Polinômio de Alexander via Linguagem Python

Alexander Polynomial via Python Language

Aldicio José Miranda
Universidade Federal de Uberlândia (UFU), Faculdade de Matemática, Uberlândia, MG, Brasil
https://orcid.org/0000-0002-6285-5210, aldicio@ufu.br

Taciana Oliveira Souza
Universidade Federal de Uberlândia (UFU), Faculdade de Matemática, Uberlândia, MG, Brasil
https://orcid.org/0000-0002-2495-8761, tacioli@ufu.br

Rui Marcos de Oliveira Barros
Universidade Estadual de Maringá (UEM), Departamento de Matemática, Maringá, PR, Brasil
https://orcid.org/0000-0001-9337-4770, rmobarros@uem.br

Informações do Artigo

Como citar este artigo

MIRANDA, Aldicio José; SOUZA, Taciana Oliveira; BARROS, Rui Marcos de Oliveira. Polinômio de Alexander via Linguagem Python. **REMAT: Revista Eletrônica da Matemática**, Bento Gonçalves, RS, v. 6, n. 1, p. 1-16, 30 jun. 2020. DOI: https://doi.org/10.35819/remat2020v6i1id3862

Histórico do Artigo
Submissão: 11 de janeiro de 2020.
Aceite: 27 de abril de 2020.

Palavras-chave
Topologia
Polinômio de Alexander
Linguagem Python

Resumo
Um nó clássico é um mergulho de uma esfera unidimensional S^1 em um ambiente tridimensional real, geralmente $\mathbb{R}^3$. Nestas condições é possível considerar o diagrama do nó, isto é, a projeção planar do mergulho. Esta assemelha-se a uma curva na qual os cruzamentos são trocados por interrupções no traço da mesma, indicando desta maneira que um arco passa por sobre o outro. Na Teoria dos Nós estudam-se invariantes algébricos extraídos do complementar do mergulho, e este complementar é visível no caso do mergulho $S^1 \to \mathbb{R}^3$. Um dos invariantes extraídos do diagrama do nó é o polinômio de Alexander. Neste artigo mostramos como o processo de determinar o polinômio de Alexander de um nó pode ser transportado para um algoritmo implementado no Python e obtido a partir de um diagrama desenhado com o auxílio do mouse.

Keywords
Topology
Alexander Polynomial
Python Language

Abstract
A classic knot is an embedding of an one-dimensional sphere S^1 in a real three-dimensional environment, usually $\mathbb{R}^3$. Under these conditions it is possible to consider the knot diagram, that is, the planar projection of the embedding. This resembles a curve in which the intersections are exchanged for interruptions in its trace, thus indicating that one arc passes over the other. In Knot Theory we study algebraic invariants extracted from the complement of the embedding, and this complement is visible in the case of embedding $S^1 \to \mathbb{R}^3$. One of the invariants extracted from the knot diagram is the Alexander Polynomial. In this article we show how the process of determining the Alexander Polynomial of a knot can be transported to an algorithm implemented in Python Language and obtained from a drawn diagram with the aid of a computer mouse.

1 Introdução

A Topologia é um dos mais importantes e antigos campos da Matemática. Essa área estuda propriedades de objetos geométricos que permanecem inalteradas quando são aplicados movimentos contínuos no próprio objeto ou no ambiente no qual ele está contido. Como ilustração do que falamos, podemos observar um toro sólido e uma xícara na Figura 1. Para a Topologia esses são objetos idênticos, pois um pode ser obtido a partir do outro por meio da aplicação de uma deformação contínua. Essa identificação entre o toro sólido e a xícara pode ser melhor compreendida mediante a observação de que cada um desses objetos possui um vazio, ou um buraco, e que essa característica não se altera sob a ação de deformações contínuas.

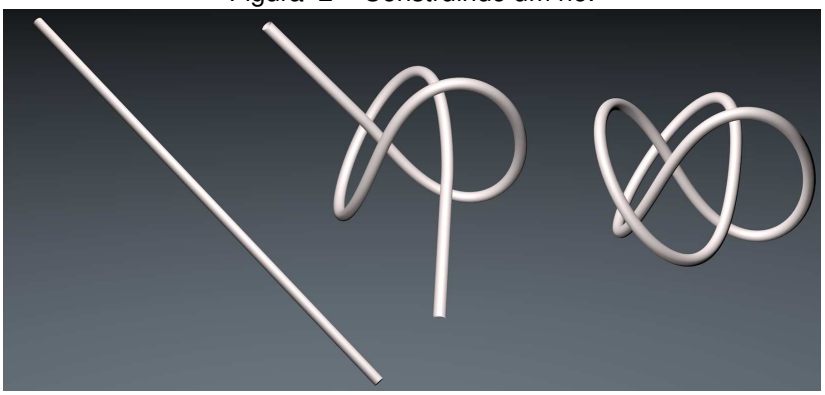
Figura 1 – Deformação de uma xícara em um toro sólido e vice-versa.

Fonte: Elaboração dos autores (2019).

Dentro da grande área da Topologia existe uma subárea chamada Teoria dos Nós. Essa teoria (a clássica) pode ser compreendida como o estudo das curvas fechadas, sem autointerseções, em três dimensões. De modo intuitivo, um nó é construído torcendo e entrelaçando um fio e unindo suas extremidades como mostra a Figura 2. O início do conceito matemático de "nó" aparece nos anos de 1830 com pesquisas de Carl Friedrich Gauss (1777-1855) (GAUSS, 1833). Seu interesse, na época, era aplicar esse conceito na área de eletrodinâmica. Nesse mesmo século, décadas depois, outro grande pesquisador, William Thomson (Lord Kelvin) (1824-1907), se interessou pelo assunto, pois acreditava que os "nós" seriam a chave para a compreensão das substâncias químicas (OSSERMAN, 2019) que, de acordo com suas crenças, seriam descritas pelas formas dos nós. Apesar de tal crença não condizer com a verdade, a Teoria dos Nós continuou a ser matematicamente

estudada e após os anos de 1960 aplicações foram sendo descobertas. Fato que merece destaque é o que ocorreu em 1990. Nesse ano, os pesquisadores Vaughan Jones e Edward Witten ganharam o maior prêmio de reconhecimento científico matemático, a Medalha Fields, pela descoberta da conexão entre a Teoria dos Nós e a Teoria do Campo Quântico (que reúne Mecânica Quântica e Teoria da Relatividade) (OSSERMAN, 2019).

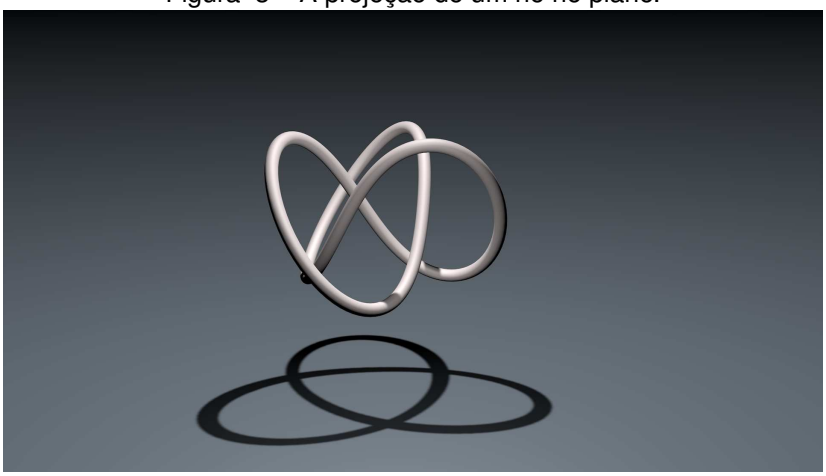
Figura 2 – Construindo um nó.



Fonte: Elaboração dos autores (2019).

Para estudarmos um nó, tendo em vista a dificuldade de desenharmos objetos tridimensionais, fixamos um plano do espaço e utilizamos sua projeção nesse plano. Sempre é possível considerar projeções que possuam apenas cruzamentos duplos. Veja a Figura 3. Nessa projeção, desenhamos pequenas interrupções próximas a cada cruzamento para indicarmos que um arco passa sobre o outro. Assim, consideramos o que chamamos de diagrama de um nó (Figura 3).

Figura 3 – A projeção de um nó no plano.

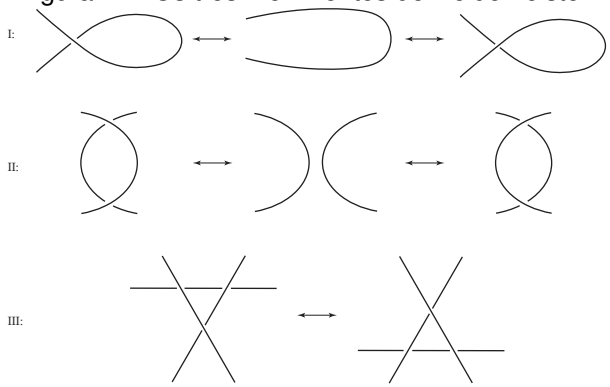


Fonte: Elaboração dos autores (2019).

Para a Teoria dos Nós, quando um nó pode ser transformado continuamente em outro que, a princípio, aparenta ser diferente, diz-se que os dois nós são equivalentes. Mas, se para registrarmos um nó utilizarmos um diagrama (uma projeção), é necessário investigar se dois diagramas que, a

princípio, são aparentemente diferentes, podem representar o mesmo nó. Esse problema foi resolvido por Kurt Reidemeister (1883-1971) em 1920. Ele propôs que se considerasse três movimentos que poderiam ser realizados em diagramas de nós (ROLFSEN, 1976). Hoje esses movimentos são chamados “movimentos de Reidemeister”. Eles estão ilustrados na Figura 4.

Figura 4 – Os três movimentos de Reidemeister.



Fonte: Elaboração dos autores (2019).

Quando um diagrama de um nó pode ser transformado em outro diagrama mediante aplicação de uma sequência finita de movimentos de Reidemeister, diz-se que esses diagramas são equivalentes. Com essa consideração, Reidemeister provou que “dois nós são equivalentes se, e somente se, os respectivos diagramas são equivalentes”.

Assim, pode-se estudar equivalência dos nós estudando equivalência dos diagramas dos nós. Apesar do excepcional resultado de Reidemeister, é geralmente difícil determinar quando um nó é equivalente a outro. Uma forma de transpor essa dificuldade é associar ao diagrama de um nó um objeto algébrico ou aritmético que permaneça o mesmo quando se efetue uma sequência finita de movimentos de Reidemeister. Assim, esse objeto pode auxiliar na detecção de nós que não são equivalentes. Chamamos tais objetos matemáticos de invariantes do nó.

A rotina em Python que apresentamos neste artigo, disponível em Miranda (2019), é uma rotina que calcula o invariante Polinômio de Alexander de um nó, ou seja, a rotina extrai o invariante de um diagrama desenhado pelo usuário.

Para tornar nossa linguagem matematicamente mais precisa, vamos esclarecer o que entendemos por deformação contínua ou por transformação contínua. Esses conceitos são explicados na próxima seção.

2 Espaços Topológicos, Homeomorfismos e Isotopias Ambientes

Tudo o que afirmamos na seção anterior são verdades demonstráveis, são Teoremas, mas não abordaremos as demonstrações desses fatos neste texto. Para prosseguirmos o artigo e apresentarmos nossa rotina temos de melhorar um pouco mais nossa linguagem.

Para a Topologia, os objetos básicos são os espaços topológicos.

Definição 2.1. *Um espaço topológico é um conjunto X munido de uma coleção de partes de X, denotada τ, que são os chamados subconjuntos abertos de X, de maneira que sejam respeitadas as três seguintes propriedades:*

i) O conjunto vazio e o conjunto X são conjuntos abertos, ou seja, $\emptyset \in \tau$ e $X \in \tau$.

ii) A interseção de dois conjuntos abertos é um conjunto aberto, ou seja, $A_1 \in \tau$ e $A_2 \in \tau \Rightarrow A_1 \cap A_2 \in \tau$.

iii) A união de uma família qualquer de conjuntos abertos é um conjunto aberto, ou seja, $(A_\gamma)_{\gamma\in\Gamma}$, $A_\gamma \in \tau \Rightarrow \bigcup_{\gamma\in\Gamma} A_\gamma \in \tau$.

Essa definição de espaço topológico generaliza a noção do espaço unidimensional $\mathbb{R}$ com sua coleção de intervalos abertos, generaliza a noção do espaço bidimensional $\mathbb{R}^2$ com sua coleção de regiões abertas, etc., e possibilita trabalhar com aplicações contínuas nos mais diversos espaços topológicos.

Para exemplificar, uma reta, uma circunferência, um disco (circunferência e seu interior), o próprio espaço $\mathbb{R}^3$, um plano, seções de um plano, uma região poligonal, curvas no plano, curvas no espaço, etc., são exemplos de espaços topológicos. A noção que temos de espaço topológico é muito mais geral, mas para o nosso interesse, esses exemplos já são boas ilustrações.

Anteriormente, falamos em deformações contínuas; a definição dessa noção intuitiva é a seguinte:

Definição 2.2. *Um homeomorfismo $h : A \to B$ entre dois espaços topológicos é uma aplicação contínua bijetora com aplicação inversa também contínua.*

Nessa situação, h é uma deformação que transforma A em B continuamente e que pode ser desfeita, pois existe $h^{-1} : B \to A$ que transforma B em A, $h^{-1}(B) = h^{-1}(h(A)) = A$.

Resgatando nosso exemplo da xícara e do toro sólido, diremos que a xícara é homeomorfa ao toro sólido. É o conceito de homeomorfismo que nos permite dizer que a topologia é, em linguagem figurada, a "geometria da borracha". Por exemplo, para a Topologia, o quadrado $Q = [-1, 1] \times [-1, 1]$ é homeomorfo ao disco $D^2 = \{(x, y) \in \mathbb{R}^2; x^2 + y^2 \leq 1\}$ de centro $(0, 0)$ e raio 1. Existe um homeomorfismo entre o quadrado Q e o disco D^2. O quadrado pode ser deformado continuamente no disco e tal deformação pode ser revertida. Para a Topologia, Q e D^2 são indistintos, são homeomorfos.

Como estamos interessados na Teoria Clássica dos Nós, devemos esclarecer o que chamamos de nó clássico.

Primeiramente, qualquer espaço homeomorfo à circunferência $\{(x, y) \in \mathbb{R}^2; x^2 + y^2 = 1\}$ será chamado de esfera unidimensional, esfera de dimensão 1, e será denotado por S^1. Por exemplo, o bordo (ou fronteira) do quadrado Q descrito anteriormente é tratado como uma esfera S^1, pois é homeomorfo a essa esfera unidimensional.

Generalizamos esses conceitos. A título de esclarecimento, qualquer espaço homeomorfo à esfera bidimensional $\{(x, y, z) \in \mathbb{R}^3; x^2 + y^2 + z^2 = 1\}$ será denotado por S^2. Assim, se consideram esferas tridimensionais S^3, quadridimensionais S^4 e assim por diante.

O que chamamos de nó (clássico) em $\mathbb{R}^3$ é a imagem de uma aplicação contínua injetiva $f : S^1 \to \mathbb{R}^3$, (chamamos isso de mergulho). Por padrão, identificamos um nó com a imagem de uma aplicação contínua injetiva e o denotamos $K = f(S^1)$. Assim, quando consideramos, por exemplo, dois nós, estamos considerando dois mergulhos $f(S^1) \subset \mathbb{R}^3$ e $g(S^1) \subset \mathbb{R}^3$. Então, a diferença entre os diversos nós não está no espaço topológico que é a imagem do mergulho, está no espaço que circunda a imagem do mergulho, está no complementar $\mathbb{R}^3 \setminus f(S^1)$. Nossa noção intuitiva de deformação de um nó K_1 em outro nó K_2 é, na verdade, a noção de um movimento contínuo no espaço ambiente $\mathbb{R}^3$ que leva continuamente o subespaço topológico K_1 sobre o subespaço topológico K_2. Essa noção é a associada à definição de isotopia ambiente.

Definição 2.3. *Considere um espaço topológico X. Uma isotopia ambiente H em X é uma aplicação contínua $H : X \times [0, 1] \to X$ tal que $H_0 : X \times \{0\} \to X$ é a aplicação identidade e $H_t : X \times \{t\} \to X$ seja homeomorfismo para todo $t \in [0, 1]$.*

A isotopia ambiente em um espaço topológico X é, então, uma coleção de funções bijetoras contínuas $H_t : X \to X$, $H_t(x) = H(x, t)$, $\forall x \in X$.

Se considerarmos que $t \in [0, 1]$ seja a medida do tempo, comprendemos uma isotopia em X como uma coleção de funções contínuas e bijetoras H_t, tal que no instante 0 tem-se a função iden-

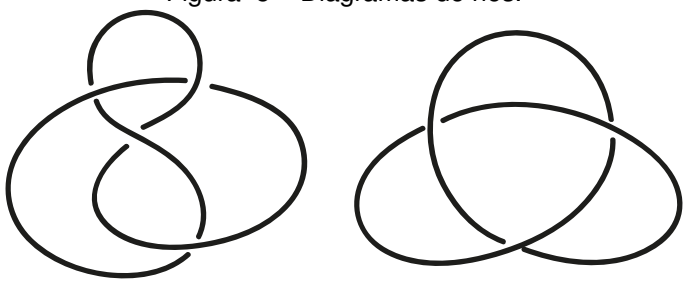
tidade e que para cada instante de tempo t tem-se uma deformação que transforma bijetivamente o espaço X.

3 A Teoria dos Nós

Podemos, agora, dizer com maior precisão matemática o que é a Teoria dos Nós em $\mathbb{R}^3$. Essa área de pesquisa consiste no estudo das várias maneiras possíveis de mergulhar uma esfera unidimensional S^1, representada por uma curva fechada, no espaço euclidiano tridimensional $\mathbb{R}^3$.

Considere a Figura 5, na qual ilustramos dois diagramas de nós K_1 e K_2 mergulhados em $\mathbb{R}^3$. Observamos que, apesar das imagens dos dois megulhos serem homeomorfas, não conseguimos uma sequência finita de movimentos de Reidemeister que posicione o nó K_1 sobre o nó K_2. A impossibilidade disso se deve ao fato de que os complementares desses nós serem intrinsecamente diferentes. Cada um dos movimentos de Reidemeister é realizável devido à existência de uma isotopia ambiente que o possibilita. Assim, uma sequência finita de movimentos de Reidemeister está associada à existência de uma sequência finita de isotopias ambiente que permitam posicionar um nó sobre o outro.

Figura 5 – Diagramas de nós.



Fonte: Elaboração dos autores (2019).

Definição 3.1. *Dizemos que dois nós K_1 e K_2 têm o mesmo tipo se existe uma isotopia ambiente $H : \mathbb{R}^3 \times [0, 1] \to \mathbb{R}^3$ tal que $H_1(K_1) = K_2$.*

Mas a observação visual de dois diagramas de nós pode ser enganosa. Às vezes, encontramos dificuldades para determinar visualmente que não existem isotopias ambientes que fornecem $H_1(K_1) = K_2$. Para isso, foram desenvolvidos vários objetos aritméticos e algébricos extraídos do complementar do nó que nos fornecem respostas acerca da não existência de tais isotopias. Esses objetos são chamados invariantes por isotopia do nó, ou apenas “invariantes do nó”.

Um dos famosos invariantes do nó é o chamado Polinômio de Alexander. Esse invariante foi determinado por James Wadwell Alexander (1888-1971) no ano de 1928. Vejamos como calcular esse invariante.

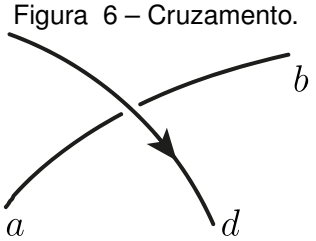
4 O Polinômio de Alexander

Para determinar o Polinômio de Alexander de um nó K, seguem-se os seguintes passos:

1. Escolha um diagrama para K e uma orientação para o diagrama.

2. Associe a cada arco uma variável e a cada cruzamento uma equação seguindo o esquema da Figura 6.

Figura 6 – Cruzamento.



Fonte: Elaboração dos autores (2019).

Equação: $b - ta - (1-t)d = 0.$

Para a ilustração da Figura 6, consideramos a variável d a que é associada ao arco que passa superiormente na região do cruzamento. Para a escolha de a e b na equação, procedemos da seguinte maneira: usamos a orientação do trecho superior do nó, no cruzamento; a deve ser identificado com a variável associada à direita de d e, consequentemente, b deve ser identificado com a variável à esquerda de d.

3. Coloque uma variável qualquer igual a zero.

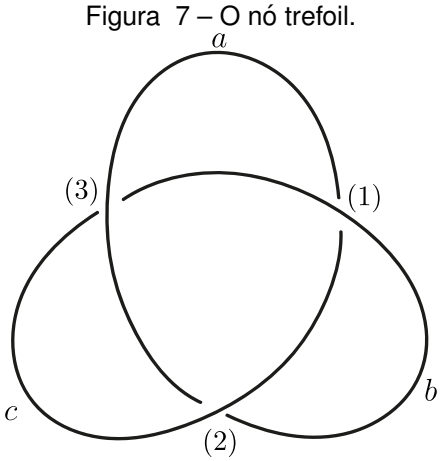
4. Descarte uma equação qualquer.

5. Escreva o sistema de equações acima com coeficientes no anel $\mathbb{Z}[t, t^{-1}]$.

6. Calcule o determinante $\delta(t)$ deste sistema, que será um elemento de $\mathbb{Z}[t, t^{-1}]$.

7. Multiplique $\delta(t)$ por $\pm t^j$ apropriado para obter $\Delta(t)$, tal que $\Delta(t) = \Delta(t^{-1})$ e $\Delta(1) = 1$.

Figura 7 – O nó trefoil.



Fonte: Elaboração dos autores (2019).

A Figura 7 exemplifica o método com o diagrama do nó trefoil.

Percorremos o trefoil na orientação “horária”, ou seja, em uma orientação que percorre os cruzamentos (3), (1) e (2), nessa ordem. Os cruzamentos (1), (2) e (3) fornecem respectivamente as três seguintes equações:

$$\begin{cases} a - tc - (1-t)b &= 0 \quad (1) \\ b - ta - (1-t)c &= 0 \quad (2) \\ c - tb - (1-t)a &= 0 \quad (3) \end{cases}$$

Faça $a = 0$ e despreze a primeira equação. Assim, obtemos o sistema

$$\begin{cases} b - (1-t)c &= 0 \\ c - tb &= 0 \end{cases}$$

cujo determinante é

$$\delta(t) = \begin{vmatrix} 1 & -(1-t) \\ -t & 1 \end{vmatrix} = 1 - t + t^2.$$

Dessa forma, o Polinômio de Alexander do nó é $\Delta(t) = t^{-1}\delta = t - 1 - t^{-1}$.

Esse invariante permite responder negativamente à questão da existência de isotopia ambiente. Se, por exemplo, para um diagrama de um nó K_1 obtivermos $\Delta_1(t) = t^2 + t - 1 - t^{-1} + t^{-2}$, podemos afirmar que o nó K_1 não é do mesmo tipo que o trefoil, já que seus respectivos polinômios

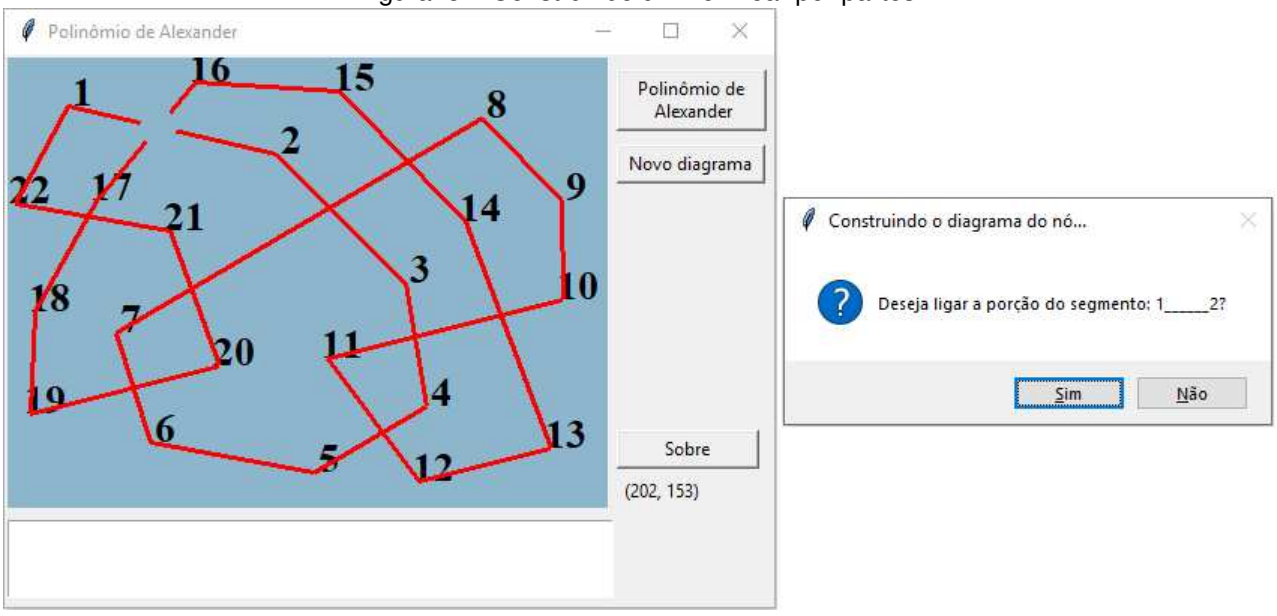
de Alexander são diferentes. Ou seja, o trefoil não poderia ser deformado continuamente até obtermos o nó K_1.

O polinômio assim obtido é na verdade chamado de normalização de Conway[1] do polinômio de Alexander. Para mais detalhes veja Kawauchi (1990, p. 105).

5 A Interface Gráfica

Para facilitar o desenho de diagramas e o cálculo dos polinômios de Alexander, adotaremos que a esfera unidimensional S^1 pode ser considerada uma sucessão finita de segmentos de reta. Assim, um mergulho de S^1 em $\mathbb{R}^3$ pode ser considerado mediante o desenho de um diagrama constituído por segmentos de reta. Vamos, para efeito de simplificação, considerar que as interseções de dois possíveis segmentos se dá em um ponto interno a cada segmento, e nunca em vértices. Na área de Topologia, diz-se que iremos trabalhar na categoria linear por partes (picewise-linear). Com essas considerações, pode-se obter um diagrama do nó, conforme a Figura 8.

Figura 8 – Construindo um nó linear por partes.



Fonte: Elaboração dos autores (2019).

Na Figura 8, o diagrama é composto por 22 segmentos de reta. Não precisaria ser sempre assim, pois a quantidade de segmentos que compõem um diagrama de um nó não influencia o

[1] Prestamos nossa homenagem ao ilustre matemático John H. Conway (1923-2020) que trabalhou em Cambridge nos anos 60 e 70 e estava desde 1987 na Universidade de Princeton, falecido dia 11 de abril de 2020 devido a complicações advindas do Covid-19. Com novos métodos combinatórios introduzidos por Conway, que combinados com mais recentes trabalhos de V. Jones, a Teoria dos Nós tem progredido bastante com outros novos invariantes.

cálculo do Polinômio de Alexander de tal nó. Para todos os efeitos, existe uma isotopia ambiente que suaviza os vértices de um diagrama (ou mergulho) de qualquer nó. A consideração desse tipo de mergulho facilita a construção do algoritmo para o desenho e o cálculo do invariante que desejamos.

O usuário dispõe de uma área de desenho de 900 $\times$ 700 pixels, na qual vai desenhar o diagrama do nó clicando no botão esquerdo do mouse para criar os vértices dos segmentos de reta que compõem o diagrama. À medida que desloca o mouse sobre a área de desenho, vai clicando no botão esquerdo e estocando as posições $A_i(x_i, y_i)$ e $A_{i+1}(x_{i+1}, y_{i+1})$.

Após o segundo ponto armazenado, determina-se a equação da reta que passa por A_i e A_{i+1} e, mediante a consideração da parametrização

$$A_i + t_i(A_{i+1} - A_i), \;\; t_i \in (0, 1),$$

desenha-se o segmento $\overline{A_i A_{i+1}}$.

Quando o usuário desejar fechar o diagrama, ligando o último ponto determinado ao primeiro ponto, ele deve clicar sobre o botão “Polinômio de Alexander”. O software traçará o segmento necessário. Um exemplo dessa primeira etapa está na Figura 8.

Ao final do processo teremos uma coleção de segmentos indexados segundo a ordem executada pelo usuário, $I_1, I_2, \ldots, I_n$, onde I_1 indica o segmento ligando 1 a 2, I_2 o segmento ligando 2 a 3 e assim por diante.

As interseções entre os segmentos são obtidas resolvendo-se o seguinte sistema:

$$\begin{aligned}
&I_1 \cap I_3,\ I_1 \cap I_4,\ I_1 \cap I_5, I_1 \cap I_6, \ldots, I_1 \cap I_n \\
&I_2 \cap I_4,\ I_2 \cap I_5, I_2 \cap I_6, \ldots, I_2 \cap I_n \\
&\vdots \\
&I_{n-3} \cap I_{n-1},\ I_{n-3} \cap I_n \\
&I_{n-2} \cap I_n
\end{aligned}$$

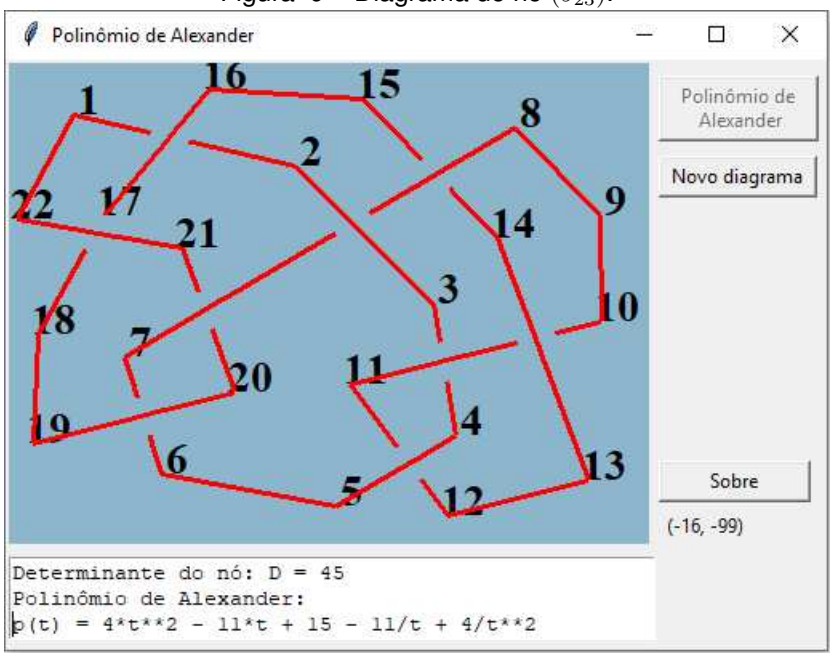
A título de explicação, fixamos o primeiro segmento e procuramos as interseções deste com os demais segmentos; não é necessário procurar interseção com o subsequente, pois tal interseção será um dos vértices. Depois, fixamos o segundo segmento e procuramos as interseções dele com os demais e assim sucessivamente. Não é necessário procurar interseção com os segmentos anteriores, já que essas possíveis interseções já teriam sido encontradas e memorizadas. Devido à

simetria existente, a cada vez que avançamos para fixar um segmento, a quantidade de possíveis pontos de interseção diminui.

Como consideramos segmentos de reta, ao determinarmos as interseções da reta R_i que passa por A_i e A_{i+1} e a reta R_j que passa pelos pontos A_j e A_{j+1}, devemos restringir os parâmetros t_i e t_j, respectivamente de R_i e R_j, ao intervalo aberto $(0, 1)$, pois não desejamos considerar pontos de interseção que não façam parte do diagrama desenhado.

Terminada essa etapa, os pontos de interseção são armazenados na memória. Na próxima etapa, o programa apaga uma pequena área circular em torno de cada um dos pontos de interseção e apresenta uma caixa de diálogo para o usuário (Figura 8). Nessa caixa de diálogo, o usuário é indagado e deve escolher qual o segmento passa por cima em cada uma das interseções. Isso será feito até que todos os cruzamentos tenham sido considerados. Ao final dessa etapa, o diagrama fica semelhante ao ilustrado na Figura 9.

Figura 9 – Diagrama do nó (9_{23}).



Fonte: Elaboração dos autores (2019).

Cada escolha fornecida pelo usuário sobre qual segmento passa por cima em cada cruzamento é armazenada em um vetor com a estrutura

$$V_i = (i, j, i \text{ ou } j).$$

As duas primeiras coordenadas desse vetor indicam os índices dos segmentos que se interceptam; a terceira coordenada será igual a i se o segmento I_i passar por cima do segmento I_j, ou

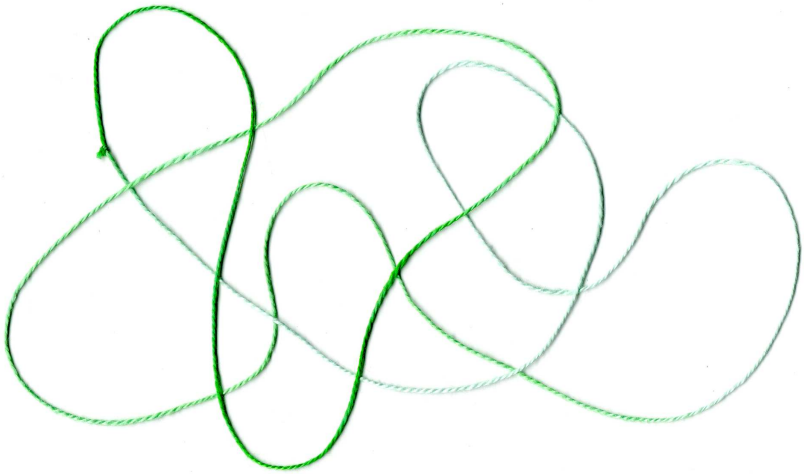
será igual a j, se o segmento I_j passar por cima do segmento I_i. Por exemplo, observando a Figura 9, temos um vetor $V_6 = (6, 19, 19)$ que indica que o segmento I_6 intercepta o segmento I_{19}, e que nesse cruzamento o segmento I_{19} passa por cima.

São as informações dos vetores V_i's que permitem determinar o sistema de equações e realizar os procedimentos descritos na Seção 4 para o cálculo do Polinômio de Alexander. Tal polinômio é exibido logo abaixo do diagrama do nó. Ademais, o programa também exibe o determinante do nó, um outro invariante dado por $\Delta(-1) = p(-1)$.

6 Exemplo – Nó Conway

Nesta seção apresentamos exemplos de nós construídos com barbante (Figuras 10 e 11). O nó da Figura 11, com 11 cruzamentos, foi apresentado pela primeira vez pelo matemático John H. Conway (1970). Mais informações podem ser encontradas em Rolfsen (1976, p. 173).

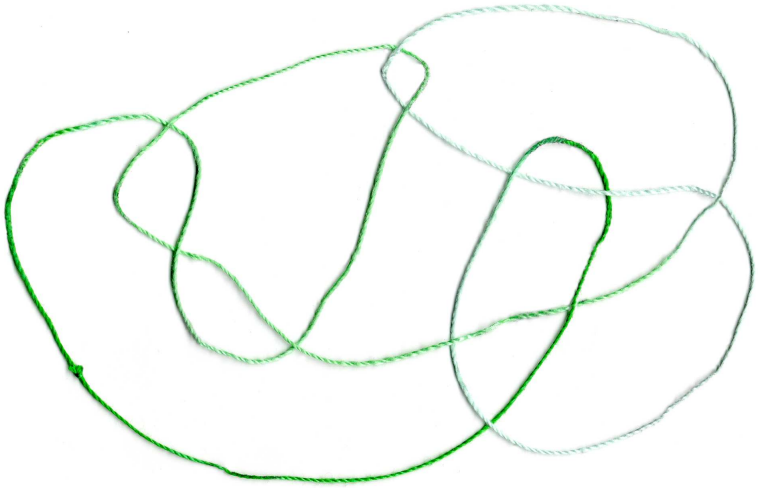
Figura 10 – Nó trivial.



Fonte: Elaboração dos autores (2019).

Convidamos o leitor a construir ambos os nós fisicamente a partir de pedaços de barbante e, em seguida, usar os movimentos de Reidemeister para concluir que o nó da Figura 10 é trivial. É conhecido que o nó obtido por Conway não é trivial. Após isso, usar os passos da seção 4, ou o algoritmo via Python disponível em Miranda (2019), para concluir que o polinômio de Alexander e o determinante do nó, nestes dois casos, são triviais. Este exemplo mostra que esses dois invariantes não são completos, isto é, $p(t) = 1$ não garante que o nó seja trivial.

Figura 11 – Nó Conway.



Fonte: Elaboração dos autores (2019).

7 Procedimentos Metodológicos

Para compreensão da construção e efetividade do invariante Polinômio de Alexander e fundamentação teórica do artigo foram necessários estudos de Topologia Geral e Teoria dos Nós, realizados em Rolfsen (1976), Hacon (1985), Murasugi (1996) e Lima (2014).

Concomitantemente com os estudos matemáticos, iniciamos a escolha da linguagem para construção do software. Adotamos o Python (PSF, 2019b), por ser uma linguagem de programação livre, de fácil aprendizagem e por conter uma robusta biblioteca para computação científica.

8 Resultados e Discussão

Neste trabalho apresentamos um dos resultados de uma pesquisa que usa ferramentas computacionais para a resolução de problemas da Matemática Pura. Mais precisamante, relatamos aqui o desenvolvimento de um software que calcula o Polinômio de Alexander, um dos invariantes mais importantes da Teoria dos Nós.

A linguagem de programação utilizada no desenvolvimento do software foi o Python, pois fornece diversas ferramentas para muitos tipos de tarefas. Por exemplo, sua biblioteca padrão conta com utilidades para escrever aplicações para a internet e módulos para criar interfaces gráficas. Destacamos ainda o fato da linguagem Python possuir interpretador disponível para diversas plataformas como, por exemplo, Windows, MacOS X, Linux, FreeBSD e Solaris.

9 Considerações Finais

Verificamos a eficiência do cálculo do Polinômio de Alexander pelo software que desenvolvemos utilizando a linguagem de programação Python. Ressaltamos a importância do conhecimento de linguagens de programação para que se possa implementar conceitos matemáticos que são descritos por meio de algoritmos, como foi o caso apresentado neste artigo. Pretendemos, com isso, incentivar o uso de ferramentas computacionais por parte dos estudantes de Matemática, facilitando a verificação de cálculos efetuados durante estudos de graduação e de pós-graduação.

As habilidades em Python são vastas e podem ser aplicadas em diversas áreas como, por exemplo, desenvolvimentos para internet, interfaces para desktop, computação numérica e científica, na área de educação, programação em redes, desenvolvimento de software e jogos. Mais detalhes podem ser encontrados em PSF (2019a). No algoritmo desenvolvido para calcular o polinômio de Alexander, fizemos uso da biblioteca *SymPy* (SYMPY, 2020), que é uma biblioteca Python e trabalha com a Matemática simbólica, sendo um sistema de álgebra computacional.

Referências

CONWAY, J. H. An enumeration of knots and links, and some of their algebraic properties. **Computational Problems in Abstract Algebra**. New York: Pergamon Press, p. 329-358, 1970.

CROWELL, R. H.; FOX, R. H. **Introduction to Knot Theory**. Graduate Texts in Mathematics, n. 57. New York: Springer Verlag, 1963.

FLAPAN, E. **When topology meets chemistry**: A topological look at molecular chirality. Cambridge: Cambridge University Press, 2000.

GAUSS, K. F. **Zur mathematischen theorie der eletrodynamischen wirkungen**. Werke Konigl. Gessell. Wiss. Gottingen, 1833.

HACON, D. **Introdução à teoria dos nós**. Rio de Janeiro: IMPA, 1985.

KAWAUCHI, A. **A Survey of Knot Theory**. Tokio: Springer-Verlag, 1990.

LIMA, E. L. **Elementos de Topologia Geral**. 3. ed. Textos Universitários. Rio de Janeiro: SBM, 2014.

MIRANDA, Aldicio José. **Polinômio de Alexander via Python**. Disponível em: `https://sites.google.com/site/aldicio/publicacoes/PolinomioAlexanderViaPython`. Acesso em: 27 nov. 2019.

MURASUGI, K. **Knot theory and its applications**. Birkhauser Boston, 1996.

OSSERMAN, Robert. Knot theory. **Encyclopaedia Britannica**. Disponível em: https://www.britannica.com/science/knot-theory. Acesso em: 7 nov. 2019.

PSF. Python Software Foundation. **Applications for Python**. Disponível em: https://www.python.org/about/apps/#software-development/. Acesso em: 17 abr. 2020.

PSF. **Python Software Foundation**. Disponível em: https://www.python.org/. Acesso em: 25 nov. 2019.

ROLFSEN, Dale. **Knots and Links**. Berkeley, CA: Publish or Perish, 1976.

SYMPY. Disponível em: https://www.sympy.org/en/index.html. Acesso em: 17 abr. 2020.