

MÉTODOS (ALGÉBRICOS) TOPOLÓGICOS EM COMPUTAÇÃO

JAIR DONADELLI

1. INTRODUÇÃO

Em Teoria da Computação buscamos compreender a dificuldade de se resolver problemas computacionais e uma questão importante nessa busca é a de determinar cotas inferiores para a complexidade dos algoritmos que resolvem um dado problema, uma tarefa reconhecidamente difícil nessa área de estudo.

Cotas inferiores para a complexidade de problemas computacionais são resultados negativos no sentido de que enunciam que o problema dado não pode ser resolvido usando menos recurso computacional que o de determinado pela cota inferior. Esse tipo de resultado, ou a prova da dificuldade da solução de problemas, é um foco central da disciplina e atualmente muitos problemas práticos são baseados nesses resultados ou na hipótese deles serem verdadeiros, pois muito esforço tem sido feito e poucos resultados com relação a problemas relevantes foram obtidos. Notadamente, a Criptografia Moderna é fundamentalmente baseada na hipótese de existirem problemas difíceis. Num outro sentido, cotas inferiores proporcionam prova de otimalidade para algoritmos. Embora conheçamos uma grande variedade de problemas resolvidos por algoritmos eficientes, pouco se sabe sobre algoritmos que são ótimos com respeito a alguma medida de complexidade.

O estudo da demanda dos recursos computacionais envolvidos na solução de problemas computacionais, ou uma prova de que não há solução, só é possível porque *modelos abstratos de computação* fazem o conceito de computação matematicamente preciso. Tais modelos formais capturam os aspectos práticos da computação e nos ajudam na compreensão das dificuldades inerentes de se resolver problemas computacionais.

Dados um modelo abstrato e um problema, determinar uma cota inferior para a complexidade do problema é saber qual é o mínimo de recursos que *qualquer* dispositivo computacional (algoritmo) nesse modelo precisa para resolver o problema. Por exemplo, se soubermos provar que algum problema NP-completo não pode ser resolvido por uma Máquina de Turing de tempo polinomial, ou seja, o problema tem cota inferior superpolinomial nesse modelo, então $P \neq NP$; ou ainda, se soubermos provar que algum problema NP-completo não pode ser computado por circuitos booleanos (um modelo de computação) de tamanho polinomial (uma medida de complexidade) então $P \neq NP$ entretanto, atualmente, não se sabe mostrar um cota inferior superpolinomial para Máquina de Turing ou não

linear para o tamanho de circuitos. Notemos que encontrar a prova de uma cota inferior é equivalente a dar uma propriedade da classe de todos os algoritmos que resolvem o problema.

Não entraremos em detalhes sobre as classes P e NP e o problema $P \times NP$, tampouco entraremos em detalhes sobre máquinas de Turing, circuitos booleanos e como eles se relacionam, os interessados podem consultar sobre esses temas em [?, ?, ?]. A palavra *algoritmo* será usada no sentido que maioria de nós estamos acostumados, como aparecem nos livros texto sobre algoritmos (veja um exemplo no Algoritmo ?? abaixo) o que significa, particularmente, que é um dispositivo de computação equivalente à máquina de Turing,

Antes de iniciarmos a discussão sobre a iteração entre topologia e computação, ilustraremos alguns conceitos ditos nos parágrafos anteriores com um resultado clássico de cota inferior. Ademais, aproveitaremos esse exemplo para introduzirmos alguma nomenclatura que será útil mais adiante. Usamos a notação

$$[n] = \{1, 2, \dots, n\}$$

e denotamos por S_n o conjunto das permutações sobre $[n]$.

Exemplo 1 (Cota inferior para ordenação). O problema de *ordenação baseada em comparações* pode ser formulado da seguinte maneira: dada uma seqüência de números inteiros a_1, a_2, \dots, a_n , que chamamos de *entrada*, determinar uma permutação $\pi \in S_n$ tal que $a_{\pi(1)} \leq a_{\pi(2)} \leq \dots \leq a_{\pi(n)}$.

Uma solução é dada pelo algoritmo *ordenação por inserção*, descrito a seguir.

Dado seqüência de inteiros a_1, \dots, a_n ;
Devolve $\pi \in S_n$ tal que $a_{\pi(1)} \leq \dots \leq a_{\pi(n)}$.
Para i de 0 até n *faça*
 $\pi(i) \leftarrow i$; *Para* i de 2 até n *faça*
 $v \leftarrow \pi(i)$;
 $j \leftarrow i$;
 Enquanto $a_{\pi(j-1)} > a_v$ e $j > 1$ *faça*
 $\pi(j) \leftarrow \pi(j-1)$;
 $j \leftarrow j-1$;
 $\pi(j) \leftarrow v$;
Devolva π .

Na discussão a seguir vamos nos ater em soluções obtidas quando a única informação a respeito da ordem relativa entre esses elementos são as que podemos dispor a partir das respostas para “ $a_i < a_j$?”. Também, supomos que os elementos da entrada são distintos para simplificar a discussão.

O comportamento desse algoritmo, bem como de qualquer outro que resolva o problema, com qualquer entrada de tamanho n pode ser representado por uma árvore binária. Uma árvore binária T é ou a árvore vazia, denotada \emptyset por abuso

de notação e que não possui vértices, ou é uma terna $T = (r, E, D)$ em que r é um vértice chamado raiz de T , e E e D são árvores binárias disjuntas e que não contêm r , chamadas subárvore esquerda e subárvore direita, respectivamente. A raiz de E , dado que $E \neq \emptyset$, é chamada *filho esquerdo* de r e a raiz de D , dado que $D \neq \emptyset$, é chamada *filho direito* de r . Nos casos $(f, \emptyset, \emptyset)$ chamamos f de *folha*, e os vértices não-folha são ditos *vértices internos*.

Nas árvores binárias que representam o comportamento dos algoritmos que resolvem o problema de ordenação de n itens, os vértices internos são rotulados com uma comparação entre dois elementos da entrada e cada folha é rotulada com uma permutação de S_n . A execução do algoritmo corresponde a um percurso nessa árvore. Esse percurso começa na raiz da árvore onde a comparação correspondente a aquele vértice é feita, se a resposta ao teste é **sim**, então o percurso continua na subárvore esquerda, senão na subárvore direita daquele vértice. A folha alcançada no final do percurso é a permutação desejada.

Uma árvore para o algoritmo ?? e $n = 3$ e dada na figura 1 abaixo.

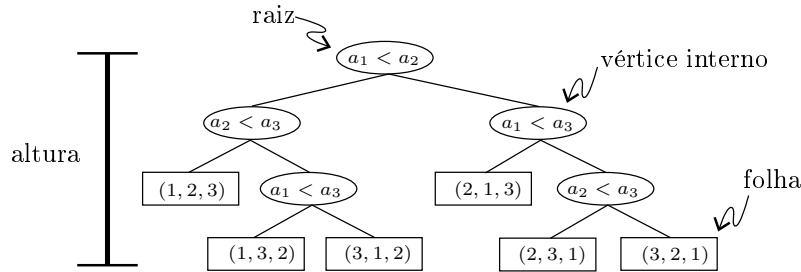


FIGURE 1. Árvore de comparações de altura 3 correspondente ao algoritmo ?? para $n = 3$. Da árvore podemos deduzir que o algoritmo ?? ordena qualquer entrada com três elementos fazendo no máximo três comparações.

A *altura* da árvore binária T é o maior número de arestas entre a raiz de T e uma folha de T .

Genericamente, todo algoritmo de ordenação baseado em comparações entre os elementos da entrada define uma seqüência de árvores de comparações T_n , em que $n \geq 1$ é o número de itens a serem ordenados. A *altura* da árvore T_n é um limitante superior para o número de comparações que o algoritmo efetua no *pior caso* nas instâncias de tamanho n . Dessa forma, se conseguimos mostrar um limitante inferior $i(n)$ para a altura de qualquer seqüência T_n dessas árvores que representam o comportamento dos algoritmos de ordenação baseados em comparações, então podemos concluir que *nenhum algoritmo de ordenação baseado em comparações entre os elementos da entrada resolve o problema da ordenação realizando menos que $i(n)$ comparações*.

O número máximo de folhas em uma árvore de altura h é 2^h e em T_n há $n!$ folhas pois cada permutação deve ocorrer em alguma folha, ou seja, toda árvore tem

altura pelo menos

$$\log_2(n!) = \sum_{k=1}^n \log_2 k \geq \sum_{k \geq n/2}^n \log_2 k \geq \frac{n}{2} \log \frac{n}{2}.$$

Isso significa que *qualquer* algoritmo baseado em comparações para ordenar n itens necessariamente realiza da ordem de $n \log n$ comparações.

Para finalizar esse exemplo, observamos que o algoritmo ?? realiza um número linear em n de comparações quando a entrada está ordenada e um número quadrático quando a entrada está em ordem decrescente, e essa é a característica da entrada que causa o pior desempenho desse algoritmo. Um algoritmo de ordenação proposto por von Neumann em 1945 e atualmente conhecido como ordenação por intercalação (*mergesort*) ordena n inteiros usando $n \lceil \log_2 n \rceil + 2n - 2^{\lceil \log_2 n \rceil + 1}$ comparações, ou seja, da ordem de $n \log n$ comparações no pior caso, é um algoritmo ótimo para o problema [?].

Neste texto usaremos um modelo algébrico de computação representado por uma árvore binária. Nos modelos algébricos as operações aritméticas $+$, $-$, \times , \div , $\sqrt{\quad}$ em \mathbb{R} são operações básicas e isso faz desses modelos, ditos não-realísticos por alguns autores, dispositivos de computação muito mais poderosos que os modelos tradicionais, e.g. máquina de Turing, que operam sobre inteiros. Sabemos, por exemplo, que se se permite aritmética com precisão arbitrária então é possível fatorar um inteiro em tempo linear [?]. Sabemos também que existem problemas NP-difíceis que admitem algoritmos algébricos de complexidade polinomial [?].

Ao que se refere a cotas inferiores, apesar do que dissemos no parágrafo acima, as cotas obtidas nos modelos mais poderosos aplicam-se também aos modelo mais fracos (e realísticos) de computação, com a vantagem de termos a possibilidade de utilizarmos resultados de outras disciplinas tradicionais da matemática, como a Topologia e Geometria Algébricas, para tentarmos derivar cotas inferiores.

A idéia central da técnica que apresentaremos a seguir é relacionar a complexidade computacional de decidir pertinência num conjunto $S \subset \mathbb{R}^n$ à complexidade topológica de S e, a partir disso, determinar cotas inferiores para a complexidade de problemas computacionais. As primeiras estimativas de cotas inferiores para a altura de árvores que estabelecem ligações entre a complexidade computacional para decidir pertinência em S e invariantes topológicos de S dependem da contagem do número de componentes conexas de S e do complemento

$$\bar{S} = \mathbb{R}^n \setminus S.$$

Esse método foi introduzido por Dobkin e Lipton [9] em 1979 para árvores com testes lineares nos vértices, depois foi estendido para árvores com testes polinomiais por Steele e Yao [14] em 1982 e logo em seguida, em 1983, Ben-Or [2] modificou e estendeu esses resultados para árvores de computação, nas quais é permitido operações aritméticas nos vértices. Daí seguiram-se vários resultados

[3, 4, ?, ?, ?, ?, 15, 16] que estabelecem conexões entre complexidade de algoritmos e invariantes topológicos. A seguir, apresentamos alguns desses resultados.

Por simplicidade, no que segue, usaremos *componente conexa* para componente conexa por caminho. Os subconjuntos do \mathbb{R}^n têm a topologia induzida pela topologia usual no \mathbb{R}^n . Como referência para os conceitos algorítmicos e notação assintótica (como Ω e O) sugerimos [7], para os conceitos topológicos sugerimos [11].

2. ÁRVORES

Dado $S \subseteq \mathbb{R}^n$, chamamos de *árvore de decisão* para S uma árvore enraizada cujos nós internos têm 3 filhos, as folhas são rotuladas com **sim** ou **não** e respondem corretamente a pergunta “ $x \in S$?”. As arestas têm rótulos em $\{<, =, >\}$ e os nós internos são rotulados com elementos de uma família \mathcal{F} de funções definidas no \mathbb{R}^n . O resultado da comparação de $f(x)$ com 0, para $f \in \mathcal{F}$, denotada por $f: 0$ em que $:$ $\in \{<, =, >\}$, determina em qual dos três filhos do nó corrente a computação prossegue até chegar num nó folha da árvore.

Assim, uma árvore computa uma função da seguinte forma: dado $x \in \mathbb{R}^n$, dependendo do sinal de $f_v(x)$ no nó v a computação segue pelo filho correspondente de v , começando da raiz da árvore até atingir uma folha onde temos o valor da função para x .

Árvores de decisão linear. Dados um poliedro $S \subseteq \mathbb{R}^n$ e $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ queremos decidir se $x \in S$ usando testes da família \mathcal{F} das funções afins definidas no \mathbb{R}^n . Uma *árvore de decisão linear* é uma árvore de decisão onde cada nó interno v está associado a uma função afim $\ell_v(x) = \sum_i a_i x_i + b$.

Exemplo 2 (Arranjo de hiperplanos). Como exemplo, considere os seguintes subconjuntos do \mathbb{R}^3 : $H_1 = \{(x, y, z) : x = 0\}$; $H_2 = \{(x, y, z) : x - y = 0\}$; $H_3 = \{(x, y, z) : x + y = 0\}$. Uma árvore de decisão linear para $S = \bigcup_{i=1}^3 H_i$ é mostrada na figura 2 abaixo.

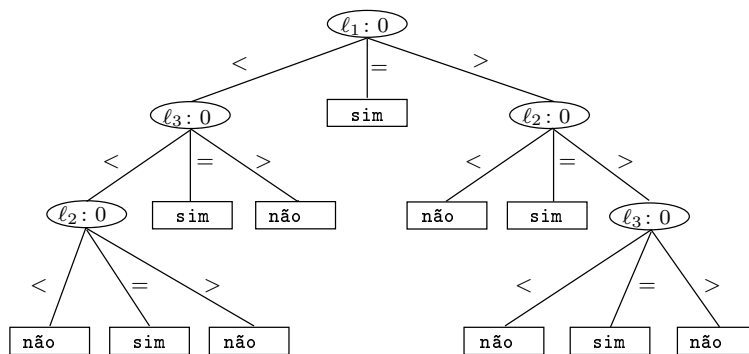


FIGURE 2. Exemplo de uma árvore de decisão linear, onde $\ell_1(x, y, z) = x$, $\ell_2(x, y, z) = x - y$ e $\ell_3(x, y, z) = x + y$.

Se conseguirmos mostrar um limitante inferior para a altura de qualquer uma dessas árvores que modelam algoritmos que decidem “ $x \in S$?”, teremos um limitante inferior para o número de testes efetuados por *qualquer* algoritmo de decisão baseado em testes lineares. A parte interessante aqui é que a complexidade do algoritmo está, num certo sentido, amarrada com a complexidade topológica de S . No exemplo acima o complemento $\mathbb{R}^3 \setminus S$ tem seis componentes conexas o que implica que toda árvore de decisão linear deve ter pelo menos seis folhas não, pois o conjunto de pontos que terminam a computação numa determinada folha é conexo.

Assim, não é difícil mostrar que a altura de qualquer árvore de decisão linear associada ao algoritmo $\mathcal{A}: \mathbb{R}^n \rightarrow \{\text{sim}, \text{não}\}$ que decide pertinência em S é pelo menos proporcional ao logaritmo da soma do número de componentes conexas de S mais o número de componentes conexas de $\mathbb{R}^n \setminus S$.

Exemplo 3 (Elementos distintos). Se o problema é decidir se $(x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ não tem duas coordenadas iguais, o número de componentes conexas do conjunto de pontos que atingem alguma folha **sim** é $n!$ (por quê?). Portanto a altura de qualquer árvore é $\Omega(\log n!) = \Omega(n \log n)$. Um algoritmo para esse problema que faz $O(n \log n)$ comparações entre elementos da entrada é: ordene usando $O(n \log n)$ comparações e verifique se posições consecutivas são idênticas usando $O(n)$ comparações.

Árvores de decisão algébrica. Uma *árvore de decisão algébrica de ordem d* é uma árvore de decisão onde cada nó interno v é rotulado com um polinômio não-nulo $p_v \in \mathbb{R}[x_1, x_2, \dots, x_n]$ de grau no máximo d .

Exemplo 4 (Elementos distintos). Decidir se os números reais x_1, x_2, \dots, x_n são todos distintos é equivalente a decidir pertinência em

$$(1) \quad S = \left\{ (x_1, x_2, \dots, x_n) \in \mathbb{R}^n : \prod_{1 \leq i < j \leq n} (x_i - x_j) \neq 0 \right\}.$$

O número de componentes conexas de S é pelo menos $n!$. De fato, se (a_1, a_2, \dots, a_n) são distintos entre si e $x = (x_1, x_2, \dots, x_n)$ e $y = (y_1, y_2, \dots, y_n)$ são duas permutações distintas dos a_i 's, então x e y estão em componentes conexas distintas. Uma justificativa é que para i, j tais que $x_i < x_j$ e $y_i > y_j$ qualquer caminho contínuo de x para y contém um ponto $z \in \mathbb{R}^n$ onde $z_i = z_j$, logo $z \notin S$.

Exemplo 5 (Problema da Mochila). Dado $(x_1, \dots, x_n) \in \mathbb{R}^n$, decidir se existe um conjunto de índices $I \subseteq \{1, 2, \dots, n\}$ tal que $\sum_{i \in I} x_i = 1$ é equivalente a decidir pertinência em

$$(2) \quad S = \left\{ (x_1, x_2, \dots, x_n) \in \mathbb{R}^n : \prod_I \left(\sum_{i \in I} x_i - 1 \right) \neq 0 \right\}.$$

O número de componentes conexas de $\mathbb{R}^n \setminus S$ é $2^{\binom{n}{2}+1}$ (veja [8]).

O fato, que veremos com mais precisão adiante, de que qualquer subconjunto $S \subseteq \mathbb{R}^n$ definido por m equações e h inequações polinomiais de grau no máximo d , $d \geq 2$ fixo, ter $O(d^{n+h})$ componentes conexas mostra o seguinte. A altura de qualquer árvore de decisão algébrica de ordem d para $\mathcal{A}: \mathbb{R}^n \rightarrow \{\text{sim}, \text{não}\}$ que decide pertinência em S é pelo menos proporcional a $\log(\beta_0(S) + \beta_0(\mathbb{R}^n \setminus S)) - n$, onde $\beta_0(X)$ é o número de componentes conexas de $X \subseteq \mathbb{R}^n$. Aplicado ao exemplo 4 acima, esse resultado diz que qualquer árvore de decisão algébrica de ordem fixa que decide o problema dos elementos distintos tem altura $\Omega(n \log n)$.

Árvores de computação algébrica. Com relação ao exemplo 4, cabe observarmos que computar o discriminante

$$\prod_{1 \leq i < j \leq n} (x_i - x_j)$$

pode ser feito num algoritmo sem ramificações em tempo $O(n \log^2 n)$, se contarmos as operações aritméticas usadas pela Transformada Rápida de Fourier. Esse fato mais uma comparação com 0 fornece um algoritmo para o problema dos elementos distintos.

A falta de um modelo indicando que mais que $O(1)$ operações eram necessárias motivou Ben-Or em [2] a definir *árvores de computação algébrica*, um modelo que permite outro tipo de nó além dos nós de ramificação. A árvore tem nós de computação e que leva em conta os custos das operações \times , \div e $\sqrt{}$.

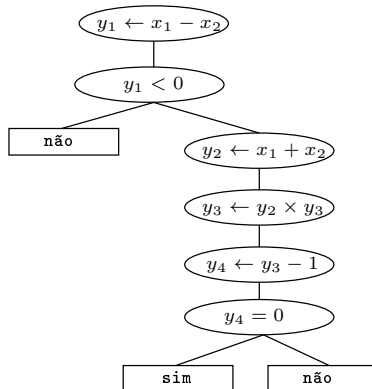
Um *nó de computação* v está associado a uma operação e uma atribuição, denotada por \leftarrow , dentre

$$y_v \leftarrow y_u \text{ op } y_z, \quad y_v \leftarrow \sqrt{y_u}, \quad y_v \leftarrow a \text{ op } y_z$$

onde $\text{op} \in \{+, -, \times, \div\}$, u e z ou são *ancestrais* de v na árvore, isto é, nós que aparecem no caminho da raiz até v , ou $y_u, y_z \in \{x_1, x_2, \dots, x_n\}$ (elementos da entrada) e a é uma constante real. Obviamente, não permitimos divisão por zero e raiz quadrada de número negativo. Um nó de computação tem um único filho.

Um *nó de ramificação* v está associado a um teste $y_u > 0$, ou $y_u \geq 0$ ou $y_u = 0$ onde u é um ancestral de v ou $y_u \in \{x_1, x_2, \dots, x_n\}$. O nó v tem *dois* filhos e a computação segue pelo filho da esquerda se o teste em v resultou *verdadeiro*, caso contrário, a computação segue pelo filho da direita. As folhas são rotuladas com **sim** ou **não**.

Dado $(x_1, x_2, \dots, x_n) \in \mathbb{R}^n$, uma computação começa na raiz da árvore e em cada nó realizamos ou a operação aritmética associada ao nó e seguimos para o filho, ou realizamos o teste associado ao nó e prosseguimos para um dos filhos de acordo o descrito no parágrafo anterior. Quando atingimos uma folha temos a resposta correta para um problema de pertinência. Veja um exemplo de árvore de computação na figura 3 abaixo.

FIGURE 3. Árvore de computação para $x_1^2 - x_2^2 = 1$.

Como no caso da árvore de decisão algébrica, a altura de qualquer árvore de computação algébrica que computa $\mathcal{A}: \mathbb{R}^n \rightarrow \{\mathbf{sim}, \mathbf{não}\}$ é pelo menos proporcional a $\log(\beta_0(S) + \beta_0(\mathbb{R}^n \setminus S)) - n$. Aplicado ao exemplo 4 esse resultado diz que qualquer árvore de computação algébrica que computa o discriminante $\prod_{i < j} (x_i - x_j)$ tem altura $\Omega(n \log n)$.

Notação. Se T é uma árvore de decisão ou computação para S , denotamos por S_w o conjunto de pontos de \mathbb{R}^n que em T terminam a computação na folha w .

No exemplo 2, se w é a folha do caminho $(\ell_1, \ell_2, \ell_3, \mathbf{sim})$ então S_w é a semirreta dada por $(\lambda, -\lambda, 0) \in \mathbb{R}^3$ para todo $\lambda > 0$; se w é a folha do caminho $(\ell_1, \ell_3, \mathbf{sim})$ então S_w é a semirreta $(\lambda, -\lambda, 0)$ para todo $\lambda < 0$; se w é a folha do caminho $(\ell_1, \ell_2, \ell_3, \mathbf{não})$ então S_w é o aberto de $(x, y, 0)$ dado por $x > |y|$.

De um modo geral, uma estratégia que associa altura de árvore à complexidade topológica é que para $S \subseteq \mathbb{R}^n$ e T uma árvore de decisão algébrica de ordem d e altura h para decidir pertinência em S , sejam $L^s = L^s(T)$ e $L^n = L^n(T)$ o conjunto das folhas \mathbf{sim} e $\mathbf{não}$, respectivamente, de T . Dessa maneira, temos

$$S = \bigcup_{w \in L^s} S_w \quad \text{e} \quad \mathbb{R}^n \setminus S = \bigcup_{w \in L^n} S_w.$$

Agora, se α é uma “medida” subaditiva da complexidade topológica de subconjuntos do \mathbb{R}^n e se $\alpha(S_w) = O(\exp(n + h))$, então

$$\alpha(S) \leq \sum_{w \in L^s} \alpha(S_w) \leq |L^s| O(\exp(n + h)) \leq 3^h O(\exp(n + h)),$$

ou seja, $h = \Omega(\log \alpha(S) - n)$. Essa medida α pode ser, por exemplo, a característica de Euler $\chi(S)$ ou o número de componentes conexas por caminho $\beta_0(S)$.

A seguir, todos os espaços têm tipo homotópico de conjuntos semialgébricos. Essa classe de espaços coincide com a dos espaços que têm tipo homotópico de complexo simplicial finito e, portanto, seus grupos de homologia singular $H_i(S) =$

$H_i(S, \mathbb{Z})$ são abelianos finitamente gerados cuja dimensão, os números de Betti $\beta_i(S)$, é finita e não-nula para uma quantidade finita de índices i ($i \geq 0$).

3. COTAS INFERIORES TOPOLÓGICAS

As primeiras estimativas de cotas inferiores para a altura de árvores de decisão que estabelecem ligações entre a complexidade computacional para decidir pertinência em S e os invariantes topológicos de S dependem da contagem do número de componentes conexas de S e/ou do complemento $\mathbb{R}^n \setminus S$.

Esse método foi introduzido por Dobkin e Lipton em 1979 para árvores de decisão linear, depois foi estendido para árvores de decisão algébrica por Steele e Yao em 1982 e logo em seguida, em 1983, Ben-Or modificou e estendeu esses resultados.

Dada uma árvore de decisão algébrica T de ordem d e altura h para testar pertinência em S formado pela união disjunta de abertos conexos $\{O_i\}_{i \in I}$, seja $L \subseteq L^s$ o subconjunto das folhas sim de T tais que no caminho da raiz de T até a folha aparecem somente desigualdades estritas. Assim, para $w \in L$ vale que

- ($d = 1$) no caso de árvores de decisão linear S_w é convexo. Logo não pode ser que ambos $S_w \cap O_i$ e $S_w \cap O_j$ sejam não-vazios para $i \neq j$. Portanto, $|I| \leq |L| \leq 3^h$;
- ($d > 1$) cada O_i deve conter pelo menos uma componente conexa de S_w , para alguma folha $w \in L$. Isso nos diz que $|I| \leq |L|\beta(dh, n) \leq 3^h\beta(dh, n)$, onde $\beta(m, n)$ é o número máximo de componentes conexas num conjunto da forma $\mathbb{R}^n \setminus p^{-1}(0)$ para p polinômio de grau m .

Esses argumentos estabelecem os seguintes resultados pioneiros.

Teorema 6 (Dobkin e Lipton '79, [9]). *Qualquer árvore de decisão linear para o problema de pertinência num aberto S do \mathbb{R}^n tem altura $\Omega(\log \beta_0(S))$.*

Teorema 7 (Steele e Yao '82, [14]). *Se T é uma árvore de decisão algébrica de ordem d para o problema de pertinência para um aberto S do \mathbb{R}^n , então a altura h de T satisfaz $3^h\beta(hd, n) \geq \beta_0(S)$.*

Agora, seja T uma árvore de computação de altura h e (v_1, \dots, v_t, w) uma seqüência de nós da raiz v_1 até uma folha w , com $t \leq h$. Essa seqüência define, de acordo com as tabelas abaixo, um sistema polinomial. Para cada nó de computação (tabela 1) temos uma equação algébrica e para cada nó de ramificação (tabela 2) temos uma equação (ou inequação) que depende do resultado do teste realizado no nó. No caso de um nó de ramificação com rótulo do tipo $y_v = 0$, se o teste for falso então a equação correspondente é da forma $y_v z_u - 1 = 0$, onde z_u é uma variável nova. Sejam y_1, \dots, y_r , para algum $r = r(w)$, as variáveis criadas nesse percurso e $s = s(w)$ o número de desigualdades no sistema polinomial, $r + s \leq t$. As soluções $(x_1, \dots, x_n, y_1, \dots, y_r) \in \mathbb{R}^{n+r}$ desse sistema polinomial definem um conjunto semialgébrico $R_w \subset \mathbb{R}^{n+r}$. No sistema polinomial o grau de qualquer polinômio é no máximo 2.

operação	equação
$y_v \leftarrow y_u + y_z$	$y_v = y_u + y_z$
$y_v \leftarrow y_u - y_z$	$y_v = y_u - y_z$
$y_v \leftarrow y_u \times y_z$	$y_v = y_u y_z$
$y_v \leftarrow y_u \div y_z$	$y_u = y_v y_z$
$y_v \leftarrow \sqrt{y_u}$	$y_u = y_v^2$

TABLE 1. À operação na coluna esquerda corresponde a equação na coluna direita.

rótulo	verdadeiro	falso
$y_v > 0$	$y_v > 0$	$-y_v \geq 0$
$y_v \geq 0$	$y_v \geq 0$	$-y_v > 0$
$y_v = 0$	$y_v = 0$	$y_v z_u - 1 = 0$

TABLE 2. Ao teste do rótulo associa-se uma (in)equação que depende do resultado.

A projeção de R_w nas n primeiras coordenadas é o conjunto $S_w \subset \mathbb{R}^n$ formado pelos pontos do \mathbb{R}^n que terminam a computação na folha w da árvore de computação T e é semialgébrico (veja [1] seção 2.4). Além disso, a projeção não aumenta o número de componentes conexas, ou seja, $\beta_0(S_w) \leq \beta_0(R_w)$.

Usando a notação

$$(3) \quad R = \{x \in \mathbb{R}^m : q_i(x) = 0 \ (1 \leq i \leq \ell), \ p_j(x) > 0 \ (1 \leq j \leq s)\},$$

onde $q_i, p_j \in \mathbb{R}[x_1, \dots, x_m]$ para todo $1 \leq i \leq \ell$ e $1 \leq j \leq s$, e

$$(4) \quad \beta_d(s, m) = \max \{ \beta_0(R) : R \text{ é definido em (3)} \},$$

onde $d = \max\{2, \text{grau}(q_i), \text{grau}(p_j) : 1 \leq i \leq \ell \text{ e } 1 \leq j \leq s\}$,

temos que o número de componentes conexas de R_w é no máximo $\beta_2(s(w), n + r(w))$.

Teorema 8 (Ben-Or '83, [2]). *Se T_c é uma árvore de computação algébrica de altura h_c e T_d uma árvore de decisão algébrica de ordem $d \geq 2$ e altura h_d para o problema de pertinência num conjunto $S \subset \mathbb{R}^n$. Então*

$$\beta_0(S) \leq 2^{h_c} \beta_2(h_c, n + h_c) \quad e \quad \beta_0(S) \leq 3^{h_d} \beta_d(h_d, n).$$

Esses resultados fornecem cotas que são triviais caso S e o seu complemento sejam de grande complexidade topológica mas conexos. O próximo passo é tentar estabelecer relações que envolvem a altura das árvores com os números de Betti de dimensões maiores que 0. Essa relações poderiam levar a resultados não-triviais para alguns problemas onde o conjunto alvo da pertinência, $S \subseteq \mathbb{R}^n$, é conexo.

Exemplo 9 (k -iguais). Considere a seguinte pergunta: qual a altura mínima de uma árvore de decisão linear para decidir se na seqüência x_1, x_2, \dots, x_n existem dois elementos iguais? Essa pergunta é equivalente a decidir se todos os elementos da entrada são diferentes (exemplo 4). Portanto, segue imediatamente do teorema 6 que a resposta para a pergunta é $\Omega(\log n!) = \Omega(n \log n)$.

E para decidir se existem $k > 2$ iguais? Nesse caso, dado o ponto $(x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ queremos decidir se esse ponto pertence ao arranjo

$$(5) \quad \mathcal{A}_{n,k} = \{H_\lambda\}_\lambda$$

dos $\binom{n}{k}$ subespaços lineares de dimensão $n - k + 1$ dados por

$$H_\lambda = \{(x_1, x_2, \dots, x_n) \in \mathbb{R}^n : x_{i_1} = x_{i_2} = \dots = x_{i_k}\}$$

para todo conjunto de k índices $\lambda = \{i_1, i_2, \dots, i_k\} \subseteq \{1, 2, \dots, n\}$.

O leitor pode perceber que

$$V_{\mathcal{A}_{n,k}} = \bigcup_{\lambda} H_\lambda \quad \text{e} \quad M_{\mathcal{A}_{n,k}} = \mathbb{R}^n \setminus V_{\mathcal{A}_{n,k}}$$

são conjuntos conexos.

Uma cota inferior não trivial para o problema dos k -iguais foi dada por Björner, Lovász e Yao cuja abordagem vamos descrever.

Considere $\widehat{\mathbb{R}}^n$ a compatificação do \mathbb{R}^n com ω “no infinito” e tome $\widehat{S} = S \cup \{\omega\}$, onde S é um poliedro fechado.

Seja T uma árvore de decisão linear para S . Note que, para cada folha w , a dimensão do poliedro convexo S_w está bem definida pois esse é aberto no seu fecho afim. De fato, o fecho afim \mathbb{A}_w é dado pela intersecção dos hiperplanos $\ell_u(x) = 0$, onde ℓ_u é um teste que resulta em igualdade no caminho da raiz até a folha w em T . Chamamos S_w de *célula* correspondente à folha w ; se $\dim(S_w) = n$ dizemos n -célula. Se essas células formam um CW-complexo no $\widehat{\mathbb{R}}^n$ então

$$\sum_{w \in L^n} (-1)^{\dim(S_w)} = \chi(S^n) - 1 - (\chi(\widehat{S}) - 1) = -\chi(\widehat{S}) + 1 + (-1)^n$$

e que

$$\sum_{w \in L^s} (-1)^{\dim(S_w)} = \chi(\widehat{S}) - 1.$$

Um problema técnico dessa idéia é que as células não necessariamente definem um complexo celular regular. Isso pode ser contornado tomando o arranjo $\mathcal{A} = \{H_u\}_{u \in T}$, onde $H_u = \{x \in \mathbb{R}^n : \ell_u(x) = 0\}$, que particiona o \mathbb{R}^n em regiões polidrais. O fecho das regiões definidas pelos hiperplanos induzem uma decomposição celular do \mathbb{R}^n . Essa solução leva ao seguinte resultado.

Teorema 10 (Björner, Lovász e Yao '92, [3]). *Seja S um poliedro fechado no \mathbb{R}^n . Em qualquer árvore de decisão linear T para S*

$$|L^s| \geq |\chi(\widehat{S}) - 1| \quad \text{e} \quad |L^n| \geq |\chi(\widehat{S}) - 1 + (-1)^{n-1}|,$$

e, portanto, tem altura $\Omega(\log |\chi(\widehat{S})|)$.

Os autores em [3] destacam a possibilidade desse resultado ser estendido para árvores de decisão algébrica. Essa extensão foi dada por Yao [15]. No resultado a seguir, as constantes foram dadas explicitamente.

Teorema 11 (Yao '92, [15]). *Para qualquer semialgébrico compacto $S \subseteq \mathbb{R}^n$.*

- *A altura de uma árvore de decisão algébrica de ordem d para pertinência em S é pelo menos*

$$c_1 \log_2 |\chi(S)| - c_2 n - c_3,$$

onde $c_i = c_i(d)$, para todo i .

- *A altura de uma árvore de computação algébrica para pertinência em S é pelo menos*

$$c_4 \log_2 |\chi(S)| - c_5 n - c_6.$$

A demonstração desse resultado está baseada num teorema de Hironaka a respeito de triangulações semialgébricas de subconjuntos semialgébricos limitados do \mathbb{R}^n e no fato demonstrado por Yao de χ estar limitado por uma função exponencial em n e h (veja (9), pág. 16). A hipótese de S ser compacto pode ser eliminada tomando-se a projeção estereográfica inversa de S no \mathbb{R}^{n+1} , que continua sendo um semialgébrico; no caso de R definido na equação (3) a imagem estereográfica inversa é definida por $\ell + 1$ equações e $s + 2$ inequações de grau no máximo d . Não entraremos nos detalhes desse resultado.

Esses limitantes baseados em características de Euler ainda podem dar respostas triviais. No caso de $M_{A_n, k}$ do exemplo 9 os números de Betti foram computados para vários valores de n e k por Björner e Welker em [6]. Por exemplo, para $n = 11$ e $k = 3$

$$\left(\beta_i(M_{11,3}) \right)_{i=0}^{11} = (1, 18943, 1005812, 986870, 0, 0, 0, 0, 0, 0, 0, 0)$$

o que resulta em $\chi(M_{11,3}) = 0$ e temos um cota trivial para o número de folhas.

Agora, se escrevemos V_i para o espaço vetorial gerado pelas i -células e $\partial_i : V_i \rightarrow V_{i-1}$ para o homomorfismo bordo usual, então

$$\dim H_i(\mathbb{R}^n \setminus S) \leq \dim(\ker \partial_i) \leq \dim V_i = \text{número de } i\text{-células.}$$

Concluimos que o número de folhas não é pelo menos $\sum_{i \geq 0} \beta_i(\mathbb{R}^n \setminus S)$. Essa idéia foi comprovada pelo seguinte resultado devido a Björner e Lovász.

Teorema 12 (Björner e Lovász '94, [4]). *Para qualquer poliedro fechado $S \subseteq \mathbb{R}^n$, e para qualquer árvore de decisão linear T para S*

$$|L^n| \geq \sum_{i=0}^n \beta_i(\mathbb{R}^n \setminus S) \quad e \quad |L^s| \geq -1 + \sum_{i=0}^n \beta_i(\mathbb{R}^n \setminus S).$$

Como exemplo vamos considerar o *problema dos 3-iguais*. Enquanto que a cota inferior baseada na característica de Euler é trivial, a cota baseada na soma dos números de Betti nos dá que qualquer árvore de decisão linear tem pelo menos 4.023.251 folhas, portanto altura pelo menos 13.

Um argumento baseado no teorema de dualidade de Alexander¹ em $S^n \cong \widehat{\mathbb{R}^n}$, fornece o seguinte resultado.

Corolário 13. *Para qualquer poliedro fechado $S \subseteq \mathbb{R}^n$, e para qualquer árvore de decisão linear T para S*

$$|L^n| \geq \sum_{i=0}^n \beta_i(\widehat{S}) \quad e \quad |L^s| \geq -1 + \sum_{i=0}^n \beta_i(\widehat{S}).$$

Logo, a altura de T é $\Omega(\log(\sum_{i \geq 0} \beta_i(\widehat{S})))$.

Vamos descrever o ponto chave na demonstração do teorema 12. Denotamos por $f^n(n-i, T)$ o número de folhas não em T cuja célula correspondente tem dimensão $n-i$. Se T é uma árvore com mais de um vértice e $\ell(x)$ o teste feito na raiz de T então definimos

$$\begin{aligned} H^+ &= \{x \in \mathbb{R}^n : \ell(x) > 0\}, \\ H^- &= \{x \in \mathbb{R}^n : \ell(x) < 0\}, \\ H &= \{x \in \mathbb{R}^n : \ell(x) = 0\} \equiv \mathbb{R}^{n-1} \end{aligned}$$

e denotamos por T^+ , T^- e T^0 as subárvores cujas raízes são os filhos da raiz de T e que testam pertinência em S para pontos em H^+ , H^- e H respectivamente.

Tomamos, para todo x em $H \setminus S$, a bola aberta O_x de centro x e tal que $O_x \subseteq \mathbb{R}^n \setminus S$ e definimos $C = \bigcup O_x$, onde a união é tomada sobre todo $x \in H \setminus S$.

Se $A^+ = (H^+ \setminus S) \cup C$ e $A^- = (H^- \setminus S) \cup C$ então usando a seqüência exata longa de Mayer-Vietoris

$$\dots \xrightarrow{\partial_*} H_i(A^+ \cap A^-) \xrightarrow{l_*} H_i(A^+) \oplus H_i(A^-) \xrightarrow{j_*} H_i(A^+ \cup A^-) \xrightarrow{\partial_*} H_{i-1}(A^+ \cap A^-) \xrightarrow{l_*} \dots,$$

donde tiramos

$$\beta_i(A^+ \cup A^-) \leq \beta_i(A^+) + \beta_i(A^-) + \beta_{i-1}(A^+ \cap A^-).$$

As equivalências homotópicas abaixo, denotadas por \simeq , dadas pela projeção ortogonal na direção de H

- (i) $A^+ \cup A^- = (H^+ \setminus S) \cup (H^- \setminus S) \cup C = \mathbb{R}^n \setminus S$,
- (ii) $A^+ \simeq H^+ \setminus S$,
- (iii) $A^- \simeq H^- \setminus S$,
- (iv) $A^+ \cap A^- = C \simeq H \setminus S$,

¹ Para a esfera o enunciado é: $H^{n-i-1}(A) \cong H_i(S^n \setminus A)$ para todo fechado triangulável $A \subset S^n$ e todo i , $1 \leq i \leq n-2$.

nos dá a desigualdade

$$\beta_i(\mathbb{R}^n \setminus S) \leq \beta_i(H^+ \setminus S) + \beta_i(H^- \setminus S) + \beta_{i-1}(H \setminus S)$$

que juntamente com um argumento indutivo e

$$\begin{aligned} f^n(n-i, T) &= f^n(n-i, T^+) + f^n(n-i, T^-) + f^n(n-i, T^0) \\ &\geq \beta_i(H^+ \setminus S) + \beta_i(H^- \setminus S) + \beta_{i-1}(H \setminus S), \end{aligned}$$

mostra que

$$f^n(n-i, T) \geq \beta_i(\mathbb{R}^n \setminus S).$$

□

Novamente, Yao estendeu esse resultado para semialgêbricos. A principal dificuldade técnica foi a não-subaditividade dos números de Betti. Não é difícil ver que β_1 , por exemplo, não é subaditivo: considere S um buquê de três esferas de dimensão um no plano com um ponto x em comum. Agora tome $S = (S \setminus \{x\}) \cup \{x\}$. Ainda, $\beta_1(S) = 3$, $\beta_1(S \setminus \{x\}) = 0$ e $\beta_1(\{x\}) = 0$ (figura 4).

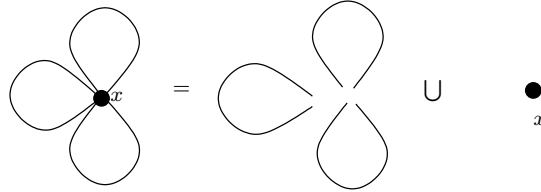


FIGURE 4. Buquê de três esferas como união de dois subconjuntos com β_1 trivial.

Para ver como o problema da não-subaditividade dos números de Betti foi contornado vamos dar algumas definições. As propriedades topológicas dizem respeito a topologia do $\widehat{\mathbb{R}^n}$.

Definimos, para cada $i \in \mathbb{N}$, o número $\beta'_i(S) = \beta'_i(\overline{S}, \delta S)$, a dimensão do grupo de homologia $H_i(\overline{S}, \delta S)$ do fecho de S , denotado por \overline{S} , relativo à fronteira $\delta S = \overline{S} \setminus S$. Dizemos que S é *semifechado* se for a diferença de dois fechados. Observamos que nesse caso δS é fechado.

Assim, temos o seguinte resultado para os números de Betti relativo: Se Q é um subconjunto semialgêbrico e semifechado do $\widehat{\mathbb{R}^n} \setminus \{\omega\}$, $q \in \mathbb{R}[x_1, \dots, x_n]$ e $A = \{x \in Q : q(x) \leq 0\}$ e $B = Q - A$, então

$$(6) \quad \beta'_i(Q) \leq \beta'_i(A) + \beta'_i(B).$$

Um esboço da prova é como segue. As inclusões $\delta Q \subseteq \overline{A} \cup \delta Q \subseteq \overline{Q}$ induzem a seqüência exata longa

$$\dots \xrightarrow{\partial_*} H_i(\overline{A} \cup \delta Q, \delta Q) \rightarrow H_i(\overline{Q}, \delta Q) \rightarrow H_i(\overline{Q}, \overline{A} \cup \delta Q) \xrightarrow{\partial_*} \dots$$

e, portanto

$$(7) \quad \beta'_i(\overline{Q}, \delta Q) \leq \beta'_i(\overline{A} \cup \delta Q, \delta Q) + \beta'_i(\overline{Q}, \overline{A} \cup \delta Q).$$

Agora, seja $V \cup W$ um complexo simplicial que define uma triangulação de $\overline{A} \cup \delta Q$, com V e W sub-complexos que triangulam os fechados \overline{A} e δX respectivamente. De $\beta_i(W, V \cap W) = \beta_i(V \cup W, V)$ (Teorema da Excisão) temos a equação (6).

Com esse resultado, Yao provou o seguinte.

Teorema 14 (Yao '94, [16]). *Se S é um subconjunto do $\widehat{\mathbb{R}^n} \setminus \{\omega\}$ semialgébrico e semifechado, e L^s o conjunto das folhas *sim* de uma árvore de decisão algébrica de ordem d para testar pertinência em S , então*

$$(8) \quad \beta'_i(S) \leq \sum_{w \in L^s} \beta'_i(S_w).$$

Esboço da prova. Dados uma árvore T para S e chamando de p o polinômio que é testado na raiz de T , definimos os conjuntos

$$\begin{aligned} S^- &= \{x \in S : p(x) < 0\} \\ S^0 &= \{x \in S : p(x) = 0\} \\ S^+ &= \{x \in S : p(x) > 0\}. \end{aligned}$$

Aplicando a desigualdade (7) para $A = S^- \cup S^0$ e $B = S \setminus A$ e depois para $A = S^- \cup S^0$ e $B = S^0$, temos

$$\beta'_i(S) \leq \beta'_i(S^- \cup S^0) + \beta'_i(S^+) \leq \beta'_i(S^-) + \beta'_i(S^0) + \beta'_i(S^+).$$

Por indução temos (8). □

Daqui por diante a situação é como nos outros casos: a propriedade subaditiva acima com um limitante $\beta'_d(h, n)$ para $\sum_i \beta'_i(S_w)$ para toda folha w , resulta em

$$\sum_{i \geq 0} \beta'_i(S) \leq \sum_{i \geq 0} \sum_{w \in L^s} \beta'_i(S_w) \leq |L^s| \beta'_d(h, n),$$

e que novamente nos levará a uma cota do tipo $\Omega(\log \sum_i \beta'_i(S) - n)$ para a altura de qualquer árvore de decisão algébrica de ordem d , para d fixo, que decide pertinência num compacto $S \subseteq \mathbb{R}^n$ (veja (10), pág. 16).

4. COMPUTANDO OS NÚMEROS DE BETTI

No que segue, como é usual, denotamos por $\tilde{\beta}_i$ a dimensão do i -ésimo grupo de homologia reduzido $\tilde{H}(\cdot, \mathbb{Z})$.

Para as aplicações, as cotas topológicas da seção anterior dependem de boas estimativas para os números de Betti dos espaços envolvidos. Os números de Betti podem ser computado explicitamente sempre que uma decomposição celular apropriada de S pode ser construída eficientemente, pois $\dim(H_i(S)) \leq \dim(C_i(S))$, onde C_i é o complexo das cadeias de dimensão i . Se houverem n_i células de dimensão i então $\beta_i(S) \leq n_i$. Ainda, há as ferramentas do dia-a-dia como, por exemplo, seqüência exata do par, seqüência de Mayer-Vietoris e seqüências espectrais.

Nessa seção, daremos limitantes para os conjuntos semialgêbricos e mostraremos algumas táticas combinatórias para computar os números de Betti de arranjos de subespaços.

4.1. Semialgêbricos. Limitantes para os números de Betti de semialgêbricos são conhecidos desde os resultados pioneiros de Oleinik, Petrovsky, Milnor e Thom.

Steele e Yao usaram nas aplicações do teorema 7 o seguinte limitante de [12]: sejam $p_1, p_2, \dots, p_k \in \mathbb{R}[x_1, x_2, \dots, x_n]$, se V é definido por $p_1 \geq 0, p_2 \geq 0, \dots, p_k \geq 0$ e a soma dos graus é d então

$$\sum_{i \geq 0} \beta_i(V) \leq \frac{1}{2}(d+2)(d+1)^{n-1},$$

e deduziram que a altura de qualquer árvore de decisão algébrica de ordem d fixa que decide pertinência em S é $\Omega(\log \beta_0(S))$.

Outro limitante de [12] é: para todo V definido por um número finito de equações polinomiais com cada polinômio de grau no máximo d

$$\sum_{i \geq 0} \beta_i(V) \leq d(2d-1)^{n-1}.$$

Como um conjunto semialgêbrico da forma (3) é a projeção de um conjunto algébrico em uma dimensão maior, Ben-Or usou o limitante acima e enunciou o seguinte.

Teorema 15 (Oleinik e Petrovsky '62, Milnor '64, Thom '65). *Para todo conjunto semialgêbrico R dado por ℓ equações e s inequações polinomiais com cada polinômio de grau no máximo d*

$$\sum_{i \geq 0} \beta_i(R) \leq d(2d-1)^{n+s-1}.$$

O limitante na soma dos números de Betti é, obviamente, um limitante para o número de componentes conexos. A conclusão de Ben-Or foi que toda árvore de decisão algébrica de ordem $d \geq 2$ fixa e toda árvore de computação algébrica para S tem altura $\Omega(\log \beta_0(S) - n)$ e $\geq 0, 38 \log N - 0, 61n$, respectivamente, onde $N = \max\{\beta_0(S), \beta_0(\mathbb{R}^n \setminus S)\}$.

Do teorema 15, Yao derivou ([15, 16])

$$(9) \quad |\chi(\widehat{R})| \leq d(2d-1)^{n+s+2},$$

$$(10) \quad \sum_{i \geq 0} \beta'_i(R) \leq (2d+1)^{n+2s+5}.$$

Em [1] são dadas várias estimativas recentes, por exemplo, Basu e outros limitaram conjuntos definidos por $S_j = \{x \in \mathbb{R}^n : p_j(x) \geq 0\}$, para $1 \leq j \leq s$. Se S é a união dos S_j então $\beta_i(S) \leq s^{i+1}O(d)^n$; se S é a intersecção dos S_j então $\beta_i(S) \leq s^{n-i}O(d)^n$. Além disso, são conhecidos algoritmos que calculam

número de Betti e característica de Euler de semialgêbricos, por exemplo, computa $\beta_\ell(\cup_i S_i)$, com $i \in [n]$, usando $O(n^{\ell+2})$ operações algébricas.

4.2. Conjuntos parcialmente ordenados e complexos simpliciais. A um conjunto (finito) parcialmente ordenado P associamos um complexo simplicial abstrato chamado *complexo da ordem* P denotado $\Delta(P)$. Este é um complexo simplicial cujos vértices são os elementos de P e os simplexes são as cadeias $x_0 < x_1 < \dots < x_k$ em P . Denotamos por \widehat{P} o conjunto parcialmente ordenado obtido a partir de P adicionando um mínimo $\widehat{0}$ e um máximo $\widehat{1}$. Como é usual, denotamos por $\mu(x, y)$ a *função de Möbius* no intervalo (x, y) de um conjunto parcialmente ordenado.

Um teorema famoso devido a Hall (veja [13]) estabelece que $\mu_{\widehat{P}}(\widehat{0}, \widehat{1}) = c_0 - c_1 + c_2 - \dots$ onde c_i é o número de cadeias $\widehat{0} = x_0 < x_1 < \dots < x_i = \widehat{1}$, portanto

$$(11) \quad \mu_{\widehat{P}}(\widehat{0}, \widehat{1}) = \chi(\Delta(P)) - 1 = \tilde{\chi}(\Delta(P)).$$

Sejam $x_1 < x_2 < \dots < x_r = x$ e $y = y_1 < y_2 < \dots < y_t$ cadeias saturadas em P (isto é, y_{i+1} cobre y_i e x_{j+1} cobre x_j) tal que x_1 é minimal e y_t é maximal e $x < y$ em P . Para o complexo da ordem $\Delta(x, y)$ definido pelo intervalo aberto $(x, y) = \{z \in P: x < z < y\}$ temos

$$(12) \quad \mu(x, y) = \tilde{\chi}(\Delta(x, y)) = \sum_i (-1)^i \tilde{\beta}_i(\Delta(x, y)).$$

Adiante, quando nos referimos a propriedades topológicas de P estamos, de fato, referindo-nos a propriedades do complexo da ordem de P . Ademais, do ponto de vista de notação, não faremos distinção entre complexo simplicial abstrato (sistema de conjuntos fechado por tomar subconjuntos) e o espaço topológico (realização geométrica do complexo) associado.

4.3. Descascabilidade. Seja Δ um complexo simplicial (abstrato), chamamos seus elementos de *faces* e de *facetras* suas faces maximais (por inclusão).

Para conjuntos $A \subseteq B$ definimos $[A, B] = \{C: A \subseteq C \subseteq B\}$ e $\overline{B} = [\emptyset, B]$ (portanto, um simplexo abstrato).

Dizemos que Δ é *descascável* se existe uma ordenação $\sigma_1, \sigma_2, \dots, \sigma_N$ das facetras tal que para todo $1 < i \leq N$ existe $\gamma_i \subseteq \sigma_i$ tal que

$$(13) \quad \overline{\sigma_i} \setminus (\overline{\sigma_1} \cup \overline{\sigma_2} \cup \dots \cup \overline{\sigma_{i-1}}) = [\gamma_i, \sigma_i].$$

Uma vantagem da descascabilidade é, por exemplo, é a seguinte descrição homotópica de um complexo descascável.

Teorema 16 (Björner e Wachs '92, [5]). *Um complexo simplicial Δ descascável tem o tipo homotópico de um buquê de esferas com n_k esferas de dimensão $k \geq 1$, onde n_k é o número de facetras σ_i de dimensão $k - 1$ tais que $\gamma_i = \sigma_i$.*

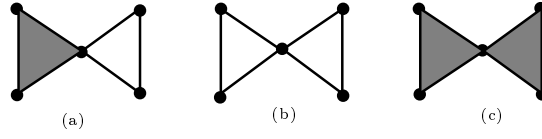


FIGURE 5. Exemplos de complexos simpliciais: (a) é descascável, (b) é descascável, e (c) não é descascável.

Exemplo 17 (Uma idéia para demonstração). Considere o complexo simplicial Δ definido pelo união dos simplexos maximais $\overline{\sigma_i}$, onde

$$\sigma_1 = \{a, c\}, \sigma_2 = \{a, b\}, \sigma_3 = \{b, c\}, \sigma_4 = \{c, d\}, \sigma_5 = \{d, e\}, \sigma_6 = \{e, f\}, \sigma_7 = \{f, d\}.$$

É apenas um pouco trabalhoso verificar que essa seqüência é um descascamento

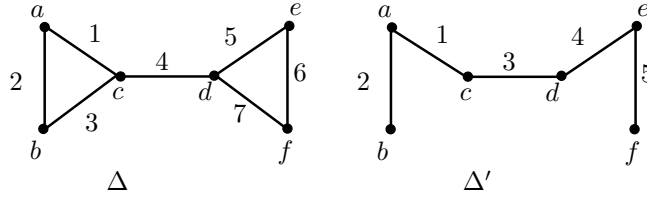


FIGURE 6. Δ e Δ' . Os números rotulando as arestas indicam a ordem que define um descascamento.

de Δ com $\gamma_1 = \{\emptyset\}, \gamma_2 = \{b\}, \gamma_3 = \{b, c\}, \gamma_4 = \{d\}, \gamma_5 = \{e\}, \gamma_6 = \{f\}, \gamma_7 = \{d, f\}$.

Façamos $\Delta' = \Delta \setminus \{\sigma_i : \gamma_i = \sigma_i\}$, ou seja, removemos de Δ as facetas σ_3 e σ_7 que têm $\gamma_i = \sigma_i$ em (13). Na figura 6 representamos os dois complexos simpliciais.

Notemos que as facetas que restaram em Δ' com a ordem induzida é um descascamento com os mesmos γ_i 's, ou seja, a seguinte ordenação das facetas de Δ' satisfaz (13)

$$(\sigma'_1, \sigma'_2, \sigma'_3, \sigma'_4, \sigma'_5) = (\sigma_1, \sigma_2, \sigma_4, \sigma_5, \sigma_6) \quad \text{e} \quad (\gamma'_1, \gamma'_2, \gamma'_3, \gamma'_4, \gamma'_5) = (\gamma_1, \gamma_2, \gamma_4, \gamma_5, \gamma_6).$$

Vejamos que Δ' é contraível. De fato, se $\Delta'_k = \cup_{i \leq k} \overline{\sigma'_i}$ é o subcomplexo dados pelas arestas da figura 6 com rótulo menor ou igual a k , então σ'_k contém o vértice γ'_k que não está em nenhuma outra aresta de Δ'_k . A remoção de γ'_k junto com as faces que o contém transforma Δ'_k em Δ'_{k-1} , ou seja, Δ'_k e Δ'_{k-1} são equivalentes homotopicamente. Resta-nos notar que $\Delta'_1 = \overline{\sigma'_1}$ é um 1-simplexo, portanto contraível.

Para recuperar Δ colamos os simplexos $\overline{\sigma_3} = \overline{\{b, c\}}$ e $\overline{\sigma_7} = \overline{\{d, f\}}$ em Δ' pelas suas fronteiras $\overline{\sigma_i} \setminus \sigma_i$, ou seja $\{b\}, \{c\}$ e $\{d\}, \{f\}$ respectivamente.

Agora, quando Δ' é deformando num ponto, os simplexos colados deformam-se numa esfera. Essas idéias, generalizadas e formalizadas, demonstram o teorema 16 acima.

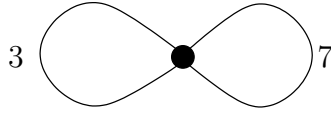


FIGURE 7. Buquê de esferas resultante de descascamento.

A definição de descascabilidade estende-se naturalmente para conjuntos parcialmente ordenados. Uma maneira de estabelecer que um conjunto parcialmente ordenado P é descascável é a seguinte: rotulamos cada aresta do diagrama de Hasse de P com elementos de um conjunto ordenado de modo que todo intervalo $[x, y]$

- (1) contém uma única cadeia maximal m^* com rótulos em ordem crescente,
- (2) qualquer outra cadeia maximal do intervalo tem a seqüência de rótulos *lexicograficamente* maior que m^* .

Agora, se P é uma ordem parcial lexicograficamente descascável então o complexo simplicial $\Delta(P \setminus \{\widehat{0}, \widehat{1}\})$ é descascável e, além disso, $\tilde{\beta}_{k-2}(\Delta(P \setminus \{\widehat{0}, \widehat{1}\}))$ é dado em função do número de cadeias maximais $\widehat{0} = x_0, x_i, \dots, x_k = \widehat{1}$ de comprimento k e tais que

$$(14) \quad \text{rótulo}(x_0x_1) \not\prec \text{rótulo}(x_1x_2) \not\prec \dots \not\prec \text{rótulo}(x_{k-1}x_k).$$

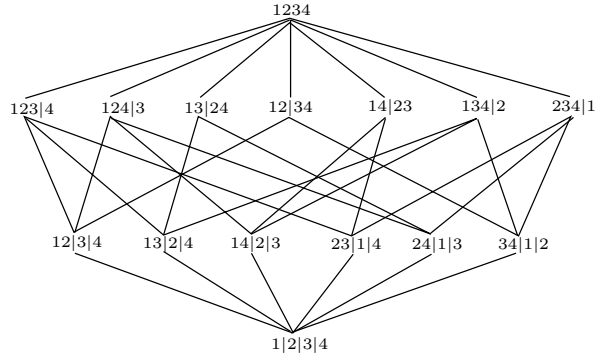
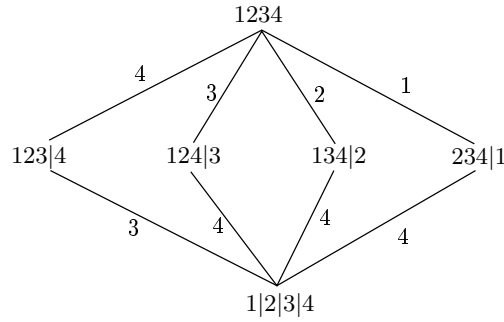
Teorema 18 (Björner e Wachs '92, [5]). *Se P é uma ordem parcial com $\widehat{0}$ e $\widehat{1}$ lexicograficamente descascável então o complexo simplicial $\Delta(P \setminus \{\widehat{0}, \widehat{1}\})$ é descascável. Além disso, para qualquer rotulação como acima $\tilde{H}_{k-2}(\Delta(P)) \cong \mathbb{Z}^n$ onde n é o número de cadeias da forma (14).*

Exemplo 19 ($\Pi_{n,k}$ é descascável). Considere o conjunto onde cada elemento é uma partição de $[n] = \{1, 2, \dots, n\}$ cujas partes têm tamanho em $\{1, k, k+1, \dots, n\}$. Ordenamos parcialmente esse conjunto por refinamento das partes, isto é, $\pi \leq \sigma$ em $\Pi_{n,k}$ se todo bloco de π está contido em um bloco de σ . Por exemplo, se $n = 8$, $\pi = 123|45|67|8$, $\sigma = 123|458|67$ e $\tau = 146|257|38$ então $\pi \leq \sigma$ e σ e τ são incomparáveis (figura 8).

Björner e Wachs [5] mostraram que rotulando as arestas do diagrama de Hasse dessa ordem parcial com elementos de $\bar{1} < \bar{2} < \dots < \bar{n} < 1 < 2 < \dots < n$ da forma descrita a seguir estabelece que $\Pi_{n,k}$ é lexicograficamente descascável. Para $\pi < \sigma$, (i) se uma nova parte B é criada em σ a partir de partes unitárias de π então o rótulo da aresta (π, σ) é $\max(B)$, (ii) se uma parte não-unitária é unida com uma parte unitária a então o rótulo da aresta (π, σ) é a , e (iii) se duas partes A e B não-unitárias são unidas então o rótulo da aresta (π, σ) é $\max(A \cup B)$.

Dessa forma podemos computar $\beta_i = \beta_i(\Delta(\Pi_{n,k}))$ contando as cadeias maximais da ordem parcial que satisfazem (14). Essa estratégia foi adotada em [5] onde calculou-se

$$\tilde{\beta}_i \neq 0 \Leftrightarrow i = n - 3 - t(k - 2),$$

FIGURE 8. Diagrama de Hasse de $\Pi_{4,2}$.FIGURE 9. Diagrama de Hasse de $\Pi_{4,3}$. A cadeia mais a esquerda tem rótulos em ordem crescente (de baixo pra cima) e todas as outras são lexicograficamente maior pois começam com 4.

para algum t , $1 \leq t \leq \lfloor n/k \rfloor$,

$$(15) \quad \tilde{\beta}_{n-3-t(k-2)} = (t-1)! \sum \prod_{j=0}^{t-1} \binom{n-jk-1-i_j}{k-1} (j+1)^{i_{j+1}-i_j}$$

onde a soma é sobre toda seqüência $0 \leq i_0 \leq i_1 \leq \dots \leq i_t = n - tk$.

Seja \mathbf{m} uma cadeia maximal que satisfaz (14). Uma cadeia dessa pode ser descrita em duas fases. O primeiro segmento de \mathbf{m} tem que ter rótulos sem barra e o segmento final rótulos com barra.

Na primeira fase, cada passo ou cria uma parte de tamanho k , que chamamos de k -bloco, ou une um ($> k$)-bloco com um 1-bloco. Seja t o número de partes no final da primeira fase.

A criação de um k -bloco diminui em $k-1$ o número de partes e os outros passos diminui em um o número de partes da partição. Portanto, o comprimento de \mathbf{m} é $n-1-t(k-2)$.

Se $i_{j+1} - i_j$ blocos unitários, $0 \leq j \leq t-2$, são unidos entre as criações dos $(j+1)$ -ésimo e $(j+2)$ -ésimo k -blocos da cadeia, e $i_t - i_{t-1}$ blocos unitários são

unidos após o t -ésimo k -bloco, o número de segmentos iniciais possíveis é

$$\prod_{j=0}^{t-1} \binom{n - jk - 1 - i_j}{k - 1} (j + 1)^{i_{j+1} - i_j},$$

pois a criação do $(j + 1)$ -ésimo k -bloco é a partir do máximo dos 1-blocos disponíveis com $k - 1$ outros quaisquer dos $n - jk - i_j - 1$ disponíveis, seguido da união em ordem decrescente dos $i_{j+1} - i_j$ maiores 1-bloco disponíveis com um dos $j + 1$ blocos não unitários disponíveis.

Para a segunda fase, consideramos os blocos B_1, B_2, \dots, B_t em ordem crescente de seus elementos máximos. Para respeitar (14), devemos unir B_t com algum dos $t - 1$ blocos restantes, seguido da união do bloco resultante com algum dos $t - 2$ restantes, e assim por diante. Isso pode ser feito de $(t - 1)!$ maneiras.

4.4. Uma cota combinatória para arranjos de subespaços. Um *arranjo* $\mathcal{A} = \{K_1, K_2, \dots, K_m\}$ é uma família finita de subconjuntos próprios não-vazios do \mathbb{R}^n . Assumimos que para todos $i, j \in \{1, 2, \dots, m\}$, com $i \neq j$, temos que $K_i \not\subseteq K_j$. Para um tal arranjo \mathcal{A} definimos

$$V_{\mathcal{A}} = \bigcup_{i=1}^m K_i \quad \text{e} \quad M_{\mathcal{A}} = \mathbb{R}^n \setminus V_{\mathcal{A}}.$$

Pomos $\widehat{V}_{\mathcal{A}}$ a compatificação usual de $V_{\mathcal{A}}$ com ω “no infinito”.

Para um arranjo \mathcal{A} definimos o *semirreticulado das intersecções* de \mathcal{A} , denotado por $L_{\mathcal{A}}$, como o conjunto parcialmente ordenado das intersecções não-vazias de elementos de \mathcal{A} ordenados por inclusão reversa, isto é, em $(L_{\mathcal{A}}, \leq)$ temos $A \leq B$ se e só se $B \subseteq A$. Note que $L_{\mathcal{A}}$ é um \wedge -semirreticulado, com o menor elemento $\widehat{0} = \mathbb{R}^n$. Se $\bigcap_i K_i \neq \emptyset$ então $L_{\mathcal{A}}$ é um reticulado com maior elemento $\widehat{1} = \bigcap_i K_i$. Seguindo o exemplo 2 da página 5, tome o arranjo de hiperplanos $\mathcal{A}_0 = \{H_1, H_2, H_3\}$. O reticulado das intersecções $L_{\mathcal{A}_0}$ do arranjo \mathcal{A}_0 é mostrado na figura 10.

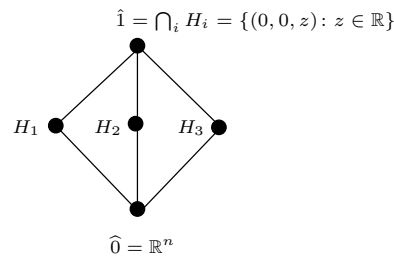


FIGURE 10. Diagrama de Hasse do reticulado $L_{\mathcal{A}_0}$

Um belo resultado que conecta alguns dos conceitos vistos é o seguinte. Se $x = K_{i_1} \cap \dots \cap K_{i_j}$ é um elemento de $L_{\mathcal{A}}$ então definimos $\dim(x) = \dim_{\text{afim}}(K_{i_1} \cap \dots \cap K_{i_j})$.

$\cdots \cap K_{i_j}$). Goresky e MacPherson [10] provaram o seguinte resultado usando Teoria de Morse (veja [17] para uma prova usando conceitos mais elementares).

Teorema 20 (Goresky, MacPherson '78, [10]). *Seja \mathcal{A} um arranjo de subespaços afins no \mathbb{R}^n . Então*

$$\tilde{H}^i(M_{\mathcal{A}}) \cong \bigoplus_{\substack{x > \hat{0} \\ x \in L_{\mathcal{A}}} \tilde{H}_{n-\dim(x)-2-i}(\Delta(\hat{0}, x)).$$

A vantagem de se expressar os números de Betti de $\hat{V}_{\mathcal{A}}$ e $M_{\mathcal{A}}$ em termos dos números de Betti de conjuntos parcialmente ordenados é que esses invariantes topológicos no primeiro caso são, em geral, difíceis de calcular, mas no segundo caso podemos usar a bem estabelecida teoria das funções de Möbius (veja [13]).

Corolário 21 (Björner, Lovász '94, [4]). *Se \mathcal{A} é um arranjo de subespaços afins no \mathbb{R}^n então*

$$\begin{aligned} \text{(i)} \quad \tilde{\beta}_i(\hat{V}_{\mathcal{A}}) &= \tilde{\beta}^{n-i-1}(M_{\mathcal{A}}) = \sum_{\substack{x > \hat{0} \\ x \in L_{\mathcal{A}}} \tilde{\beta}_{i-\dim(x)-1}(\Delta(\hat{0}, x)), \quad e \\ \text{(ii)} \quad \chi(\hat{V}_{\mathcal{A}}) - 1 &= (-1)^{n-1}(\chi(M_{\mathcal{A}}) - 1) = \sum_{\substack{x > \hat{0} \\ x \in L_{\mathcal{A}}} (-1)^{\dim(x)+1} \tilde{\chi}(\Delta(\hat{0}, x)). \end{aligned}$$

A prova de (i) é imediata do teorema de dualidade de Alexander (veja nota de rodapé na pág. 13) e da fórmula de Goresky e MacPherson.

A primeira igualdade de (ii) segue da definição de característica de Euler e do teorema de dualidade de Alexander:

$$\chi(\hat{V}_{\mathcal{A}}) - 1 = \sum_i (-1)^i \tilde{\beta}_i(\hat{V}_{\mathcal{A}}) = \sum_i (-1)^i \tilde{\beta}^{n-i-1}(M_{\mathcal{A}}) = (-1)^{n-1}(\chi(M_{\mathcal{A}}) - 1).$$

A segunda igualdade segue tomando a soma alternada em i na igualdade

$$\tilde{\beta}_i(\hat{V}_{\mathcal{A}}) = \sum_{\substack{x > \hat{0} \\ x \in L_{\mathcal{A}}} \tilde{\beta}_{i-\dim(x)-1}(\Delta(\hat{0}, x)).$$

Terminamos esta seção com o seguinte resultado, com cotas inferiores combinatorias para o problema de pertinência.

Teorema 22 (Björner e Lovász '94, [4]). *Se \mathcal{A} é um arranjo de subespaços lineares no \mathbb{R}^n e T uma árvore de decisão linear que testa pertinência em \mathcal{A} , então*

$$|L^s| \geq |\mu_{L_{\mathcal{A}}}(\hat{0}, \hat{1})| \quad e \quad |L^n| \geq |\mu_{L_{\mathcal{A}}}(\hat{0}, \hat{1})|.$$

Basta tomar os corolários 13 e 21(i) juntos com a equação (12).

Exemplo 23 (Estimativa para $\mu_{n,k}(\hat{0}, \hat{1})$). Consideremos o conjunto parcialmente ordenado $\Pi_{n,k}$ do exemplo 19. Convencionamos que $\mu_{n,k} = \mu_{\Pi_{n,k}}$. Convencionamos também que $\mu_k(n) = \mu_{n,k}(\hat{0}, \hat{1})$, e que $\mu_{n,k}(\sigma, \pi) = 0$ se algum de σ

ou π não pertence a $\Pi_{n,k}$. É conveniente estendermos a definição de $\mu_k(n)$ para quando $1 \leq n < k$ pondo $\mu_k(1) = 1$ e $\mu_k(2) = \mu_k(3) = \dots = \mu_k(k-1) = 0$.

A partir de uma recorrência para $\mu_{n,k}$ e usando técnicas de funções geradoras Björner, Lovász e Yao mostraram que

$$\exp\left(\sum_{n \geq 1} \mu_k(n) \frac{x^n}{n!}\right) = 1 + \sum_{n=1}^{k-1} \frac{x^n}{n!}.$$

Agora, se $\alpha_1, \alpha_2, \dots, \alpha_{k-1}$ são as raízes complexas de $1 + \sum_{n=1}^{k-1} x^n/n!$, então

$$(16) \quad \mu_k(n) = -(n-1)! \sum_{i=1}^{k-1} \alpha_i^{-n}.$$

De fato, vamos definir as funções geradoras exponenciais

$$F_k(x) = \sum_{n \geq 1} \mu_k(n) \frac{x^n}{n!} \quad \text{e} \quad p_k(x) = \sum_{i=0}^{k-1} \frac{x^i}{i!},$$

e considere $S_k(n, j)$ o número de partições do conjunto $\{1, 2, \dots, n\}$ em j partes de tamanho no máximo k .

É fato conhecido que

$$\sum_{n \geq 1} S_k(n, j) \frac{x^n}{n!} = \frac{1}{j!} (p_k(x) - 1)^j.$$

De um resultado de Crapo ([13], pág. 159, ex. 30) temos

$$\mu_k(n) = \sum_{j=1}^n (-1)^{j-1} (j-1)! S_k(n, j).$$

Substituindo em $F_k(x)$ temos que $F_k(x) = \ln p_k(x)$.

Agora, se α_i , $1 \leq i \leq k-1$, são as raízes (distintas) de $p_k(x)$ então $p_k(x) = \prod_i (1 - x/\alpha_i)$ e

$$F_k(x) = \sum_{i=1}^{k-1} \ln\left(1 - \frac{x}{\alpha_i}\right) = - \sum_{n \geq 1} \frac{x^n}{n} \sum_{i=1}^{k-1} \alpha_i^{-n}.$$

A comparação dos coeficientes de x^n resulta na equação (16).

Observamos que no caso $k = 3$ temos $p_3(x) = 1 + x + x^2/2$, que tem raízes $r_1 = -1+i$ e $r_2 = -1-i$. Portanto, $\mu_3(n) = -(n-1)!((-1+i)^{-n} + (-1-i)^{-n}) = -(n-1)!(-1-i)^{-n}(i^n + 1)$. Assim, $\mu_3(n) = 0$ se $n \equiv 2 \pmod{4}$. Para nossas cotas gostaríamos que $\mu_k(n)$ fosse grande mas, como acabamos de ver, isso nem sempre é verdade.

Entretanto, $\mu_k(n)$ é grande o suficiente em média. Para quaisquer n, k com $2 \leq k \leq n$ existe m tal que $n - k + 2 \leq m \leq n$ e

$$(17) \quad |\mu_k(m)| > (m-1)!(k-1)^{-m-1}.$$

Isso porque podemos supor que $|\alpha_1| < k - 1$ e se escrevemos

$$q(x) = \prod_{i=2}^{k-1} (x - \alpha_i) = \sum_{j=0}^{k-2} b_j x^j$$

então

$$q \left(\left(\frac{\mu_k(n-j)}{(n-j-1)!} \right)^{1/j} \right) = - \sum_{j=0}^{k-2} \sum_{i=1}^{k-1} b_j \alpha_i^{-n+j} = -q(\alpha_1) \alpha_1^{-n} = \alpha_1^{-n+k-1},$$

portanto, existe um j para o qual

$$\left| b_j \frac{\mu_k(n-j)}{(n-j-1)!} \right| > \frac{1}{k-1} |\alpha_1|^{-n+k-1}.$$

Por indução, podemos provar $b_j = (k-1)! \alpha_1^{-j-1} (p_k(\alpha_1) - p_{j+1}(\alpha_1))$ e isso implica que $|b_j| \leq (k-1)^{k-j} |\alpha_1|^{-1}$. Conseqüentemente,

$$|\mu_k(n-j)| > (n-j-1)! (k-1)^{-n+j-1}.$$

Esse limitante pode ser combinado com um argumento de monotonicidade do número de folhas em função da dimensão do espaço ambiente.

Defina $\ell^s(n, k)$ (resp., $\ell^n(n, k)$) como o número mínimo de folhas **sim** (resp., **não**) em qualquer árvore de decisão linear para o problema dos k -iguais no \mathbb{R}^n . Ambas as funções $\ell^n(n, k)$ e $\ell^s(n, k)$ são monótonas em n . De fato, tome $m \leq n$. Toda árvore de decisão linear T para n elementos pode ser usada para obtermos uma árvore de decisão linear para m elementos, adicionando-se a uma entrada de m elementos, $n - m$ novos elementos x, x^2, \dots, x^{n-m} , com $x > 1$ suficientemente grande.

Vamos ver se a árvore de decisão linear responde certo o problema dos k -iguais dentre m elementos. Se a resposta dada para uma certa entrada foi **não**, claramente a resposta está certa. No caso da resposta **sim**, se os coeficientes dos novos elementos forem zero nos testes da raiz até a folha, então a decisão foi tomada sobre os elementos velhos da entrada e, portanto, a árvore respondeu certo. Caso contrário, o sinal da função é trivialmente determinada devido a escolha do novos elementos e, portanto, a resposta também foi dada sobre decisões baseadas nos elementos velhos da entrada.

5. ALGUNS EXEMPLOS

Inclusão de conjuntos. Dados $A = \{x_1, x_2, \dots, x_n\}$ e $B = \{y_1, y_2, \dots, y_n\}$ decidir se $A \subseteq B$.

Uma árvore de computação que decide corretamente, também decidirá o caso particular de $B = \{1, 2, \dots, n\}$; nesse caso

$$S = \{(x_1, x_2, \dots, x_n) \in \mathbb{R}^n : \{x_1, x_2, \dots, x_n\} \subseteq \{1, 2, \dots, n\}\}$$

tem n^n componentes conexas, portanto, qualquer árvore de computação (ou de decisão com ordem fixa) tem altura $\Omega(n \log n)$.

Problema da mochila. Vimos no exemplo 5 que o problema da mochila é equivalente a decidir pertinência em um conjunto $S \subset \mathbb{R}^n$ tal que número de componentes conexas de $\mathbb{R}^n \setminus S$ é $2^{\binom{n}{2}+1}$, portanto, qualquer árvore de computação (ou de decisão com ordem fixa) tem altura $\Omega(n^2)$.

Conjuntos disjuntos. Dados $A = \{x_1, x_2, \dots, x_n\}$ e $B = \{y_1, y_2, \dots, y_n\}$ decidir se $A \cap B = \emptyset$.

Nesse caso

$$S = \{(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n) : \prod_{i,j} (x_i - y_j) \neq 0\}$$

tem $(n!)^2$ componentes conexas, portanto, qualquer árvore de computação (ou de decisão com ordem fixa) algébrica tem altura $\Omega(n \log n)$.

Para o número de componentes de S definimos

$$S_{\pi, \sigma} = \{(x_1, \dots, x_n, y_1, \dots, y_n) \in \mathbb{R}^{2n} : x_{\pi(1)} < y_{\sigma(1)} < x_{\pi(2)} < y_{\sigma(2)} < \dots < x_{\pi(n)}\},$$

onde π e σ são permutações, e temos $\bigcup_{\pi, \sigma} S_{\pi, \sigma} \subseteq S$. Definimos a função contínua $f_{i,j}(x_1, \dots, x_n, y_1, \dots, y_n) = x_i - y_j$ e notamos que se $z \in S_{\pi_1, \sigma_1}$ e $w \in S_{\pi_2, \sigma_2}$ então para qualquer caminho que ligue z a w devem existir coordenadas k e ℓ e um ponto x do caminho tal que $f_{k,\ell}(x) = 0$. Logo $x \notin S$ e podemos concluir que $\beta_0(S) \geq \beta_0\left(\bigcup_{\pi, \sigma} S_{\pi, \sigma}\right) \geq (n!)^2$.

Resultante. Dados $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n$, computar $\prod_{i,j} (x_i - y_j)$.

Qualquer algoritmo para computar esse produto com mais um teste para “ $\neq 0$?” torna-se um algoritmo para *conjuntos disjuntos*. Portanto, qualquer árvore de computação (ou de decisão com ordem fixa) algébrica tem altura $\Omega(n \log n)$.

Fecho convexo. Dados os pontos no plano $p_1, p_2, \dots, p_n \in \mathbb{R}^2$ devolver a lista ordenada no sentido anti-horário dos pontos que são os vértices do fecho convexo dos n pontos, denotada por $\text{FC}(p_1, \dots, p_n)$.

Um algoritmo para *fecho convexo* resolve o problema *ordenação*: dados os números reais x_1, x_2, \dots, x_n forme os pontos $p_i = (x_i, x_i^2)$ para todo $i \in [n]$. Esses pontos estão na parábola $y = x^2$ e, claramente, todos são vértices em $\text{FC}(p_1, \dots, p_n)$. O menor x_i pode ser computado com $O(n)$ comparações e, a partir desse ponto, os vértices do fecho no sentido anti-horário é uma ordenação das abscissas. Portanto, qualquer árvore para *fecho convexo* deve ter altura $\Omega(n \log n)$.

k -iguais. *Dados n números reais x_1, \dots, x_n e um inteiro $k \geq 2$ decidir se existem k deles iguais.*

O reticulado das intersecções de $\mathcal{A}_{n,k}$ (veja equação (5)), para $2 \leq k \leq n$, é isomorfo ao reticulado $\Pi_{n,k}$ do exemplo 19. Se, sob esse isomorfismo, um elemento $x \in L_{\mathcal{A}_{n,k}}$ corresponde a uma partição em $\Pi_{n,k}$ com j blocos, então $\dim(x) = j$.

Com os resultados que temos até agora podemos pôr o seguinte.

Corolário 24. *Se T é uma árvore de decisão linear para o problema dos k -iguais então*

- (i) $|L^s|, |L^n| \geq |\mu_{n,k}(\widehat{0}, \widehat{1})|$, e
- (ii) $|L^s|, |L^n| \geq \sum_i \tilde{\beta}_i(\Delta(\Pi_{n,k}))$.

Teorema 25 (Björner, Lovász '94, [4]). *Toda árvore de decisão linear para o problema dos k -iguais tem tamanho pelo menos $\max\{\binom{n}{k}, (n/3k)^n\}$, e altura pelo menos $\max\{n-1, n \log_3(n/3k)\}$.*

Esboço de prova. Seja T uma árvore de decisão linear para o problema dos k -iguais que não realiza testes desnecessários.

Para todo subconjunto $1 \leq i_1 < i_2 < \dots < i_k \leq n$ de índices, considere a entrada $x_{i_1} = x_{i_2} = \dots = x_{i_k} = 0$ e as outras coordenadas não-nulas e duas-a-duas distintas. Para dois desses subconjuntos de índices distintos devemos ter duas folhas **sim** diferentes, portanto, o número de folhas **sim** e conseqüentemente o tamanho da árvore é pelo menos $\binom{n}{k}$.

Notemos que $M_{\mathcal{A}_{n,k}}$ não pode conter um subespaço afim de dimensão k então, se considerarmos uma folha **não** de T , a célula correspondente não contém um subespaço afim k -dimensional, portanto, devemos ter pelo menos $n - k + 1$ nós no caminho que leva a tal folha. Seja w o pai, em T , de uma folha **não** que corresponde a uma n -célula. Alguma folha w' descendente de w deve ter rótulo **sim**, pois a árvore não faz testes desnecessários. Uma célula correspondente a uma folha **sim** está contida em $V_{\mathcal{A}_{n,k}}$, logo tem dimensão no máximo $n - k + 1$. Assim, como no raiz- w caminho em T nenhum teste resulta em igualdade (pois um filho de w tem dimensão n), no w - w' caminho em T devem existir pelo menos $k - 1$ testes resultando em igualdade. Esses testes mais os $n - k$ testes até w resulta em pelo menos $n - 1$ testes, isto é, a altura de T é pelo menos $n - 1$.

Para $k \geq n/3$, os máximos do enunciado do teorema ocorrem em $\binom{n}{k}$ e $n - 1$ e, portanto, ele está provado nesse caso. Vamos assumir $k < n/3$. Pela equação (17), escolhendo m com $n \geq m \geq n - k + 2$ temos

$$\ell^n(n, k) \geq \ell^n(m, k) \geq |\mu_k(m)| \geq (m-1)!(k-1)^{-m-1}.$$

Usando o fato de $v!(k-1)^{-v}$ ser não-decrescente para $v \geq k-1$, temos

$$\begin{aligned} \ell^n(n, k) &\geq (n-k+1)!(k-1)^{-n+k-3} \geq (n-k)!k^{-n+k-3} > \left(\frac{n-k}{3}\right)^{n-k} k^{-n+k-3} \\ &> \left(\frac{n-k}{k}\right)^{n-k} 3^{-n+k} k^{-3} > \left(\frac{n}{2k}\right)^{n/2} 3^{-n} = \left(\frac{n}{18k}\right)^{n/2} \end{aligned}$$

que, portanto, é um limitante inferior para o tamanho da árvore. O resultado na altura segue tomando-se o logaritmo. \square

O limitante obtido na demonstração é um pouco pior do que foi prometido no enunciado do teorema. Usando cota do item (ii) do corolário 24 junto com os números de Betti calculados por Björner e Welker, equação (15) para $t = \lfloor n/(2k) \rfloor$ e $i_0 = \dots = i_{t-1} = 0$, resultam no limitante prometido.

Yao [15] mostrou as cotas $\Omega(n \log(2n/k))$ e $\Omega(n \log(n/(k-1)) + (n-k) \log(n/(n-k+1)))$ sob árvore de computação e árvore de decisão algébricas de ordem fixa, respectivamente, para o problema dos k -iguais.

k -divisibilidade. *Dados n números reais x_1, \dots, x_n e um inteiro $k \geq 2$ decidir se o número de x_i 's iguais a qualquer número real é divisível por k .*

Aqui, supomos que k divide n e assumimos que $2 \leq k \leq n/2$. Denote por $\mathcal{D}_{n,k}$ o arranjo de todos os subespaços lineares B_π , onde π é uma partição de $\{1, 2, \dots, n\}$ em blocos de tamanho divisível por k . Então k -divisibilidade é exatamente o problema de pertinência em $V_{\mathcal{D}_{n,k}}$. O reticulado das intersecções desse arranjo tem a seguinte descrição combinatória:

$$\Pi_n^{(k)} = \{\pi \in \Pi_n : \text{todos os blocos de } \pi \text{ têm tamanho divisível por } k\} \cup \{\widehat{0}\}.$$

Definimos o *conjunto descendente* de uma permutação $\sigma \in S_n$ por

$$\{i \in [n-1] : \sigma(i) > \sigma(i+1)\}.$$

Stanley (veja [13, Prop. 4.3, pág. 353]) provou o seguinte resultado referente à função de Möbius $\mu_n^{(k)}$ do reticulado $\Pi_n^{(k)}$: $|\mu_n^{(k)}(\widehat{0}, \widehat{1})|$ é o número de permutações de $[n-1]$ com conjunto descendente $\{k, 2k, \dots, n-k\}$. Com esse resultado em mão temos o seguinte

Lema 26 (Björner, Lovász '94, [4]). *Se $t = n/k$ então*

$$|\mu_n^{(k)}(\widehat{0}, \widehat{1})| > \frac{(n-t)!(t-1)!}{((k-1)!)^t}.$$

Proof. Particione $\{t, \dots, n-1\}$ em t blocos A_1, A_2, \dots, A_t de tamanho $k-1$. Considere a permutação que começa com os elementos de A_1 em ordem crescente, seguido por qualquer elemento de $\{1, \dots, t-1\}$; então chegam os elementos de A_2 em ordem crescente, seguido por qualquer outro elemento de $\{1, \dots, t-1\}$; e assim por diante até terminarmos com os elementos de A_t em ordem crescente.

Claramente essa permutação tem o conjunto descendente certo. O número de permutações dessa forma é

$$\frac{(n-t)!(t-1)!}{((k-1)!)^t}.$$

□

Imediatamente tiramos a seguinte conclusão.

Corolário 27. *Se T é uma árvore de decisão linear para o problema da k -divisibilidade, então o número de folhas *sim* de T , bem como o número de folhas *não*, é pelo menos*

$$\frac{(n-t)!(t-1)!}{((k-1)!)^t},$$

onde $t = n/k$.

Usando a igualdade de Stirling temos um resultado mais explícito.

Teorema 28 (Björner, Lovász '94, [4]). *Toda árvore de decisão linear para o problema da k -divisibilidade tem tamanho pelo menos $(n/(k + 4 \log k))^n$ e altura pelo menos $n \log_3(n/(k + 4 \log k))$.*

5.1. Algoritmos baseados em comparações. Um fato interessante é que para quase todos esses problemas existe um algoritmo simples baseado em comparações dos elementos da entrada com complexidade da mesma ordem que a cota inferior dada pelas árvores de decisão.

REFERÊNCIAS

- [1] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in real algebraic geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, second edition, 2006.
- [2] Micahel Ben-Or. Lower bounds for algebraic computation trees. In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing (STOC 83, Boston, MA)*, pages 80–86, New York, 1983. ACM.
- [3] A. Björner, L. Lovász, and A.C.-C. Yao. Linear decision trees: volume estimates and topological bounds. In *24th Annual ACM Symposium on the Theory of Computing (STOC 92, Victoria, BC, Canada)*, pages 170–177, 1992.
- [4] Anders Björner and László Lovász. Linear decision trees, subspace arrangements and Möbius functions. *J. Amer. Math. Soc.*, 7(3):677–706, 1994.
- [5] Anders Björner and Michelle L. Wachs. Shellable nonpure complexes and posets. I. *Trans. Amer. Math. Soc.*, 348(4):1299–1327, 1996.
- [6] Anders Björner and Volkmar Welker. The homology of “ k -equal” manifolds and related partition lattices. *Adv. Math.*, 110(2):277–313, 1995.
- [7] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to algorithms*. The MIT Electrical Engineering and Computer Science Series. MIT Press, Cambridge, MA, 1990.
- [8] David Dobkin. A nonlinear lower bound on linear search tree programs for solving knapsack problems. *J. Comput. System Sci.*, 13(1):69–73, 1976.

- [9] David P. Dobkin and Richard J. Lipton. On the complexity of computations under varying sets of primitives. In *Automata theory and formal languages (Second GI Conf., Kaiserslautern, 1975)*, pages 110–117. Lecture Notes in Comput. Sci., Vol. 33. Springer, Berlin, 1975.
- [10] Mark Goresky and Robert MacPherson. *Stratified Morse theory*, volume 14 of *Ergebnisse der Mathematik und ihrer Grenzgebiete (3) [Results in Mathematics and Related Areas (3)]*. Springer-Verlag, Berlin, 1988.
- [11] Allen Hatcher. *Algebraic topology*. Cambridge University Press, Cambridge, 2002. Disponível em <http://www.math.cornell.edu/hatcher/AT/ATpage.html>.
- [12] J. Milnor. On the Betti numbers of real varieties. *Proc. Amer. Math. Soc.*, 15:275–280, 1964.
- [13] Richard P. Stanley. *Enumerative combinatorics. Vol. 1*, volume 49 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 1997. With a foreword by Gian-Carlo Rota, Corrected reprint of the 1986 original.
- [14] J. Michael Steele and Andrew C. Yao. Lower bounds for algebraic decision trees. *J. Algorithms*, 3(1):1–8, 1982.
- [15] Andrew Chi-Chih Yao. Algebraic decision trees and Euler characteristics. *Theoret. Comput. Sci.*, 141(1-2):133–150, 1995.
- [16] Andrew Chi-Chih Yao. Decision tree complexity and Betti numbers. *J. Comput. System Sci.*, 55(1, part 1):36–43, 1997. 26th Annual ACM Symposium on the Theory of Computing (STOC '94) (Montreal, PQ, 1994).
- [17] Günter M. Ziegler and Rade T. Živaljević. Homotopy types of subspace arrangements via diagrams of spaces. *Math. Ann.*, 295(3):527–548, 1993.