mathcal{Q} à mesma razão e desloquemos o quadrado resultante acima do primeiro, a $\frac{1}{2}$ de distância da origem (que é agora o novo lado do primeiro subquadrado), uma vez que a orientação já está correta. Para agilizar, daremos direto a transformação composta resultante por: $\mathfrak{H}_1 z = \frac{1}{2}z + \frac{i}{2}$.

Raciocínio análogo será dado ao terceiro e próximo subquadrado, uma vez que sentido e orientação do vetor são os mesmos do anterior, bastando apenas que mude a colocação do quadrado resultante a $\frac{1}{2}$ de distância acima da origem e também $\frac{1}{2}$ de distância à direita do quadrado anterior. A transformação composta resultante, portanto, é dada por: $\mathfrak{H}_2 z = \frac{1}{2}z + \frac{1}{2} + \frac{i}{2}$.

Por fim, para o último subquadrado precisamos contrair \mathcal{Q} à mesma razão, mas agora mudar a orientação do vetor para baixo, aplicando uma rotação de -90° : $z' = -\frac{1}{2}zi$. Isto muda nosso quadrado para o quarto quadrante. Mais ainda, isto deixa o vetor preso à parede esquerda do quadrado. Note a orientação do vetor dado no terceiro subquadrado, onde este aponta para a parede direita. Para corrigirmos isto

basta refletirmos nosso quadrado em relação ao eixo imaginário: $z'' = -\overline{z'}$, o que faz com que o quadrado ocupe, agora, o terceiro quadrante. Com orientação ajustada, basta então transladarmos este quadrado para a sua posição correta: deslocamos $\frac{1}{2}$ unidade para cima para que ocupe o segundo quadrante, $\frac{1}{2}$ unidade à direita para que sobreponha o primeiro subquadrado, mais $\frac{1}{2}$ unidade à direita para finalizar: $z''' = z'' + \frac{1}{2} + \frac{1}{2} + \frac{i}{2}$. A composta resultante, portanto, será dada por: $\mathfrak{H}_3 z = -\frac{1}{2}\overline{z}i + 1 + \frac{i}{2}$.

As transformações $\mathfrak{H}_0 z$, $\mathfrak{H}_1 z$, $\mathfrak{H}_2 z$ e $\mathfrak{H}_3 z$, quando aplicadas a \mathcal{Q} , recobrem o mesmo espaço ocupado por este e definem a primeira iteração da curva. Quando aplicamos as mesmas quatro transformações a cada um dos subquadrados resultantes da iteração anterior, obtemos a segunda iteração, conforme mostra a Figura 5, e assim sucessivamente.

Tomemos então a seguinte definição: seja $\mathfrak{H}_{q_j} z = \frac{1}{2}H_{q_j} z + \frac{1}{2}h_{q_j}$, $j \in \{0, 1, 2, 3\}$ onde $H_0 z = \overline{z}i$, $H_1 z = z$, $H_2 z = z$, $H_3 z = -\overline{z}i$, $h_0 = 0$, $h_1 = i$, $h_2 = 1 + i$ e $h_3 = 2 + i$. Decompondo as transformações em partes real e imaginária produz o seguinte catálogo de transformações de similaridades:

$$\begin{aligned}\mathfrak{H}_0 \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \frac{1}{2}H_0 \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2}h_0 \\ \mathfrak{H}_1 \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{1}{2}H_1 \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2}h_1 \\ \mathfrak{H}_2 \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{2}H_2 \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2}h_2 \\ \mathfrak{H}_3 \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 2 \\ 1 \end{pmatrix} = \frac{1}{2}H_3 \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2}h_3\end{aligned}\tag{3.1}$$

Estas são as fórmulas de Wunderlich apresentou. Pela natureza do processo de geração da curva de Hilbert, temos que será bastante simples proceder os cálculos se assumirmos $t \in \mathcal{I}$ na representação quaternária: $t = [0, q_1 q_2 q_3 \dots]_4$, onde $q_j \in \{0, 1, 2, 3\}$.

Assim, t recai no $(q_1 + 1)$ -ésimo subintervalo da primeira partição de \mathcal{I} em quatro subintervalos congruentes. Consequentemente, sua imagem recai no $(q_1 + 1)$ -ésimo subquadrado da primeira partição de \mathcal{Q} em quatro subquadrados congruentes: $f_H(t) \in$

$\mathfrak{H}_{q_1} \mathcal{Q}$. Continuando a análise tal como fizemos com a curva de Peano, temos também que a imagem de t recairá no $(q_2 + 1)$ -ésimo subquadrado da segunda partição de \mathcal{Q} , dentro do $(q_1 + 1)$ -ésimo subquadrado $\mathfrak{H}_{q_1} \mathcal{Q}$ da primeira partição, nomeadamente $f_H(t) \in \mathfrak{H}_{q_1} \mathfrak{H}_{q_2} \mathcal{Q}$.

Continuando este processo *ad infinitum*, chegamos na definição:

$$f_H(t) = \lim_{n \rightarrow \infty} \mathfrak{H}_{q_1} \mathfrak{H}_{q_2} \mathfrak{H}_{q_3} \dots \mathfrak{H}_{q_n} \mathcal{Q} \quad (3.2)$$

onde o sinal de inclusão \in foi trocado pelo sinal de igual uma vez que os subquadrados reduzem-se a pontos.

Por substituições sucessivas de H_j, h_j em 3.2, obtemos que, para uma representação finita de t e com $H_{q_0} z = z$, temos:

$$\begin{aligned} \mathfrak{H}_{q_1} \mathfrak{H}_{q_2} \mathfrak{H}_{q_3} \dots \mathfrak{H}_{q_n} \mathcal{Q} &= \frac{1}{2} H_{q_1} \left(\mathfrak{H}_{q_2} \mathfrak{H}_{q_3} \dots \mathfrak{H}_{q_n} \mathcal{Q} \right) + \frac{1}{2} h_{q_1} = \\ &= \frac{1}{2} H_{q_1} \left[\frac{1}{2} H_{q_2} \left(\mathfrak{H}_{q_3} \dots \mathfrak{H}_{q_n} \mathcal{Q} \right) + \frac{1}{2} h_{q_2} \right] + \frac{1}{2} h_{q_1} = \dots \\ &= \frac{1}{2^n} H_{q_1} H_{q_2} \dots H_{q_n} + \frac{1}{2^n} H_{q_1} H_{q_2} \dots H_{q_{n-1}} h_{q_n} + \frac{1}{2^{n-1}} H_{q_1} H_{q_2} \dots H_{q_{n-2}} h_{q_{n-1}} + \dots + \frac{1}{2} h_{q_1} = \\ &= \sum_{j=1}^n \frac{1}{2^j} H_{q_0} H_{q_1} H_{q_2} H_{q_3} \dots H_{q_{j-1}} h_{q_j} \end{aligned}$$

Consequentemente, $f_H(0, q_1 q_2 q_3 \dots q_n) = \sum_{j=1}^n \frac{1}{2^j} H_{q_0} H_{q_1} H_{q_2} H_{q_3} \dots H_{q_{j-1}} h_{q_j}$ e, em vista da continuidade de f_H :

$$f_H(0, q_1 q_2 q_3 \dots) = \sum_{j=1}^{\infty} \frac{1}{2^j} H_{q_0} H_{q_1} H_{q_2} H_{q_3} \dots H_{q_{j-1}} h_{q_j} \quad (3.3)$$

Analisemos então como simplificar esta fórmula. Primeiro, notemos que $H_1 z = H_2 z = z$, o que nos permite simplificar todos estes índices por acabarem não alterando z . Depois, tomemos nota destas seguintes características:

- $H_0^2 z = H_0 H_0(a + bi) = [\overline{(a + bi)}i]i = [(a - bi)i]i = (b + ai)i = (b - ai)i = a + bi = z;$
- $H_3^2 z = H_3 H_3(a + bi) = -[\overline{-(a + bi)}i]i = -[-(a - bi)i]i = -[-(b + ai)]i = -(-b - ai)i = -(-b + ai)i = -(a - bi)i = a + bi = z;$
- $H_0 H_3 z = H_0 H_3(a + bi) = [\overline{-(a + bi)}i]i = [-(a - bi)i]i = [-(b + ai)]i = (-b - ai)i = (-b + ai)i = -a - bi = -z;$
- $H_3 H_0 z = H_3 H_0(a + bi) = -[\overline{(a + bi)}i]i = -[(a - bi)i]i = -(b + ai)i = -(b - ai)i = -(a + bi) = -a - bi = -z.$

Os dois primeiros itens revelam que a cada par de vezes que utilizarmos H_0 ou H_3 teremos como resultado o próprio z utilizado; os dois últimos itens revelam que não importa se utilizamos primeiro H_0 e em seguida H_3 ou vice-versa, a aplicação independe da ordem, revelando ser uma operação comutativa.

Com base nisto, podemos eliminar então todos os fatores H_1 e H_2 e reorganizar todos os fatores H_0 e H_3 restantes de forma a agrupá-los segundo uma potência, importando apenas se seu expoente resultante é aplicado um número par ou um número ímpar de vezes. Podemos, portanto, reescrever nossa fórmula em 3.3 como segue:

$$f_H(0, q_1 q_2 q_3 \dots) = \sum_{j=1}^{\infty} \frac{1}{2^j} H_0^{e_{0j}} H_3^{e_{3j}} h_{q_j} \quad (3.4)$$

onde e_{kj} indica a paridade do número de vezes em que k precede q_j para $k = 0$ ou 3 ; ou seja, $e_{kj} = 0$ ou 1 .

Observemos também que como $H_0 z = \bar{z}i = -(-\bar{z}i) = -H_3 z$, podemos reescrever o seguinte trecho da equação em 3.4 como segue:

$$H_0^{e_{0j}} H_3^{e_{3j}} h_{q_j} = (-H_3)^{e_{0j}} H_3^{e_{3j}} h_{q_j} = (-1)^{e_{0j}} H_3^{e_{0j}+e_{3j}} h_{q_j}$$

Observando que $q_j = 0$ leva ao termo $H_0^{e_{0j}} H_3^{e_{3j}} h_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, os demais casos se resumem a:

$$h_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}; h_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}; h_3 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}; \text{ donde } h_k = \begin{pmatrix} k-1 \\ 1 \end{pmatrix}.$$

Como podemos eliminar cada par de vezes que H_3 é aplicado, tomemos $d_j = e_{0j} + e_{3j} \pmod{2}$. Desmembrando H_3 em sua forma de matriz, obtemos:

$$(-1)^{e_{0j}} H_3^{d_j} h_{q_j} = (-1)^{e_{0j}} \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}^{d_j} \begin{pmatrix} q_j - 1 \\ 1 \end{pmatrix}$$

Como $d_j = \begin{cases} 0, & \text{se } e_{0j} \equiv e_{3j} \pmod{2} \\ 1, & \text{caso contrário} \end{cases}$, analisemos ambos os valores possíveis para d_j :

$$\begin{aligned}
d_j = 0 &\Rightarrow \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}^0 \begin{pmatrix} q_j - 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \cdot q_j - 1 \\ 1 - 0 \end{pmatrix} = \\
&= \begin{pmatrix} (1 - 0)q_j - 1 \\ 1 - 0 \cdot q_j \end{pmatrix} = \begin{pmatrix} (1 - d_j)q_j - 1 \\ 1 - d_j q_j \end{pmatrix} \\
d_j = 1 &\Rightarrow \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}^1 \begin{pmatrix} q_j - 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 - 1 \\ (1 - q_j) + 0 \end{pmatrix} = \\
&= \begin{pmatrix} (1 - 1)q_j - 1 \\ 1 - 1 \cdot q_j \end{pmatrix} = \begin{pmatrix} (1 - d_j)q_j - 1 \\ 1 - d_j q_j \end{pmatrix}
\end{aligned}$$

Portanto, independente do valor que d_j assuma, e considerando que estamos trabalhando apenas com a imagem de dígitos q_j positivos, transformamos finalmente a equação 3.4 em:

$$f_H(0, q_1 q_2 q_3 \dots) = \sum_{j=1}^{\infty} \frac{1}{2^j} (-1)^{e_{0j}} \begin{pmatrix} (1 - d_j)q_j - 1 \\ 1 - d_j q_j \end{pmatrix} \quad (3.5)$$

Para ilustrar o uso da fórmula concebida, vamos encontrar a imagem do quaternário $q = [0, 301]_4$. Temos pela fórmula em 3.4 que:

$$f_H(0, 301) = \sum_{j=1}^3 \frac{1}{2^j} H_0^{e_{0j}} H_3^{e_{3j}} h_{q_j} = \frac{1}{2^1} H_0^{e_{01}} H_3^{e_{31}} h_3 + \frac{1}{2^2} H_0^{e_{02}} H_3^{e_{32}} h_0 + \frac{1}{2^3} H_0^{e_{03}} H_3^{e_{33}} h_1$$

$$e_{01} = \text{número de 0's antes de } (q_1 = 3) = 0$$

$$e_{31} = \text{número de 3's antes de } (q_1 = 3) = 0$$

$$e_{02} = \text{número de 0's antes de } (q_2 = 0) = 0$$

$$e_{32} = \text{número de 3's antes de } (q_2 = 0) = 1$$

$$e_{03} = \text{número de 0's antes de } (q_3 = 1) = 1$$

$$e_{33} = \text{número de 3's antes de } (q_3 = 1) = 1$$

$$\therefore f_H(0, 301) = \frac{1}{2} H_0^0 H_3^0 h_2 + \frac{1}{4} H_0^0 H_3^1 h_0 + \frac{1}{8} H_0^1 H_3^1 h_3$$

$$d_1 = e_{01} + e_{31} \pmod{2} = 0 + 0 \pmod{2} = 0$$

$$d_2 = e_{02} + e_{32} \pmod{2} = 0 + 1 \pmod{2} = 1$$

$$d_3 = e_{03} + e_{33} \pmod{2} = 1 + 1 \pmod{2} = 0$$

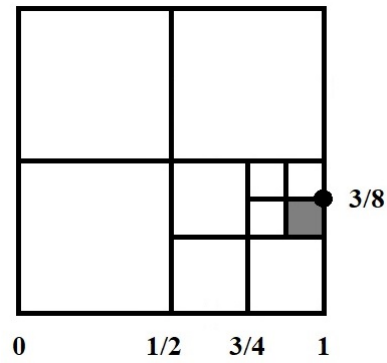


Figura 6: Imagem do ponto de coordenada quaternária 0,203.

Consequentemente, como $h_0 = 0$, temos pela fórmula simplificada em 3.5 que:

$$\begin{aligned}
 f_H(0, 301) &= \frac{1}{2}(-1)^0 \begin{pmatrix} (1-0) \cdot 3 - 1 \\ 1 - 0 \cdot 3 \end{pmatrix} + 0 + \frac{1}{8}(-1)^1 \begin{pmatrix} (1-0) \cdot 1 - 1 \\ 1 - 0 \cdot 1 \end{pmatrix} = \\
 &= \frac{1}{2}(+1) \begin{pmatrix} 2 \\ 1 \end{pmatrix} + \frac{1}{8}(-1) \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \\
 &= \begin{pmatrix} 1 \\ 1/2 \end{pmatrix} - \begin{pmatrix} 0 \\ 1/8 \end{pmatrix} = \begin{pmatrix} 1 \\ 3/8 \end{pmatrix}
 \end{aligned}$$

Esta imagem representa o ponto de entrada no segundo subquadrado da terceira partição dentro do primeiro subquadrado da segunda partição dentro do quarto subquadrado da primeira partição, conforme ilustra a Figura 6.

CURVA DE LEBESGUE

Pelo capítulo anterior, vemos que a curva inicial de Peano pôde ser abstraída de uma fórmula rígida para uma que oferece todas as condições necessárias para uma abstração mais ampla através de processos de partições do quadrado unitário em subquadrados congruentes. Ainda assim, o resultado encontrado, por mais geral que seja, ainda esbarra em algumas limitações do processo de geração como o de orientação dos subquadrados para que a imagem de pontos adjacentes recaiam em subquadrados adjacentes.

Uma análise mais fria deste resultado nos permite intuir que podemos reorganizar os subquadrados no plano que, sendo a curva contínua, ela preencherá o espaço da mesma forma. O que a reorganização ocasiona é um esticamento da curva, visto que para ir de um subquadrado a outro adjacente a curva certamente percorre um espaço menor que a qualquer outro localizado em coordenadas mais distantes no plano.

É neste cenário que H. Lebesgue entra, em 1904, com uma proposta que permitiria a redistribuição dos subquadrados no plano. Preenchendo o espaço de cada subquadrado tal como fizemos com as partições anteriores, cuidamos da reorganização destes subquadrados através de interpolações, como veremos adiante. Tudo isto foi possível graças aos estudos sobre um conjunto que ocupa espaço zero, iniciados por G. Cantor em 1883, conjunto este que ficou mais tarde conhecido por Conjunto de Cantor. É em cima deste conjunto que centraremos nosso próximo estudo.

4.1 CONJUNTO DE CANTOR

Foi no apêndice de uma obra sobre espaços vetoriais infinitos, escrito em 1883, que G. Cantor procurava uma caracterização do contínuo, oferecendo o conjunto de todos

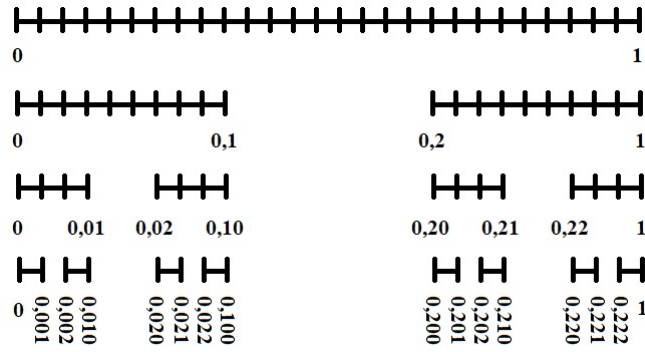


Figura 7: Construção do conjunto de Cantor a partir de \mathcal{I} .

os pontos que podiam ser escritos na forma $\frac{2t_1}{3^1} + \frac{2t_2}{3^2} + \frac{2t_3}{3^3} + \dots, t_j \in \{0, 1\}$ como exemplo de um conjunto perfeito (um conjunto que é igual ao conjunto de todos os seus pontos de acumulação). Este pequeno exemplo acabou por tomar vida própria, servindo de inspiração para os mais remotos temas da Análise Matemática.

Este conjunto, que chamaremos de **conjunto de Cantor**, será denotado e representado por $\Gamma = \{[0, (2t_1)(2t_2)(2t_3) \dots]_3 | t_j = 0 \text{ ou } 1\}$. (A fim de simplificar notação ao longo deste capítulo, tomaremos números t na base 3, salvo mencionado contrário). Veremos a seguir algumas propriedades intrínsecas deste conjunto.

O conjunto de Cantor é conhecido por ser obtido por remoção dos terços intermediários de um intervalo unitário por um número infinito de iterações. Tomando \mathcal{I} , dividindo-o em três partes iguais e removendo o terço central, obtemos os intervalos $[0, \frac{1}{3}]$ e $[\frac{2}{3}, 1]$. Depois, dividimos cada subintervalo resultante em três partes e novamente removemos seus terços centrais, obtendo agora os subintervalos $[0, \frac{1}{9}]$, $[\frac{2}{9}, \frac{1}{3}]$, $[\frac{2}{3}, \frac{7}{9}]$ e $[\frac{8}{9}, 1]$. A seguir continuamos com o mesmo processo em cada subintervalo resultante *ad infinitum*.

Note que a construção do conjunto de Cantor por este processo acaba por eliminar todos os ternários com algum dígito 1: tomando $t = 0, t_1 t_2 t_3 \dots$ e removendo o terço central de \mathcal{I} compreendido em $(\frac{1}{3}, \frac{2}{3})$ durante a primeira iteração, acabamos por remover todo número escrito da forma $0, 1 t_2 t_3 t_4 \dots$ exceto $0, 1$, que reescrevemos da forma $0, 0\bar{2}$. A seguir, durante a segunda iteração, removemos todos os números das formas $0, 01 t_3 t_4 t_5 \dots$ e $0, 21 t_3 t_4 t_5 \dots$ exceto $0, 01$ e $0, 21$, que reescrevemos como $0, 00\bar{2}$ e $0, 20\bar{2}$. Prosseguindo com as iterações *ad infinitum*, somos levados ao seguinte lema:

Lema 4.1. *Um ternário t pertence ao conjunto de Cantor, $t \in \Gamma$, se, e somente se, uma de suas representações não possui nenhum dígito 1 (notando que uma representação finita*

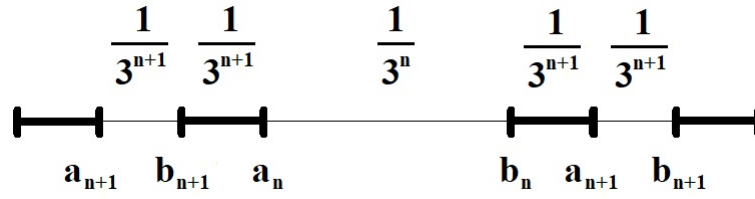


Figura 8: $(n + 1)$ -ésima iteração na construção do conjunto de Cantor, a partir de (a_n, b_n) .

terminada em 1 pode ser reescrita como uma representação terminada em infinitos dígitos 2).

Demonstração. Vimos que, para $n = 1$, temos que um intervalo (a_n, b_n) é removido se $a_1 = 0,1$ e $b_1 = 0,2$. Para $n = 2$, um intervalo é removido se $a_2 = 0,01$ e $b_2 = 0,02$ ou se $a_2 = 0,21$ e $b_2 = 0,22$. Em geral, um intervalo é removido se a_n, b_n podem ser escritos da forma:

$$a_n = 0, (2a_1)(2a_2)(2a_3) \dots (2a_{n-1})1 \text{ e } b_n = 0, (2b_1)(2b_2)(2b_3) \dots (2b_{n-1})2 \quad (4.1)$$

Suponha isto verdadeiro. Depois, teremos na $(n + 1)$ -ésima iteração, de acordo com a figura 8, que:

- $a_{n+1} = a_n - \frac{2}{3^{n+1}} = 0, (2a_1)(2a_2)(2a_3) \dots (2a_{n-1})1 - \frac{2}{3^{n+1}} = 0, (2a_1)(2a_2)(2a_3) \dots (2a_{n-1})01$; e
- $b_{n+1} = a_n - \frac{1}{3^{n+1}} = 0, (2a_1)(2a_2)(2a_3) \dots (2a_{n-1})1 - \frac{1}{3^{n+1}} = 0, (2a_1)(2a_2)(2a_3) \dots (2a_{n-1})02$; ou
- $a_{n+1} = b_n + \frac{1}{3^{n+1}} = 0, (2b_1)(2b_2)(2b_3) \dots (2b_{n-1})2 + \frac{1}{3^{n+1}} = 0, (2b_1)(2b_2)(2b_3) \dots (2b_{n-1})21$; e
- $b_{n+1} = b_n + \frac{2}{3^{n+1}} = 0, (2b_1)(2b_2)(2b_3) \dots (2b_{n-1})2 + \frac{2}{3^{n+1}} = 0, (2b_1)(2b_2)(2b_3) \dots (2b_{n-1})22$.

E com isto recaímos de volta no caso 4.1 com $n + 1$ ao invés de n . Portanto, prova-se por indução que para todo $n \in \mathbb{N}$ o afirmado vale para a n -ésima iteração.

Repare que se um ternário t está no intervalo removido, $t \in (a_n, b_n)$, então ele é da forma $t = 0, (2t_1)(2t_2)(2t_3) \dots (2t_{n-1})1r_{n+1}r_{n+2} \dots$. Se todos r_{n+j} forem iguais a zero, então $t = a_n \notin (a_n, b_n)$; e que se todos r_{n+j} forem iguais a 2, então $t = b_n \notin (a_n, b_n)$. Portanto, existe pelo menos um $r_{n+j} \neq 0$ e pelo menos um $r_{n+j} \neq 2$, o que faz com

que um ternário seja removido da construção de Γ se, e somente se, uma de suas representações contenha um dígito 1, o que prova o lema. \square

Tomemos:

$$\Omega_1 = \left(\frac{1}{3}, \frac{2}{3}\right)$$

$$\Omega_2 = \left(\frac{1}{9}, \frac{2}{9}\right) \cup \left(\frac{7}{9}, \frac{8}{9}\right)$$

$$\Omega_3 = \left(\frac{1}{27}, \frac{2}{27}\right) \cup \left(\frac{7}{27}, \frac{8}{27}\right) \cup \left(\frac{19}{27}, \frac{20}{27}\right) \cup \left(\frac{25}{27}, \frac{26}{27}\right)$$

$$\dots \quad \Omega_n = (a_n^{(1)}, b_n^{(1)}) \cup (a_n^{(2)}, b_n^{(2)}) \cup (a_n^{(3)}, b_n^{(3)}) \cup \dots \cup (a_n^{(2^{n-1})}, b_n^{(2^{n-1})})$$

onde $(a_n^{(k)}, b_n^{(k)})$ são os 2^{n-1} intervalos abertos removidos de \mathcal{I} na n -ésima iteração da construção de Γ . Já que todos Ω_j são abertos, então $\Gamma^c = \mathcal{I} \setminus \Gamma = \bigcup_{j=1}^{\infty} \Omega_j$ é aberto, o que faz de Γ fechado. Logo, Γ contém todos seus pontos de acumulação. De fato, Γ é um conjunto perfeito: é igual ao conjunto Γ' de todos seus pontos de acumulação, conforme prova o lema a seguir:

Lema 4.2. *O conjunto de Cantor é igual ao conjunto de todos os seus pontos de acumulação; ou seja, $\Gamma = \Gamma'$. Mais precisamente, todo ponto da forma $0, (2t_1)(2t_2)(2t_3) \dots (2t_n) \in \Gamma$ é ponto de acumulação à esquerda, todo ponto da forma $0, (2t_1)(2t_2)(2t_3) \dots (2t_n)\bar{2} \in \Gamma$ é ponto de acumulação à direita, e todos os outros pontos de Γ são pontos de acumulação à direita e à esquerda.*

Demonstração. Seja $t = 0, (2t_1)(2t_2)(2t_3) \dots = \sum_{j=1}^{\infty} \frac{2t_j}{3^j} \in \Gamma$, $t_j = 0$ ou 1 . Tomando δ suficientemente grande tal que $\frac{1}{3^\delta} < \epsilon$, vamos definir um ponto $t' = \sum_{j=1}^{\delta} \frac{2t_j}{3^j}$, e reparar que $t' \in \Gamma$. Portanto, $|t - t'| = \sum_{j=\delta+1}^{\infty} \frac{2t_j}{3^j} \leq \sum_{j=\delta+1}^{\infty} \frac{2}{3^j} = \frac{2}{3^{\delta+1}} \cdot \frac{3}{2} = \frac{1}{3^\delta} < \epsilon$, o que prova, para todo $\epsilon > 0$, sempre existir pontos de vizinhança à esquerda de t . Isto faz de todo ponto de $\Gamma \setminus \{0\}$ um ponto de acumulação à esquerda.

De forma análoga, tomando também δ suficientemente grande tal que $\frac{1}{3^\delta} < \epsilon$, vamos definir um ponto $t'' = \sum_{j=1}^{\delta} \frac{2t_j}{3^j} + \sum_{j=\delta+1}^{\infty} \frac{2}{3^j}$, e reparar que $t'' \in \Gamma$. Portanto, $|t'' - t| = \sum_{j=\delta+1}^{\infty} \frac{2 - 2t_j}{3^j} \leq \sum_{j=\delta+1}^{\infty} \frac{2}{3^j} = \frac{2}{3^{\delta+1}} \cdot \frac{3}{2} = \frac{1}{3^\delta} < \epsilon$, o que prova, para todo $\epsilon > 0$, sempre existir pontos de vizinhança à direita de t . Isto faz de todo ponto de $\Gamma \setminus \{1\}$ um ponto de acumulação à direita.

As duas afirmações provam que todo ponto $t \in \Gamma \setminus \{0, 1\}$ é ponto de acumulação à direita e à esquerda, o que prova a proposição. \square

Notemos que, por ser um conjunto criado a partir de \mathcal{I} , vale a inclusão $\Gamma \subset \mathcal{I}$. Se tomarmos a função g definida, para cada $t = [0, (2t_1)(2t_2)(2t_3)\dots]_3 \in \Gamma$, por $g(t) = [0, t_1t_2t_3\dots]_2, t_j = 0$ ou 1 , teremos definido uma sobrejeção de Γ no conjunto $\mathcal{B} = \{[0, b_1b_2b_3\dots]_2 \mid b_j = 0 \text{ ou } 1\}$ de todos os números binários, notando que $\mathcal{I} = \mathcal{B}$. Assim, $|\mathcal{B}| \leq |\Gamma| \leq |\mathcal{I}| = |\mathcal{B}|$; portanto, pelo Teorema de Cantor-Bernstein, existe uma bijeção entre Γ e \mathcal{I} , provando que o conjunto de Cantor tem a mesma cardinalidade do contínuo. Isto é de grande interesse uma vez que possibilita a criação de funções $f : \Gamma \rightarrow \mathcal{Q}$ e $F : \Gamma \rightarrow \mathcal{C}$ tais como:

$$f([0, (2t_1)(2t_2)\dots]_3) = \begin{pmatrix} [0, t_1t_3t_5\dots]_2 \\ [0, t_2t_4t_6\dots]_2 \end{pmatrix} \text{ e } F([0, (2t_1)(2t_2)\dots]_3) = \begin{pmatrix} [0, t_1t_4t_7\dots]_2 \\ [0, t_2t_5t_8\dots]_2 \\ [0, t_3t_6t_9\dots]_2 \end{pmatrix} \quad (4.2)$$

com $t_j \in \{0, 1\}$ (sendo estas funções, por 1.21, claramente não injetoras).

Proposição 4.3. *As funções $f : \Gamma \rightarrow \mathcal{Q}$ e $F : \Gamma \rightarrow \mathcal{C}$ são sobrejetoras.*

Demonstração. Ao tomar um ponto p do quadrado unitário, $p \in \mathcal{Q}$, ou um ponto q do cubo unitário, $q \in \mathcal{C}$, podemos representar suas coordenadas em notação binária:

$$p = \begin{pmatrix} 0, a_1a_2a_3\dots \\ 0, b_1b_2b_3\dots \end{pmatrix} \text{ e } q = \begin{pmatrix} 0, c_1c_2c_3\dots \\ 0, d_1d_2d_3\dots \\ 0, e_1e_2e_3\dots \end{pmatrix}. \text{ Basta então tomarmos}$$

$$t' = [0, (2a_1)(2b_1)(2a_2)(2b_2)(2a_3)(2b_3)\dots]_3 \text{ e } t'' = [0, (2c_1)(2d_1)(2e_1)(2c_2)(2d_2)(2e_2)\dots]_3$$

que teremos, respectivamente, $f(t') = p$ e $F(t'') = q$. \square

Proposição 4.4. *As funções $f : \Gamma \rightarrow \mathcal{Q}$ e $F : \Gamma \rightarrow \mathcal{C}$ são contínuas.*

Demonstração. Vamos supor dois pontos $t, t_0 \in \Gamma$ próximos de tal sorte que $|t - t_0| < \frac{1}{3^{2n}}$. Isto significa que estes pontos possuem invariáveis seus $2n$ primeiros dígitos, tendo $t_0 = 0, (2t_1)(2t_2)(2t_3)\dots(2t_{2n})(2t_{2n+1})\dots$ e $t = 0, (2t_1)(2t_2)(2t_3)\dots(2t_{2n})(2r_{2n+1})\dots$. Supondo, por absurdo, que os números sejam diferentes; tomemos, por exemplo, os dígitos na $2n$ -ésima posição como diferentes: $t_{2n} \neq r_{2n}$. Como $t_j = 0$ ou 1 , temos obrigatoriamente que $|t_{2n} - r_{2n}| = 1$. Temos que $t - t_0 = \frac{2t_{2n} - 2r_{2n}}{3^{2n}} + \frac{2t_{2n+1} - 2r_{2n+1}}{3^{2n+1}} + \dots$, cujo numerador pode unicamente assumir os valores $-2, 0$ ou 2 . Como o módulo da primeira parcela obrigatoriamente terá numerador 2 , então a soma será certamente maior ou igual que uma soma com o restante das parcelas possuindo numerador negativo. Portanto, $|t - t_0| \geq \frac{2}{3^{2n}} - \frac{2}{3^{2n-1}} - \frac{2}{3^{2n-2}} - \dots = \frac{2}{3^{2n}} - \frac{2}{3^{2n+1}} (1 + \frac{1}{3} + \frac{1}{9} + \dots) =$

$\frac{2}{3^{2n}} - \frac{2}{3^{2n+1}} \cdot \frac{3}{2} = \frac{2}{3^{2n}} - \frac{1}{3^{2n}} = \frac{1}{3^{2n}}$, que contraria a hipótese inicial de que $|t - t_0| < \frac{1}{3^{2n}}$.

Aplicando estes pontos à função, temos $f(t) - f(t_0) = \left(\begin{array}{c} \frac{t_{2n+1} - r_{2n+1}}{2^{n+1}} + \frac{t_{2n+3} - r_{2n+3}}{2^{n+2}} + \dots \\ \frac{t_{2n+2} - r_{2n+2}}{2^{n+1}} + \frac{t_{2n+4} - r_{2n+4}}{2^{n+2}} + \dots \end{array} \right)$,

donde $\|f(t) - f(t_0)\| =$

$$\begin{aligned} &= \sqrt{\left(\frac{t_{2n+1} - r_{2n+1}}{2^{n+1}} + \frac{t_{2n+3} - r_{2n+3}}{2^{n+2}} + \dots \right)^2 + \left(\frac{t_{2n+2} - r_{2n+2}}{2^{n+1}} + \frac{t_{2n+4} - r_{2n+4}}{2^{n+2}} + \dots \right)^2} \leq \\ &\leq \sqrt{\left[\frac{1}{2^{n+1}} \left(1 + \frac{1}{2} + \frac{1}{4} + \dots \right) \right]^2 + \left[\frac{1}{2^{n+1}} \left(1 + \frac{1}{2} + \frac{1}{4} + \dots \right) \right]^2} = \\ &= \sqrt{\left(\frac{2}{2^{n+1}} \right)^2 + \left(\frac{2}{2^{n+1}} \right)^2} = \sqrt{\frac{1}{2^{2n}} + \frac{1}{2^{2n}}} = \frac{\sqrt{2}}{2^n} \rightarrow 0 \end{aligned}$$

quando $n \rightarrow \infty$, o que prova a continuidade.

Raciocínio análogo é utilizado para a função F , mudando apenas a quantidade de componentes na norma: $\|F(t) - F(t_0)\| \leq \frac{\sqrt{3}}{2^n} \rightarrow 0$ quando $n \rightarrow \infty$. \square

Com os resultados das proposições 4.3 e 4.4, podemos garantir então que a função que assim relaciona o conjunto de Cantor no quadrado (ou cubo) unitário são sobrejetoras e contínuas (ou seja, definem uma curva de preenchimento de espaço, mesmo com domínio reduzido).

4.2 DEFINIÇÃO DE LEBESGUE PARA CURVA DE PREENCHIMENTO DE ESPAÇO

Em 1904, H. Lebesgue tomou proveito destas propriedades do Conjunto de Cantor para definir uma função que permitira o espalhamento dos subquadrados gerados pelas partições sucessivas, retirando a necessidade do vínculo entre suas arestas, fazendo isto por meio de interpolação linear.

Se (a, b) é um intervalo removido na construção de Γ na n -ésima iteração, então estendemos a função em 4.2, definindo f_I em termos de f naquele intervalo como segue:

$$f_I(t) = \frac{1}{b-a} [f(a)(b-t) + f(b)(t-a)], \quad a < t < b \quad (4.3)$$

Temos, portanto, que f_I é contínua em Γ^C por ser uma função afim definida neste conjunto, e que leva \mathcal{I} em \mathcal{Q} , porque f já faz isso. Demonstraremos como unir esta função com a função definida dentro do conjunto de Cantor com o teorema a seguir:

Teorema 4.5. *A função f_L definida em 4.2 e estendida em Γ^C como em 4.3 é contínua em \mathcal{I} .*

Demonstração. Para provar a continuidade da curva, analisaremos que, tomando qualquer ponto de \mathcal{Q} , existe um ponto de \mathcal{I} que corresponde àquela imagem. Ao se tomar $f_L(t)$, temos duas possibilidades para t : ou $t \in \Gamma$, ou $t \in \Gamma^C$, e como para cada um deles a imagem é definida por uma função diferente (ou por 4.2 ou por 4.3), estudaremos a vizinhança de cada um destes casos separadamente também.

Caso 1. $t \in \Gamma^C$: por 4.3, temos que $f_L(t) = f_I(t)$. Logo, como $f_L = f_I$ num intervalo aberto $I = (a, b)$ contendo t e f_I é contínua, então f_L é contínua em t .

Caso 2. $t \in \Gamma$: então pelo lema 4.2 temos que t pode ser ponto de acumulação à esquerda, ponto de acumulação à direita, ou ponto de acumulação à esquerda e à direita. Analisemos cada um deles.

Para mostrar que f é contínua à esquerda em t , temos que mostrar que para dado um $\epsilon > 0$ qualquer, existe $\delta > 0$ tal que, para todo $t_0 \in (t - \delta, t)$, tem-se $\|f(t) - f(t_0)\| < \epsilon$. Fixemos $\epsilon > 0$. Graças à proposição 4.4, conseguimos um $\delta_0 > 0$ que garante o desejado para todo $t_0 \in (t - \delta_0, t) \cap \Gamma$. Temos, porém, de encontrar um δ que garanta o resultado para todo $t_0 \in (t - \delta, t)$, não apenas para os que estão em Γ .

Para $t_0 \in \Gamma^C$, temos que t_0 pertence a algum intervalo (a, b) à esquerda de t removido durante a construção de Γ , donde, por 4.3, temos:

$$\begin{aligned} f_L(t) - f_L(t_0) &= f(t) - f_I(t_0) = \\ f(t) - \frac{1}{b-a} [f(a)(b-t_0) + f(b)(t_0-a)] &= \\ \frac{1}{b-a} [f(t)(b-a) - f(a)(b-t_0) - f(b)(t_0-a)] &= \\ \frac{1}{b-a} [f(t)b - f(t)a + f(t)t_0 - f(t)t_0 - f(a)b + f(a)t_0 - f(b)t_0 + f(b)a] &= \\ \frac{1}{b-a} [t_0(f(t) - f(b)) - a(f(t) - f(b)) + b(f(t) - f(a)) - t_0(f(t) - f(a))] &= \\ \frac{1}{b-a} [(f(t) - f(b))(t_0 - a) + (f(t) - f(a))(b - t_0)] & \end{aligned}$$

Novamente evocando 4.4, em que f é contínua em Γ , temos que para todo $\epsilon > 0$ existe um $\delta' > 0$ tal que, para qualquer $t_0 \in \Gamma$, $|t - t_0| \leq \delta' \Rightarrow \|f(t) - f(t_0)\| < \epsilon$. Se $t - \delta' \in \Gamma$, defina $\delta'' = \delta'$; se $t - \delta' \in \Gamma^C$, então $t - \delta' \in (a', b')$ que foi removido durante a construção de Γ , donde definimos $\delta'' = b'$. Tomando então $t_0 \in \Gamma^C \cap (t - \delta'', t)$, sejam

$a, b \in \Gamma$ tais que $t_0 \in (a, b) \subset \Gamma^C$. Pela escolha de δ'' , vale que $t - \delta'' \leq a < b < t$; logo teremos $\|f(t) - f(a)\| < \epsilon$ e $\|f(t) - f(b)\| < \epsilon$, donde:

$$\begin{aligned} & \left\| \frac{1}{b-a} [(f(t) - f(b))(t_0 - a) + (f(t) - f(a))(b - t_0)] \right\| \leq \\ & \frac{1}{b-a} [\|f(t) - f(b)\|(t_0 - a) + \|f(t) - f(a)\|(b - t_0)] < \\ & \frac{1}{b-a} [\epsilon(t_0 - a) + \epsilon(b - t_0)] = \\ & \frac{1}{b-a} (t_0 - a + b - t_0) \epsilon = \\ & \frac{1}{b-a} (b - a) \epsilon = \epsilon. \end{aligned}$$

Portanto, tomando $\delta = \min\{\delta_0, \delta''\}$, podemos enfim garantir que, para todo t_0 dado, se $|t - t_0| < \delta$, então $\|f(t) - f(t_0)\| < \epsilon$.

Prova similar é dada para pontos de acumulação à direita: se tivermos $t, t_0 \in \Gamma$, então pela continuidade de f em Γ temos que para todo $\epsilon > 0$ existe $\delta > 0$ tal que exista a vizinhança; caso $t \in \Gamma$ e $t_0 \in \Gamma^C$, temos um argumento análogo ao anterior, com $t < a < b \leq t + \delta$.

Ambas as provas garantem que para todo ponto de acumulação de Γ (ou seja, para todos os pontos de Γ) exista uma vizinhança na interpolação; e os três casos (dois pontos em Γ^C , dois pontos em Γ , ou um em cada) garantem a continuidade para todo ponto em \mathcal{I} . \square

4.3 APROXIMAÇÃO POLIGONAL PARA CURVA DE LEBESGUE

Vamos, com base na definição, utilizar uma aproximação através de polígonos para tentar esboçar a construção da curva de Lebesgue através de cada uma de suas iterações. Vamos primeiro dividir \mathcal{I} nos subintervalos fechados $[0, \frac{1}{9}]$, $[\frac{2}{9}, \frac{1}{3}]$, $[\frac{2}{3}, \frac{7}{9}]$ e $[\frac{8}{9}, 1]$, resultados das duas primeiras iterações na construção de Γ . A seguir, tal como fizemos no capítulo anterior, particionaremos \mathcal{Q} em quatro subquadrados congruentes \mathcal{Q}_{00} , \mathcal{Q}_{01} , \mathcal{Q}_{10} , \mathcal{Q}_{11} e vamos associar cada um deles, respectivamente, aos subintervalos criados, conforme mostra a figura 9.

As transformações que geram estas subdivisões de \mathcal{Q} podem ser representadas tal como Wunderlich apresentou em 3.1, assim:

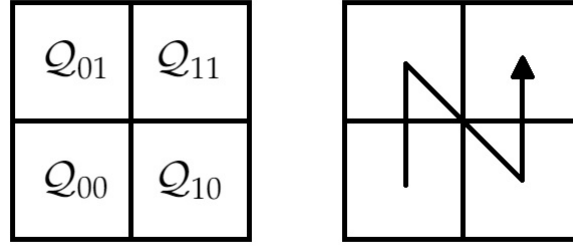


Figura 9: Subdivisões de \mathcal{Q} associada à primeira iteração da construção de Γ .

$$\begin{aligned} \mathfrak{L}_{00} \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \frac{1}{2}\mathcal{Q} + \frac{1}{2}\ell_{00} \\ \mathfrak{L}_{01} \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{1}{2}\mathcal{Q} + \frac{1}{2}\ell_{01} \\ \mathfrak{L}_{10} \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{2}\mathcal{Q} + \frac{1}{2}\ell_{10} \\ \mathfrak{L}_{11} \begin{pmatrix} \xi \\ \eta \end{pmatrix} &= \frac{1}{2} \begin{pmatrix} \xi \\ \eta \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{2}\mathcal{Q} + \frac{1}{2}\ell_{11} \end{aligned}$$

Pela associação proposta entre os subintervalos e os subquadrados, temos que:

$$\begin{aligned} t \in \left[0, \frac{1}{9}\right] &\text{ quando } t = 0,00(2t_3)(2t_4)(2t_5) \dots \\ t \in \left[\frac{2}{9}, \frac{1}{3}\right] &\text{ quando } t = 0,02(2t_3)(2t_4)(2t_5) \dots \\ t \in \left[\frac{2}{3}, \frac{7}{9}\right] &\text{ quando } t = 0,20(2t_3)(2t_4)(2t_5) \dots \\ t \in \left[\frac{8}{9}, 1\right] &\text{ quando } t = 0,22(2t_3)(2t_4)(2t_5) \dots \end{aligned}$$

donde podemos concluir que $\ell_{ij} = \begin{pmatrix} t_i \\ t_j \end{pmatrix}$, sendo $t = 0, (2t_1)(2t_2)(2t_3) \dots$

A seguir, aplicamos estas transformações a cada subquadrado gerado anteriormente, associando às 16 subdivisões de \mathcal{Q} os subintervalos gerados da segunda iteração da construção de Γ conforme figura 10.

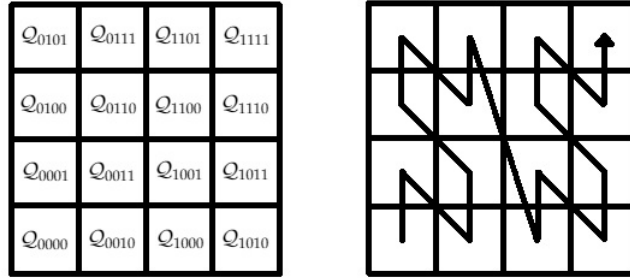


Figura 10: Subdivisões de \mathcal{Q} associada à segunda iteração da construção de Γ .

Continuamos com esse processo *ad infinitum* até obter a imagem $f_L(t)$ de $t \in \Gamma$, tal qual procedemos em 3.3:

$$f_L(0, (2t_1)(2t_2)(2t_3) \dots) = \sum_{j=1}^{\infty} \frac{1}{2^j} \ell_{2^{j-1} 2^j} = \sum_{j=1}^{\infty} \frac{1}{2^j} \begin{pmatrix} t_{2^{j-1}} \\ t_{2^j} \end{pmatrix} = \begin{pmatrix} 0, t_1 t_3 t_5 \dots \\ 0, t_2 t_4 t_6 \dots \end{pmatrix}$$

que é precisamente a definição dada para f_L em 4.2.

Os pontos iniciais dos intervalos fechados na n -ésima iteração da construção de Γ têm a forma $t_i = 0, (2t_1)(2t_2)(2t_3) \dots (2t_{2n})$ e imagem $f_L(t_i) = \begin{pmatrix} 0, t_1 t_3 \dots t_{2n-1} \\ 0, t_2 t_4 \dots t_{2n} \end{pmatrix}$, enquanto os pontos finais nesta iteração têm a forma $t_f = 0, (2t_1)(2t_2)(2t_3) \dots (2t_{2n}) + \frac{1}{3^{2n}} = 0, (2t_1)(2t_2)(2t_3) \dots (2t_{2n})\bar{2}$ e imagem $f_L(t_f) = \begin{pmatrix} 0, t_1 t_3 \dots t_{2n-1}\bar{1} \\ 0, t_2 t_4 \dots t_{2n}\bar{1} \end{pmatrix}$, que está diametralmente oposto em relação a $f_L(t_i)$ no quadrado $\mathcal{Q}_{t_1 t_2 \dots t_{2n}}$ de lado com medida $\frac{1}{2^n}$. Então, a curva de Lebesgue entra o subquadrado $\mathcal{Q}_{t_1 t_2 \dots t_{2n}}$ na borda inferior esquerda e deixa o quadrado na borda superior direita. Por fim, ligamos o ponto de saída de um subquadrado com o ponto de entrada do próximo subquadrado através de uma linha reta, que é precisamente a representação da interpolação linear definida em 4.3 (os intervalos abertos de $\mathcal{I} \setminus \Gamma$ são relacionados a pontos do quadrado através de junções que partem da borda superior direita de um subquadrado até a borda inferior esquerda do subquadrado seguinte).

Com isto, definimos como polígono de aproximação para a curva de Lebesgue o polígono formado pelas diagonais, definidas pelos pontos de entrada e saída de cada subquadrado, mais as junções. A representação do polígono de aproximação para as duas primeiras iterações está representada pela figura 11. Notando que a cada iteração temos que a distância do polígono de aproximação para a curva de Lebesgue é limitada superiormente pelo comprimento da diagonal do subquadrado, $\sqrt{\frac{2}{2^n}}$, e coin-

$$\begin{aligned} f_L(t') &= \frac{1}{1/3} \cdot \left[\begin{pmatrix} 1/6 \\ 1/2 \end{pmatrix} \cdot \frac{1}{6} + \begin{pmatrix} 1/3 \\ 0 \end{pmatrix} \cdot \frac{1}{6} \right] = 3 \cdot \left[\begin{pmatrix} 1/36 \\ 1/12 \end{pmatrix} + \begin{pmatrix} 1/18 \\ 0 \end{pmatrix} \right] = \\ &= 3 \cdot \begin{pmatrix} 1/12 \\ 1/12 \end{pmatrix} = \begin{pmatrix} 1/4 \\ 1/4 \end{pmatrix}. \end{aligned}$$

MODELAGEM COMPUTACIONAL

Para a aprendizagem de curvas que preenchem o espaço, acabamos por lidar com demonstrações e algoritmos muitas vezes intrincados, quando na verdade podemos partir de premissas simples que, além de motivar o interesse pelo tema, contribuem em grande parte para a visualização de resultados e principalmente para o questionamento da maneira e forma como são construídos, levando o aluno a indagar sobre como podemos refinar este conhecimento adquirido e aprender mais com base no que eles experienciam com o concreto.

É sob esta perspectiva que um *software* de visualização atua: é uma ferramenta que permite “traduzir” muito do linguajar matemático de forma que o aluno não só entenda como infira diretamente aspectos de construção da curva e tire suas conclusões.

O projeto utilizado na construção desta obra visa à construção de uma curva de preenchimento de espaço. Tendo um conjunto fechado e limitado bidimensional tomado por um quadrado unitário Q , chamaremos de subquadrados as partições tomadas por divisões sucessivas de Q em regiões congruentes, que serão manipuladas pelo usuário da aplicação.

Para a construção da curva que passará por cada um destes subquadrados gerados e fará o papel de nossa função de preenchimento de espaço, tomaremos os pontos de coordenada central de cada subquadrado gerado dessa partição, e ligaremos dois a dois destes pontos segundo uma ordem também definida pelo nosso usuário.

A sequência de pontos adotada definirá uma curva, e chamaremos de iteração cada um dos níveis que a curva percorrerá, conforme subdividimos ainda mais o plano em cada um destes níveis. Eis a estratégia adotada.

5.1 AMBIENTE DE PROGRAMAÇÃO

O programa será desenvolvido na plataforma *Visual Studio* da *Microsoft*. A escolha se dá pelo fato de a maioria das redes de ensino contarem com um ambiente de informática que conta com o sistema operacional *Windows*. O programa conta com todas as bibliotecas necessárias para compilação do código-fonte neste ambiente.

O programa será desenvolvido em linguagem de programação *C#*, padrão da atual versão do *Visual Studio* utilizada. O programa pode também ser desenvolvido de forma que não faça uso de instaladores ou outros recursos pré-configurados na máquina cliente, tal como ocorre com o uso de outras linguagens de programação.

A distribuição do programa pode ser tão simples quanto apenas transferir seu arquivo executável para as máquinas correspondentes e colocá-lo em execução. A única limitação técnica é, como todo programa que replica imagens *ad infinitum*, a capacidade de memória do computador em que será alocado. Mas o projeto contemplará níveis de replicação o suficiente para que mesmo as máquinas com menor capacidade não se encontrem impossibilitadas de usar o *software*.

5.2 RESTRIÇÕES ADOTADAS NO PROJETO

O projeto contemplará algumas restrições que visam a uma boa utilização pelo usuário de forma a contornar possíveis falhas e percalços.

Por definição, uma reta pode ser traçada por no mínimo dois pontos; logo, não faz sentido adotarmos como subquadrado um quadrado na iteração de ordem 1. Subdivisões de ordem 1 vão gerar exatamente o mesmo quadrado inicial, e o centro deste não será ligado a nenhum outro por só existir um quadrado. Temos, portanto, como configuração mínima a ordem de 2, que criará duas divisões de largura e altura e, com isso, serão gerados quatro outros quadrados congruentes ao primeiro.

Por segurança de implementação, nenhuma curva será gerada enquanto o usuário está fazendo a seleção de pontos que irão compor a função. Muitos níveis de iteração podem já estar sendo executados quando a inclusão ou exclusão de um ponto pode gerar sobrecarga no recarregamento da imagem.

Por padrão, deixaremos o nível inicial de iteração em um para simbolizar a tela inicial que a curva ocupará, com os pontos fornecidos. Em respeito ao limite físico de

processamento e memória característicos de cada computador a que se destinará o uso do programa, o máximo de iterações possível foi definido para dez e o maior valor de ordem de subdivisões em grupos autossimilares foi definido para dez.

5.3 IMPLEMENTAÇÃO DO PROGRAMA

O programa contará com interface única de interação com o usuário, onde nela o usuário pode tanto definir a ordem de divisões do quadrado unitário \mathcal{Q} (número de partições em subquadrados congruentes, que define a quantidade destes na ordem 2^n a cada n iterações) quanto a ordem dos pontos que nossa função percorrerá. Nesta mesma tela, também, estarão os botões responsáveis por iterar entre os diferentes níveis de preenchimento do espaço pela curva.

Para tanto, precisamos de uma classe que servirá de repositório para os atributos necessários à execução (ordem, iteração e conjunto de pontos de nossa função mapa), além dos métodos acessores convenientes; bem como a classe de formulário que servirá para visualização e interação com o usuário.

Nossa classe repositório de dados armazenados pelo usuário será, por padrão, uma classe estática. Não haverá necessidade de instanciação de vários objetos para os propósitos deste projeto, uma vez que apenas um nível de uma curva será visualizado em qualquer dado momento.

O método-chave que permitirá a transformação da ordem dos pontos definida em uma curva é, em termos de programação, um método recursivo: um método que chama a si próprio indefinidamente, até que algum critério de parada seja satisfeito (no caso, o nível de iteração que o usuário irá fornecer definirá quantas vezes nosso método irá recriar a curva).

Muitos dos valores, por definição, passados via método recursivo, serão, por definição, passados por referência, para que a cada passagem do método os valores alterados sejam utilizados de entrada para os próximos níveis de iteração.

Detalhes de como foram construídas estas classes através de seus códigos-fontes estão disponíveis nos Apêndices desta obra. A modelagem de dados adotada para padronização de projeto pode ser consultada em Guedes [3], assim como o guia de linguagem de programação utilizada está disponível em Saade [9].

DEMONSTRAÇÃO DE USO

A figura 12 mostra a tela desenvolvida pela codificação proposta nos anexos.

Nela, percebe-se a tela que será desenhada a curva e o painel de propriedades com os seguintes elementos:

- Um selecionador da ordem de dimensão em que serão subdivididos nosso quadrado unitário Q ;
- Um painel em que o usuário poderá clicar para definir a ordem em que os pontos serão ligados em nossa curva. Esta ordem reflete a associação entre cada subintervalo encaixante de \mathcal{I} com os subquadrados gerados em tela;
- Os botões de iteração entre os diferentes níveis da replicação de nossa curva;
- Um botão de redefinir, que trará de volta nosso painel para as configurações iniciais (iteração nível 1, dimensão 2, nenhum ponto da curva escolhido).

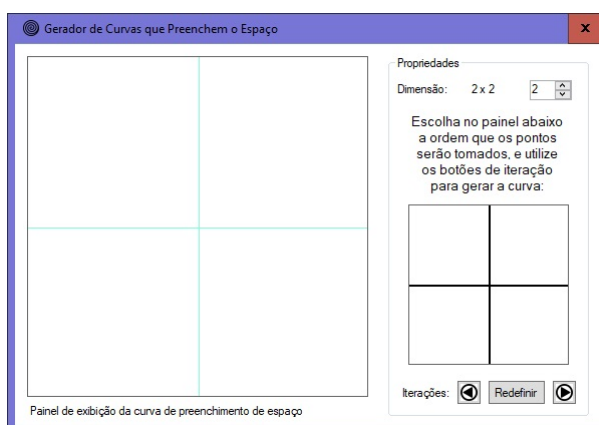


Figura 12: Tela inicial do programa.

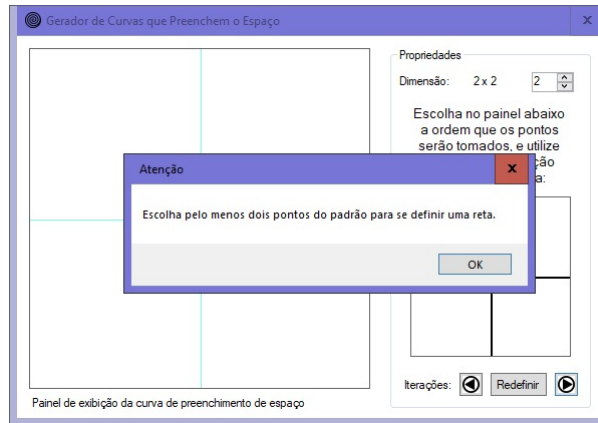


Figura 13: Aviso de ações a serem tomadas pelo usuário.

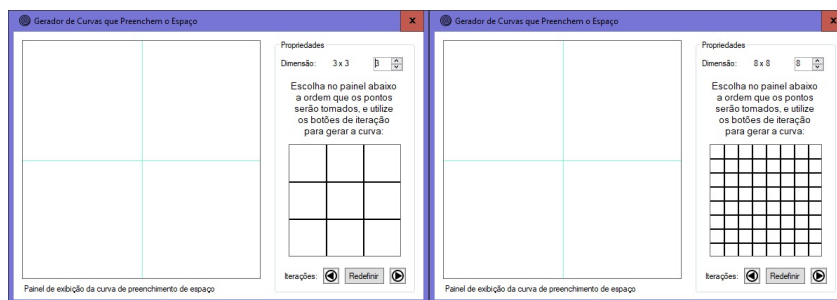


Figura 14: Exemplos de redimensionamento para 3 e para 8.

Notemos o uso de algumas precauções básicas referentes ao uso da ferramenta. Percebendo que o usuário estará livre para manipular a ferramenta como preferir, ele bem pode começar fazendo uso do botão de iteração antes da escolha de qualquer dimensão ou pontos de curva. Isto implica que nenhuma curva será gerada; logo, avisaremos o usuário que a escolha de pontos é o ponto de partida para a visualização de algum resultado. Tal mensagem também é indicada como resultado de pressionar o botão redefinir.

A alteração na escolha da dimensão acarreta na divisão automática na zona de pontos que o usuário pode escolher para a nova função. A alteração da dimensão com pontos previamente selecionados acarreta o descarte automático destes (afinal, se por exemplo escolhermos todos os nove pontos de uma região 3×3 e redimensionarmos para uma região 2×2 , teremos de descartar todos os pontos escolhidos para que o software não tente replicar a curva num espaço inexistente).



Figura 15: Exemplo de sequenciamento das regiões do padrão.

Após a escolha da dimensão que lhe agradar, o usuário pode livremente escolher a ordem em que deseja tomar os subquadrados no painel de escolha. A sequência que o usuário adotar na escolha irá determinar a sequência em que as linhas da curva serão desenhadas no painel principal. Um clique em um dos subquadrados atribuirá a ele um número de sequência, que aparecerá no painel de escolha. Um clique em cima de um quadrado que já possua numeração irá excluí-lo da sequência. Conforme precaução tomada anteriormente, nenhuma curva será desenhada enquanto o usuário estiver sequenciando os pontos, de forma que não há nenhum prejuízo na ordem, escolha ou alteração destes (nem mesmo que haja pontos não escolhidos, conforme veremos mais adiante). A única observação, claro, como já abordado, é o aviso que o usuário receberá quando não houver pontos o suficiente para se desenhar uma linha.

Pressionar os botões de iteração navegará pelos diferentes níveis de replicação da curva. Em um dos últimos níveis de iteração, que exige maior capacidade de processamento do computador, quando houver falha de carregamento da imagem, o botão mostrar também permite o recarregamento da curva. Os botões de navegação são intuitivos e servem para incrementar ou decrementar o nível de iteração.

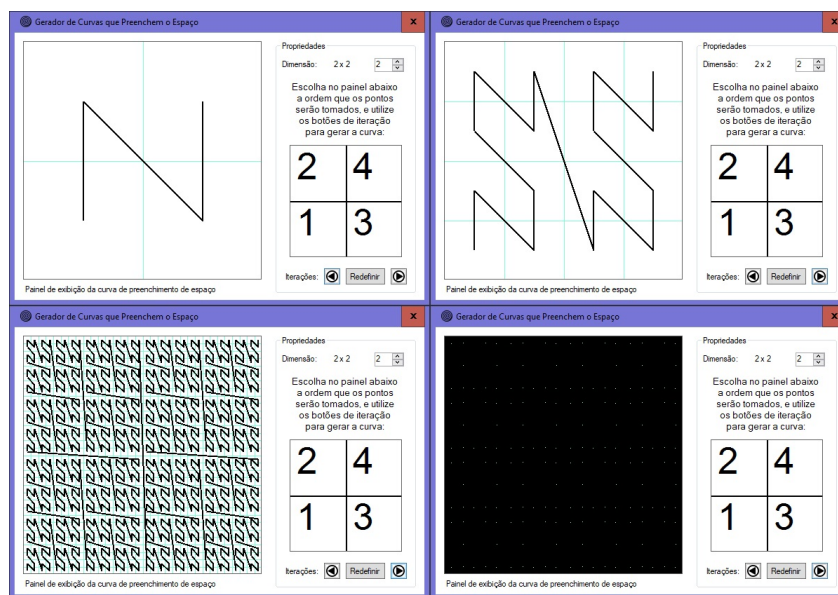


Figura 16: Diferentes níveis de exibição da curva pelos botões de iteração (respectivamente, níveis 1, 2, 5 e 8).

ANÁLISE DE RESULTADOS

O uso da ferramenta permite ao usuário explorar a construção de uma curva que preencha o espaço, e a consequente visualização de que a escolha de pontos, de fato, acarreta em todo o espaço preenchido conforme tendemos o nível de iteração ao infinito. O usuário poderá acompanhar este fato pois a curva passará por mais e mais pontos do plano até que, em alguma iteração, fique indistinguível o que é o plano de fundo e o que é a curva (o resultado é a tela toda preta, cor da curva representada, pois o fundo é branco e a grade de divisões de subquadrados é azul).

7.1 CONTINUIDADE DO PROJETO

O importante neste projeto, além do aprendizado que o professor poderá explorar sobre construção de curvas que preenchem o espaço e noção de continuidade, é também o que os alunos podem tirar de melhoria da própria ferramenta em si.

Por exemplo, este projeto é apenas um caso particular de curvas que preenchem o espaço tomando pontos médios de quadrados autossimilares. Mas podemos expandir esta noção tomando outros pontos quaisquer da região interna de nosso quadrado que as curvas construídas também preencherão o espaço. Podemos inclusive tomar pontos quaisquer de regiões que não sejam um quadrado, que também é um caso particularizado.

O projeto também pode ser estendido a ponto de podermos provar a afirmativa de Cantor de que espaços euclidianos de dimensão finita possuem mesma cardinalidade, não importando a dimensão que estes possuam. Logo, este projeto pode ser aplicado a pontos de uma função que preencha uma porção do plano; podemos tomar pontos

de uma curva que preencham uma porção de um espaço tridimensional ou mesmo exemplos mais intrincados de estruturas n -dimensionais quaisquer.

7.2 PASSEIO DO CAVALO

Tão importante quanto entender os resultados e conceitos por trás da construção de uma curva de preenchimento de espaço é a capacidade do aluno de vislumbrar aplicações. Uma interessante aplicação de uma curva que preenche o espaço está no problema do Passeio do Cavalo.

O cavalo é uma peça do xadrez “que se movimenta em L”. Considerando o tabuleiro de xadrez como uma região 8×8 e o movimento do cavalo como sendo de duas casas em uma direção mais uma casa noutra direção perpendicular, podemos propor o problema de como fazer esta peça, independente de que casa inicial ocupe no tabuleiro, visitar uma única vez todas as casas do tabuleiro com seu movimento válido

Este não é um problema de solução fácil ou imediata, mas existem inúmeras possibilidades de ser testado e infinitas soluções possíveis. E uma vez que uma solução seja encontrada (como mostrado na figura 17), é fácil verificar que o desenho formado pelo trajeto é o de uma curva de preenchimento de espaço, uma vez que o cavalo visita cada casa do tabuleiro.

E, após o trajeto ser desenhado no *software* proposto, vemos, através das iterações, que conforme subdividimos o plano em mais e mais "tabuleiros de xadrez", o percurso do cavalo se torna cada vez mais intrincado até que este acaba por preencher todos os pontos visíveis do quadrado unitário (experimente!).

7.3 FRACTAIS

Conforme sugerido anteriormente durante a demonstração de uso da ferramenta, a curva representada de fato preencherá todo o espaço disponível do painel principal da aplicação **desde que sejam escolhidos todos os pontos de nossa área padrão.**

É claro que o professor já pode partir do pressuposto de que este conhecimento seja necessário para que o aluno visualize, de fato, como a curva preenche todo um dado

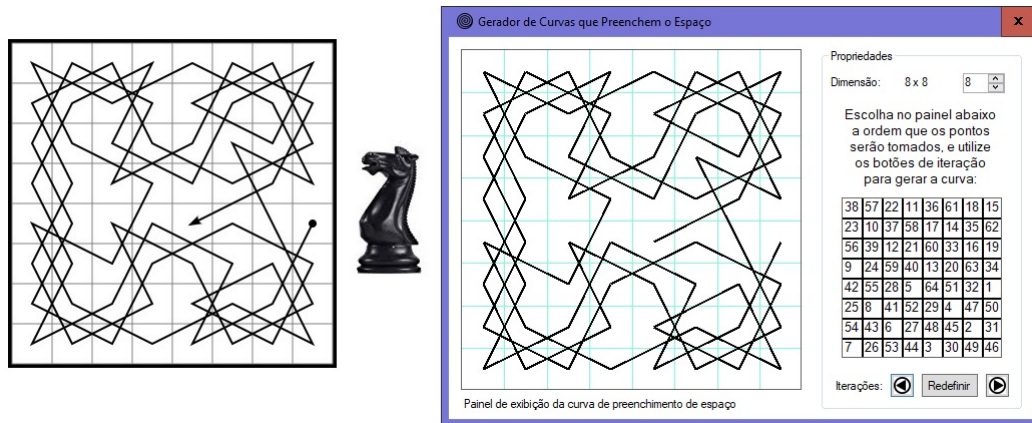


Figura 17: Exemplo de uma possível trajetória do cavalo no tabuleiro.

espaço; como também pode adotar uma postura de fazer o aluno descobrir isto diante de experimentos.

Logo, caso o aluno escolha alguns subquadrados que não todos, o resultado mediante iterações não será o plano completamente preenchido. Em vez disso, o aluno perceberá que, por mais iterações que faça, a curva não sairá dos subquadrados os quais estão vinculados às regiões escolhidas pelo usuário.

Uma curva que preenche apenas parte do plano, se replicando indefinidamente em regiões autossimilares cada vez menores tendendo ao infinito, é a precisa definição de um fractal. Logo, são correlatos o estudo de fractais e o estudo de curvas de preenchimento de espaço.

A figura 18 ilustra o mesmo padrão de dimensão e preenchimento de pontos adotado na figura 16, mas com um dos pontos de padrão faltando. Conforme iteramos pelos níveis desta figura, vemos que a mesma região continua equivalentemente em branco em todas as demais replicações da figura, gerando o famoso fractal Triângulo de Sierpiński.

7.4 ESTUDO DE CASO

Vamos agora analisar como os conceitos estudados podem ser aplicados na modelagem de um problema, proposto pela III Olimpíada Iberoamericana de Matemática Universitária [1]:

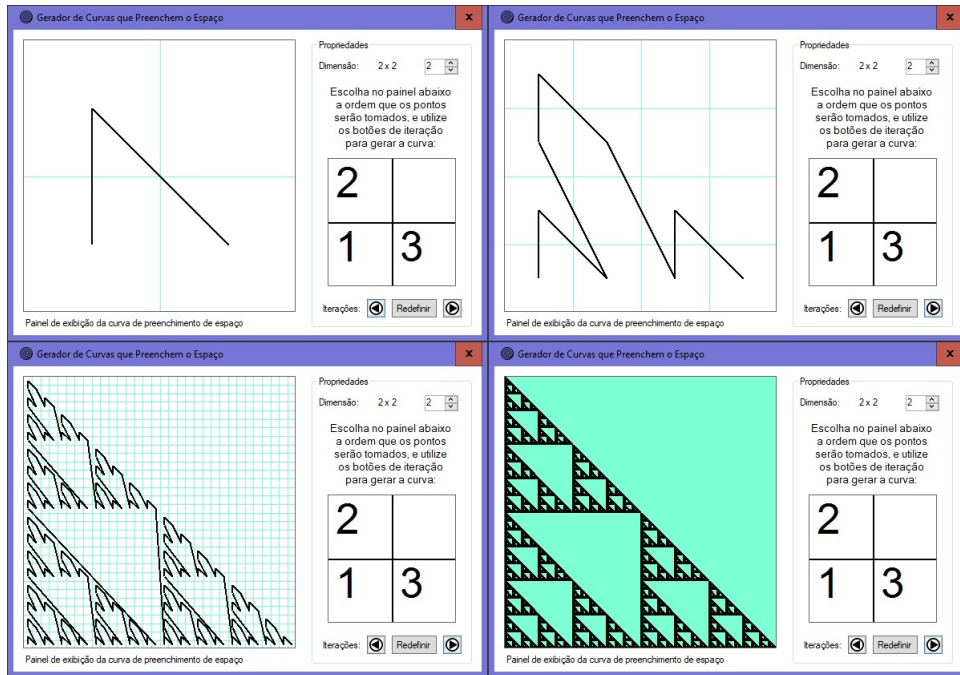


Figura 18: Construção do Triângulo de Sierpiński; níveis 1, 2, 5 e 10.

"No plano se move de qualquer maneira um ponto (um porco) com velocidade não superior a 1 km/h, descrevendo uma curva contínua $\lambda : [0, 1] \rightarrow \mathbb{R}^2$, onde $[0, 1]$ é um intervalo de tempo de uma hora. Sabe-se que o porco encontra-se inicialmente em um quadrado de lado 8 km. No centro deste quadrado se encontra um demônio da Tasmânia cego que não pode saber a posição do porco, mas pode se mover com qualquer velocidade. Encontrar uma curva contínua $\gamma : [0, 1] \rightarrow \mathbb{R}^2$ (o caminho percorrido pelo demônio da Tasmânia) tal que em algum momento de tempo $t \in [0, 1]$ se obtém a igualdade $\gamma(t) = \lambda(t)$, isto é, o demônio da Tasmânia captura o porco independentemente do caminho que este último escolha".

O problema de propor ao estudante uma curva que intercepte outra independentemente do traço que esta apresente tem íntima relação com uma curva de preenchimento de espaço: dada qualquer forma de percurso na região quadrada fornecida que λ possa fazer, ao tomarmos γ de tal sorte que esta percorra todos os pontos desta mesma região, teremos necessariamente uma interseção. O desafio neste problema é garantir a existência de um tempo $t \in [0, 1]$ no qual isto ocorra.

Para construir uma curva que seja capaz de preencher todos os pontos do espaço fornecido, façamos como a teoria apresentada até aqui nos sugere: particionemos nossa

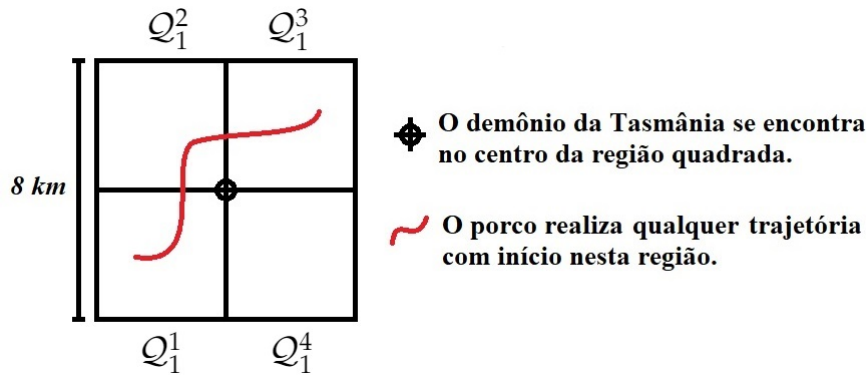


Figura 19: Condições iniciais do problema das duas curvas.

região quadrada Q de 8 km de lado em subregiões de modo que o demônio da Tasmânia demore frações do intervalo de $1h$ a percorrer cada uma. Para facilitar os cálculos, como apresentado até aqui, podemos dividir esta região em quatro subquadrados congruentes $Q_1^1, Q_1^2, Q_1^3, Q_1^4$ para primeira iteração, conforme mostra a figura 19, assim podemos partilhar tempos iguais a serem gastos em cada subregião.

Para não sermos forçados a nos preocupar com a devida orientação do percurso de cada subquadrado conforme fazemos mais e mais iterações de partição dos subquadrados (que nos levaria a definir a curva de Hilbert), podemos definir uma rota padrão que nosso demônio da Tasmânia percorrerá, e aplicar sempre o mesmo passo a cada subregião definida durante as iterações (o que acaba por definir uma curva similar à de Lebesgue). A restrição para aplicar esta solução corresponde a, durante a primeira iteração, não particionar o intervalo de tempo em quatro subintervalos (correspondente ao número de subquadrados particionados), mas sim em um número maior de divisões de tal forma que ele leve menos tempo percorrendo cada região, mas tenha tempo para migrar de uma região para outra. Como o demônio da Tasmânia é dito mover-se com qualquer velocidade, por maior que seja, temos que é possível cumprir esta restrição.

Isto nos permite fazer com que o demônio da Tasmânia percorra como quisermos uma região Q_1^1 (1), tenha tempo hábil para migrar para Q_1^2 (2), percorra do mesmo modo Q_1^2 (3), tenha tempo hábil para migrar para Q_1^3 (4), percorra do mesmo modo Q_1^3 (5), tenha tempo hábil para migrar para a última região Q_1^4 (6), e por fim percorra do mesmo modo Q_1^4 (7). Com esta análise, faz sentido convencionarmos então que particionemos, na primeira iteração, nosso intervalo de tempo em 7 subintervalos con-

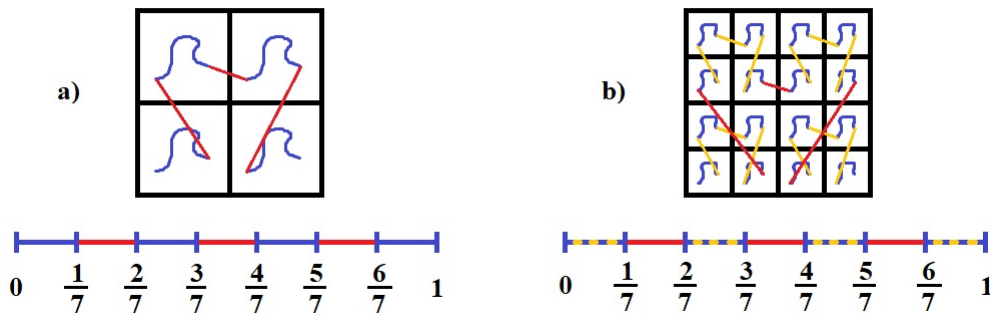


Figura 20: Percorrendo a região segundo uma curva de Lebesgue.

gruentes, relacionando o período de tempo de $[0, \frac{1}{7}]$ a Q_1^1 ; de $[\frac{2}{7}, \frac{3}{7}]$ a Q_1^2 ; de $[\frac{4}{7}, \frac{5}{7}]$ a Q_1^3 ; e de $[\frac{6}{7}, 1]$ a Q_1^4 .

Definido como percorrermos o percurso na primeira iteração, basta que na segunda dividamos cada subintervalo novamente em 7 e relacionemos, na mesma ordem, as frações de tempo com as frações de uma nova partição da região do quadrado percorrida naquele intervalo de tempo em mais quatro subregiões congruentes, tal como ilustra a figura 20: de $[0, \frac{1}{49}]$ a Q_2^1 ; de $[\frac{2}{49}, \frac{3}{49}]$ a Q_2^2 ; e assim sucessivamente para todos Q_n^j , onde n é o número de iterações e $j \in \{i \in \mathbb{N}^*; i \leq 4^n\}$ é uma das subregiões existentes pelas partições naquela iteração.

Repetimos este processo para cada $n \in \mathbb{N}$, definindo um conjunto de Cantor (que não um ternário, mas sim um em que removemos intervalos pares de uma divisão de \mathcal{I} por 7) no qual, como as diagonais de cada subquadrado vinculado tendem a zero, a seqüência de curvas construída converge uniformemente; logo, pelo teorema 1.25, isto define uma função contínua sobre o conjunto de Cantor. Terminamos estendendo a função a \mathcal{I} por interpolações conforme construímos nossa curva de Lebesgue no teorema 4.5.

Mas analisemos agora uma sutileza encontrada no enunciado dado. Enquanto que no instante $t = 0$ o demônio da Tasmânia é dito encontrar-se no centro da região quadrada de lado $8km$, o porco é dito encontrar-se *em qualquer região deste quadrado* se locomovendo a velocidade limitada. Portanto, por mais rápido que o demônio se locomova, se escolhermos como ponto inicial ocupado pelo porco alguma região da borda ou próxima dela, dependendo da velocidade que se tome a perseguição, teremos que o porco pode acabar ultrapassando os limites definidos por esta região quadrada. Precisamos, portanto, revisar nossa curva.

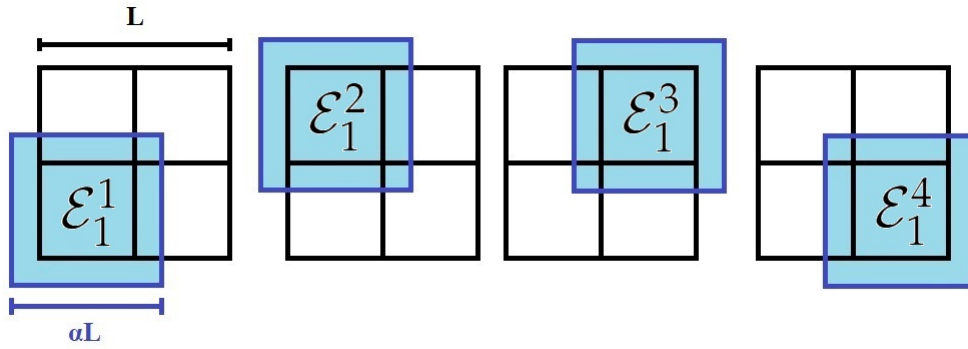


Figura 21: Como recobrir cada subregião da primeira iteração com margem. Note que esta margem representa o máximo que o porco pode percorrer dentro do intervalo de $\frac{1}{7}$ de hora.

Ao invés de aplicarmos a razão de semelhança $\frac{1}{2}$ no percurso da região quadrada e distribuí-lo em cada uma das 4 novas subregiões dada pela divisão na primeira iteração, façamos com que o quadrado seja reduzido a uma razão α de tal modo que ao realocar o novo percurso de modo justo às subregiões, possamos recobrir cada uma com uma certa margem, conforme ilustra a figura 21, onde cada \mathcal{E}_1^k representa um subquadrado expandido de Q_1^k à razão α . Esta margem representa o máximo que o porco pode percorrer para fora da subregião percorrida, dado qualquer ponto inicial que se encontre quando o demônio da Tasmânia adentre esta subregião (incluindo de bordas).

Notemos, pelo esquema da figura, que para esta forma de particionar o plano devemos ter $\frac{1}{2} < \alpha < 1$ para que a nova subregião tenha lado menor que o lado da região da iteração anterior, mas maior que a metade deste para que haja margem. Claro que, ao fazermos isto, estamos criando regiões de percurso redundantes, já visitados antes pelo demônio da Tasmânia, mas fazemos isto para garantir um instante de interseção das curvas.

Note também que, conforme avançamos nas iterações, expandimos um pouco mais a subregião de percurso, conforme ilustra a figura 22. Por mais que haja uma pequena expansão, estamos cada vez mais reduzindo as regiões de caça pelo porco. Como a diagonal de cada uma destas subregiões está tendendo a zero, por mais que o porco se afaste do quadrado original, o demônio da Tasmânia estará se aproximando o suficiente para que haja a captura.

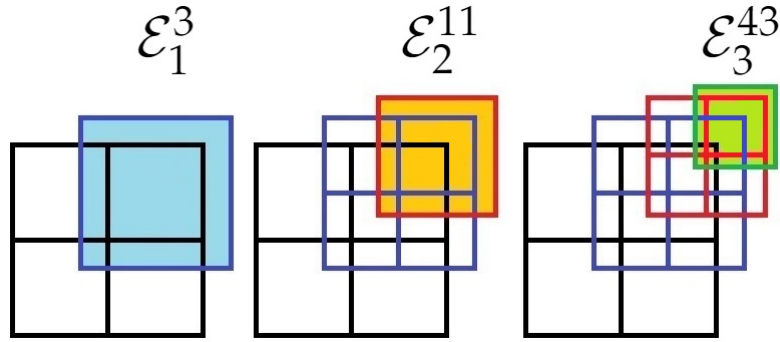


Figura 22: Como ficam as subregiões de perseguição conforme fazemos as três primeiras iterações.

Vejam os porquê que esta perseguição resulta na captura do porco. No instante $t = 0$ sabemos que o demônio da Tasmânia está no centro da região quadrada Q de $8km$ de lado e o porco encontra-se em algum ponto do interior deste quadrado. Isto é verdadeiro. Para que comece a caçada, na primeira iteração temos que o porco pode ou não estar na região \mathcal{E}_1^1 da primeira iteração que definimos.

Se ele estiver, queremos que ele não tenha tempo hábil de deixar esta região. Mas escolheremos um α de modo que isto não aconteça: dentro do sétimo de hora que o demônio da Tasmânia permanecerá em \mathcal{E}_1^1 , o porco também se encontrará. Se o porco permanecer dentro deste quadrado neste tempo, o demônio da Tasmânia se aproximará nas próximas iterações.

Se o porco escapar do quadrado de lado $\frac{L}{2}$, temos que a distância máxima que este percorre é dado pela margem por $d_{max} = \frac{1}{2}(\alpha L - \frac{L}{2}) = (2\alpha - 1)\frac{L}{4}$. Sendo V a velocidade máxima do porco (que é limitada) e t_{min} o tempo mínimo gasto pelo porco para fuga, como a velocidade do porco é uma constante temos que $V = \frac{\Delta S}{\Delta T} \Rightarrow t_{min} = (2\alpha - 1)\frac{L}{4V}$.

Queremos que este tempo seja superior ao tempo T gasto pelo demônio da Tasmânia para captura: $T < (2\alpha - 1)\frac{L}{4V} \Rightarrow 2\alpha - 1 > \frac{4VT}{L} \Rightarrow \alpha > \frac{1}{2} + \frac{2VT}{L}$. Isto parece atender ao requisito de que a constante obedeça $\frac{1}{2} < \alpha < 1$, uma vez que os valores de L , V e T são positivos. E para os dados do enunciado isto se torna possível: temos que a região quadrada possui lado $L = 8km$, a velocidade do porco é limitada em $V = 1km/h$ e o tempo de perseguição está restrito a $T = 1h$, donde $\alpha > \frac{1}{2} + \frac{2 \cdot 1 \cdot 1}{8} = \frac{1}{2} + \frac{1}{4} = \frac{3}{4}$ e $\alpha < 1$.

Note que isto resolve não somente a primeira iteração, mas também todas as outras: durante a perseguição, sabemos que o porco se encontra inicialmente em alguma das regiões $\mathcal{E}_1^1, \mathcal{E}_1^2, \mathcal{E}_1^3, \mathcal{E}_1^4$ para algum tempo $t \in [0, 1]$. Seja \mathcal{E}_1^2 esta região por exemplo,

apenas para efeito de visualização. Sabemos que o demônio da Tasmânia adentra esta região no instante $t' = \frac{2}{7}$ e a deixa no instante $t'' = \frac{3}{7}$, período o qual o porco não consegue escapar por uma escolha conveniente de α .

Incrementando para a segunda iteração, sabemos que existe alguma dentre as regiões $\mathcal{E}_2^5, \mathcal{E}_2^6, \mathcal{E}_2^7, \mathcal{E}_2^8$, partições expandidas de \mathcal{E}_1^2 , da qual o porco não consegue escapar para algum tempo $t \in [\frac{2}{7}, \frac{3}{7}]$. Temos relacionados a estas regiões os subintervalos de tempo $[\frac{14}{49}, \frac{15}{49}]$, $[\frac{16}{49}, \frac{17}{49}]$, $[\frac{18}{49}, \frac{19}{49}]$ e $[\frac{20}{49}, \frac{21}{49}]$. Suponha que \mathcal{E}_2^7 é esta região em que se encontra o porco. Temos que o demônio da Tasmânia adentra esta região no instante $t' = \frac{18}{49}$ e a deixa no instante $t'' = \frac{19}{49}$, período no qual o porco ainda encontra-se encerrado em \mathcal{E}_1^2 . E assim sucessivamente.

Note que a cada passo da iteração, o lado máximo da região que o porco consegue percorrer está sendo multiplicado por $\frac{3}{4} < \alpha < 1$; porém, o tempo de que o porco dispõe para percorrê-lo está sendo multiplicado por $\frac{1}{7}$. Ou seja: estamos dando ao porco cada vez menos tempo para a fuga da região em que se encontra, ao mesmo tempo em que tendemos a distância entre o porco e o demônio da Tasmânia para zero. Com sua mesma velocidade limitada, o porco acaba por não conseguir escapar de nenhuma destas sequências de regiões encaixantes.

A cada uma destas sequências de quadrados está relacionada uma sequência de intervalos encaixantes de tempo. Graças ao teorema mostrado em 1.28, esta sequência acaba por definir um único tempo $t \in \mathcal{I}$, que representa quando ocorre a captura. Para este instante t , temos que $\|\gamma(t) - \lambda(t)\| < \alpha^n L$ para todo $n \in \mathbb{N}$; e, como $\frac{3}{4} < \alpha < 1$, temos que $\alpha^n L \rightarrow 0$ quando $n \rightarrow \infty$. Ou seja, este é o instante t que demonstra onde ocorre $\gamma(t) = \lambda(t)$, o encontro das duas curvas. Isto conclui o problema.

CONSIDERAÇÕES FINAIS

Este trabalho destina-se à precisa definição de curva de preenchimento de espaço. Muito do que se entende por “uma forma que passa por todos os pontos de uma região” é descrito meramente como “curva de Peano”, muito por causa da homenagem dada à curva de preenchimento de espaço percursora do estudo de tais curvas. Fazemos, porém, a distinção entre o que é uma curva de preenchimento de espaço, o que ela deve cumprir para ser classificada como tal, bem como exemplos de curvas distintas. Exemplificamos como o comportamento no plano é bastante específico de uma para a outra, sendo a curva de Peano apenas uma das várias curvas existentes apresentadas.

A obra visa ao leitor um olhar crítico das propriedades matemáticas que cada curva apresenta. Após acompanhar as demonstrações, o leitor pode ser capaz de inferir as condições necessárias para a construção de cada uma das curvas apresentadas, bem como deparar com alguma curva não exposta aqui, mas mesmo assim saber requisitos mínimos para que esta seja classificada como curva de preenchimento de espaço.

Longe de almejar ser uma obra completa acerca do tema, proporciona-se aqui uma experiência ao leitor para que este se interesse pelo tema, busque mais sobre outros tipos de curvas, busque outros exemplos em seu meio ou à natureza. Recomenda-se também aos interessados estudos sobre Topologia, que muito tem a acrescentar sobre estas curvas. A forma de caracterizar espaços métricos e topológicos que são imagens contínuas do intervalo \mathcal{I} diz muito acerca de como são construídas estas curvas.

Este trabalho auxilia a desmistificar um dos grandes entraves no estudo de curvas de preenchimento de espaço: o comportamento da curva no plano. O software gerador de curvas aqui apresentado é capaz de elaborar, iteração por iteração, uma curva de preenchimento de espaço inspirada na curva de Lebesgue para que o usuário possa acompanhar cada etapa de sua construção. Isto agiliza a visualização da curva, pu-

lando etapas de contas por vezes trabalhosas, apresentando um resultado consistente na medida em que é necessária para visualizar o plano preenchido (lembrando que a curva construída tem limite no número de iterações, ao passo que a curva de preenchimento de espaço é o resultado de infinitas iterações).

Por fim, satisfaz ao autor da obra que novos horizontes possam ser enxergados ao fim desta leitura. Com os parâmetros apresentados, o leitor pode se interessar em caracterizar demais curvas que se apresentem por curvas de preenchimento de espaço. O software proposto pode ser aplicado em salas de aula pelo seu fácil manuseio e pelo interesse que desperta ao estudante em enxergar o fenômeno de construções destas curvas. O software, cuja codificação encontra-se presente nos Apêndices, está aberto a modificações na sua estrutura para comportar novas funcionalidades: um aperfeiçoamento no número de iterações; um visualizador para cada tipo de curva (Peano, Hilbert, Lebesgue, etc.); um visualizador para curvas no espaço, não apenas no plano; dentre tantas outras ideias cujo limite é a imaginação.



APÊNDICE A

Código da classe estática **Curva**:

```
01 using System.Collections.Generic;
02 using System.Drawing;

03 namespace SpaceFillingCurves
04 {
05     static class Curva
06     {
07         private static int iteracao = 1;
08         private static int dimensao = 2;
09         private static List<Point> pontos = new List<Point>();

10         public static void alterarIteracao(int _iteracao)
11         {
12             if (_iteracao > 0 && _iteracao < 11)
13                 iteracao = _iteracao;
14         }

15         public static void alterarDimensao(int _dimensao)
16         {
17             iteracao = 1;
18             dimensao = _dimensao;
19             pontos = new List<Point>();
20         }
    }
}
```

```
21     public static void alterarPontos(Point _pontos)
22     {
23         if (pontos.IndexOf(_pontos) > -1)
24             pontos.Remove(_pontos);
25         else
26             pontos.Add(_pontos);
27     }

28     public static int retornarIteracao()
29     {
30         return iteracao;
31     }

32     public static int retornarDimensao()
33     {
34         return dimensao;
35     }

36     public static List<Point> retornarPontos()
37     {
38         return pontos;
39     }

40     public static void redefinir()
41     {
42         iteracao = 1;
43         dimensao = 2;
44         pontos = new List<Point>();
45     }
46 }
47 }
```

B

APÊNDICE B

Código da classe **Formulário**:

```
001 using System;
002 using System.Drawing;
003 using System.Windows.Forms;

004 namespace SpaceFillingCurves
005 {
006     public partial class Form1 : Form
007     {
008         private Graphics tela;
009         private Pen caneta = new Pen(Color.Black, 2);
010         private Pen caneta2 = new Pen(Color.Aquamarine, 1);
011         private SolidBrush pincel = new SolidBrush(Color.Black);

012         public Form1()
013         {
014             InitializeComponent();
015         }

016         private void numericUpDownDimensao_ValueChanged
017             (object sender, EventArgs e)
018         {
019             Curva.alterarDimensao((int) numericUpDownDimensao.Value);
020             labelDimensao.Text = String.Format("{0} x {1}",
```

```
021     numericUpDownDimensao.Value, numericUpDownDimensao.Value);
022     panelPontos.Invalidate();
023 }

024     private void panelPontos_MouseClick
025     (object sender, MouseEventArgs e)
026     {
027         int X = e.X / (panelPontos.Size.Width /
028         Curva.retornarDimensao());
029         int Y = (panelPontos.Size.Height - e.Y) /
030         (panelPontos.Size.Height / Curva.retornarDimensao());
031         Curva.alterarPontos(new Point(X, Y));
032         panelPontos.Invalidate();
033     }

034     private void panelPontos_Paint
035     (object sender, PaintEventArgs e)
036     {
037         tela = panelPontos.CreateGraphics();

038         for (int i = 1; i < Curva.retornarDimensao(); i++)
039         {
040             int x = i * panelPontos.Size.Width /
041             Curva.retornarDimensao();
042             int y = i * panelPontos.Size.Height /
043             Curva.retornarDimensao();
044             tela.DrawLine(caneta, x, 0, x,
045             panelPontos.Size.Height);
046             tela.DrawLine(caneta, 0, y,
047             panelPontos.Size.Width, y);
048         }

049         int k = 0;
050         foreach (Point point in Curva.retornarPontos())
051         {
052             tela.DrawString(++k.ToString(),
```

```
053     new Font("Arial",
054           160 / (2 * Curva.retornarDimensao())), pincel,
055           point.X * panelPontos.Size.Width / Curva.retornarDimensao(),
056           160 - (point.Y + 1) * panelPontos.Size.Height
057           / Curva.retornarDimensao());
058     }
059 }

060 private void buttonRetornar_Click
061     (object sender, EventArgs e)
062     {
063         if (Curva.retornarPontos().Count < 2)
064             MessageBox.Show("Escolha pelo menos dois pontos
065 do padrão para se definir uma reta.", "Atenção",
066             MessageBoxButtons.OK, MessageBoxIcon.None,
067             MessageBoxDefaultButton.Button1);
068         else
069             {
070                 Curva.alterarIteracao(Curva.retornarIteracao() - 1);
071                 panelCurva.Invalidate();
072             }
073     }

074 private void buttonAvancar_Click
075     (object sender, EventArgs e)
076     {
077         if (Curva.retornarPontos().Count < 2)
078             MessageBox.Show("Escolha pelo menos dois pontos
079 do padrão para se definir uma reta.", "Atenção",
080             MessageBoxButtons.OK, MessageBoxIcon.None,
081             MessageBoxDefaultButton.Button1);
082         else
083             {
084                 Curva.alterarIteracao(Curva.retornarIteracao() + 1);
085                 panelCurva.Invalidate();
086             }
087     }
```

```
087     }

088     private void panelCurva_Paint
089     (object sender, PaintEventArgs e)
090     {
091         tela = panelCurva.CreateGraphics();

092         double power = Math.Pow(Curva.retornarDimensao(),
093         Curva.retornarIteracao());
094         for (int i = 1; i < power; i++)
095         {
096             int x = i * panelCurva.Size.Width / (int) power;
097             int y = i * panelCurva.Size.Height / (int) power;
098             tela.DrawLine(caneta2, x, 0, x,
099             panelCurva.Size.Height);
100             tela.DrawLine(caneta2, 0, y,
101             panelCurva.Size.Width, y);
102         }

103         int ordem = 0;
104         double factorX = panelCurva.Size.Width / power,
105         factorY = panelCurva.Size.Height / power;
106         double[,] matriz = new double[(int)Math.Pow(
107         Curva.retornarPontos().Count,
108         Curva.retornarIteracao()), 2];
109         iterar(1, ref ordem, factorX, factorY,
110         new Point(0, 0), ref matriz);

111         if (matriz.GetLength(0) > 0)
112         {
113             double previousX = matriz[0, 0],
114             previousY = matriz[0, 1];
115             for (int i = 1; i < matriz.GetLength(0); i++)
116             {
117                 tela.DrawLine(caneta, (int) previousX, (int) previousY,
118                 (int) matriz[i, 0], (int) matriz[i, 1]);
```

```
119     previousX = matriz[i, 0];
120     previousY = matriz[i, 1];
121     }
122     }
123     }

124     private void iterar(int nivel, ref int ordem,
125     double factorX, double factorY, Point padrao,
126     ref double[,] matriz)
127     {
128     foreach (Point point in Curva.retornarPontos())
129     {
130     Point novoPadrao = new Point(
131     padrao.X * Curva.retornarDimensao() + point.X,
132     padrao.Y * Curva.retornarDimensao() + point.Y);

133     if (nivel == Curva.retornarIteracao())
134     {
135     matriz[ordem, 0] = novoPadrao.X * factorX
136     + factorX / 2;
137     matriz[ordem, 1] = panelCurva.Size.Height
138     - (novoPadrao.Y * factorY + factorY / 2);
139     }
140     else
141     {
142     iterar(nivel + 1, ref ordem, factorX,
143     factorY, novoPadrao, ref matriz);
144     continue;
145     }
146     ordem += 1;
147     }
148     }

149     private void buttonRedefinir_Click
150     (object sender, EventArgs e)
151     {
```

```
152     Curva.redefinir();
153     labelDimensao.Text = String.Format("0 x 1",
154     Curva.retornarDimensao(), Curva.retornarDimensao());
155     numericUpDownDimensao.Value = Curva.retornarDimensao();
156     panelCurva.Invalidate();
157     }
158 }
159 }
```

BIBLIOGRAFIA

- [1] AMORIM, É. *A curva de Peano e um demônio da Tasmânia cego puntiforme infinitamente rápido*. Disponível em: <http://legal.icmc.usp.br/lib/execute.php?media=slides:tasmania.pdf> (Acesso 10-2019)
- [2] BYRNE, K. *An Introduction to Space-Filling Curves*. Disponível em: http://www.academia.edu/28633281/An_Introduction_to_Space-Filling_Curves (Acesso 02-2019)
- [3] GUEDES, G. T. A. *UML 2: Uma Abordagem Prática*, 2. ed. São Paulo: Novatec, 2011. 30-41p.
- [4] KAPLANSKY, I. *Set Theory and Metric Spaces*, University of Chicago. Allyn and Bacon, Inc. Boston, 1972. 67-83p.
- [5] LIMA, E. L. *Análise Real, volume 1: Funções de Uma Variável*, 12. ed. Rio de Janeiro: IMPA, 2017.
- [6] LIMA, E. L. *Espaços Métricos*, 5. ed. Rio de Janeiro: IMPA, 2017. 1-80p.
- [7] MÜLLER, J. G. BAIER, T. *Topologia: Fatos Históricos e Curiosidades*, Universidade Regional de Blumenau – FURB/PPGECIM, Santa Catarina, s/d.
- [8] PEANO, G. *Sur une courbe, qui remplit toute une aire plane*. *Mathematische Annalen*, 36 N° 1, 1890. 157–160p.
- [9] SAADE, J. *C# Guia do Programador*, 1. ed. São Paulo: Novatec, 2011. 117-173p.
- [10] SAGAN, H. *Some Reflections on the Emergence of Space-filling Curves: The Way it Could Have Happened and Should Have Happened, but did not Happen*, *Journal of the Franklin Institute*, Vol. 328, N° 4, 1991. 419-430p.
- [11] SAGAN, H. *Space-Filling Curves*, Springer Science+Business Media. Nova Iorque, 1994. 1-82p.