

# Limite superior algorítmico para a contagem de intervalo

Lívia Salgado Medeiros<sup>1</sup>, Fabiano de Souza Oliveira<sup>1</sup>, Jayme Luiz Szwarcfiter<sup>1,2</sup>

<sup>1</sup>Instituto de Matemática e Estatística – Universidade do Estado do Rio de Janeiro (UERJ)  
CEP 20550-900 – Rio de Janeiro – RJ – Brazil

<sup>2</sup>COPPE, NCE, IM – Universidade Federal do Rio de Janeiro (UFRJ)  
CEP 21941-600 – Rio de Janeiro – RJ – Brazil

liviasmedeiros@gmail.com, fabiano.oliveira@ime.uerj.br, jayme@nce.ufrj.br

**Abstract.** *The employment of combinatorial optimization techniques associated with the interval count problem was explored by [Joos et al. 2014], efficiently solving a restricted version of the interval count 2 problem. For the problem of determining the general interval count, no similar technique is currently known. We devise quadratic programming formulations, leading to a pseudopolynomial algorithm, for providing an upper bound on the general interval count of orders, which is empirically shown to be close to the respective actual values. For the class of semiorders, the upper bound is shown to be tight.*

## 1. Introdução

O grafo de intervalo de uma família  $\mathcal{R}$  de intervalos fechados na reta real é o grafo  $G$  tal que  $V(G) = \mathcal{R}$  e, para todo  $I, J \in V(G)$ ,  $(I, J) \in E(G)$  se e somente se  $I \cap J \neq \emptyset$ . Chamamos  $\mathcal{R}$  de um modelo de  $G$ . Uma ordem  $P = (X, \prec)$  é uma relação binária  $\prec$  no conjunto  $X$  que é irreflexiva e transitiva. Além disso,  $P$  é uma ordem de intervalo se existe um modelo de intervalo  $\mathcal{R} = \{[\ell_x, r_x] \mid x \in X\}$  tal que  $x \prec y$  se e somente se  $r_x < \ell_y$ . O menor número  $IC(G)$  (resp.  $IC(P)$ ) de tamanhos distintos de intervalos necessário em um modelo de  $G$  (resp.  $P$ ) é chamado contagem de intervalo de  $G$  (resp.  $P$ ). O problema da contagem de intervalo foi inicialmente sugerido por Ronald Graham na década de 80 e, até o momento, a complexidade de reconhecer se  $IC(G) \leq k$  ( $IC(P) \leq k$ ) para todo  $k \geq 2$  é um problema em aberto [Fishburn 1985, Klavík et al. 2019].

Em [Joos et al. 2014] foi desenvolvido um algoritmo polinomial, usando programação linear, para determinar se, dados um grafo conexo  $G$  e uma bipartição dos seus vértices em dois conjuntos  $A$  e  $B$ , existe um modelo  $\{I_v \mid v \in V(G)\}$  de  $G$  tal que  $|I_v| = a$  para todo  $v \in A$  e  $|I_v| = b$  para todo  $v \in B$ , para algum par de constantes  $a < b$ . Os autores consideraram portanto o problema da contagem de intervalo restrito às classes de contagem igual a dois. Além deste trabalho, nenhuma outra técnica de otimização combinatória para o problema é conhecida, em especial para o problema geral. Neste trabalho, apresentamos formulações de programação quadrática que podem ser resolvidas em tempo pseudopolinomial fornecendo um limite superior para a contagem de intervalo de uma dada ordem de intervalo. Apesar de não haver garantias sobre o quão justo é o limite, os experimentos com exemplos da literatura que são considerados difíceis para o problema tiveram bons resultados na prática. Na Seção 2, apresentamos o algoritmo que fornece o limite superior baseado em formulações de programação quadrática. Na Seção 3, apresentamos os experimentos realizados e as observações finais na Seção 4.

---

O segundo e terceiro autores são parcialmente financiados por FAPERJ e CNPq.

## 2. O algoritmo para o limite superior

Um problema com formulação de programação quadrática tem por objetivo encontrar um vetor  $x$   $n$ -dimensional que satisfaça um conjunto de restrições enquanto minimiza certa função objetivo, com formulação geral

$$\begin{array}{ll} \text{minimizar} & f(x) = \frac{1}{2}x^T Fx + q^T \\ \text{sujeito a} & Ax = b \\ & Qx \leq h \end{array} \quad (1)$$

onde  $F$ ,  $A$ ,  $Q$  são matrizes e  $q$ ,  $b$ ,  $h$  são vetores constantes, todos de dimensões apropriadas. Uma matriz  $M$  de dimensão  $n$  é *semidefinida positiva* se  $x^T Mx \geq 0$  para todo  $0 \neq x \in \mathbb{R}^n$ . Quando a matriz  $F$  de (1) é semidefinida positiva, o método solucionador elipsóide resolve a programação quadrática em tempo fracamente polinomial [Kozlov et al. 1981].

Na aplicação ao problema de contagem de intervalo, o vetor variável  $x$  representa um modelo de intervalo, consistindo de

$$x = [s_1, s_2, \dots, s_n, \ell_1, r_1, \ell_2, r_2, \dots, \ell_n, r_n, \epsilon]$$

onde  $\ell_i$ ,  $r_i$ ,  $s_i$  representam respectivamente a extremidade esquerda, a extremidade direita e o tamanho do  $i$ -ésimo intervalo deste modelo e  $\epsilon$  o mínimo distanciamento entre dois intervalos que não se intersectam. Consideramos duas propostas de função objetivo, nomeadamente as funções  $f_1(x)$  e  $f_2(x)$  definidas como

$$f_1(x) = \sum_{i=1}^n (s_i - 1)^2 \quad f_2(x) = \sum_{i=1}^n \sum_{j=i+1}^n (s_i - s_j)^2 + \sum_{i=1}^n s_i \quad (2)$$

O conjunto de restrições faz com que uma solução viável  $x$  represente um modelo:

$$\left\{ \begin{array}{ll} r_i - \ell_i = s_i & \text{para todo } i = 1, 2, \dots, n \\ r_u \geq \ell_v \text{ e } r_v \geq \ell_u & \text{para todo } u, v \in X, u \not\prec v \text{ e } v \not\prec u \\ r_u + \epsilon \leq \ell_v & \text{para todo } u, v \in X \text{ e } u \prec v \\ \ell_i \geq 0, r_i \geq 0 \text{ e } s_i \geq 1 & \text{para todo } i = 1, 2, \dots, n \\ \epsilon = 0,01 & \end{array} \right. \quad (3)$$

Para executar a otimização com as formulações definidas acima, utilizamos o CVXPY que é uma biblioteca de domínio específico para otimização convexa e pode ser importada para Python.

O algoritmo recebe como entrada um modelo  $\mathcal{R}$  de uma ordem  $P$  e define as matrizes  $A$ ,  $Q$  e vetores  $b$ ,  $h$  que correspondem, em formato matricial, às restrições definidas em (3). Em seguida, define a matriz  $F$  e o vetor  $q$  correspondente a  $f_1(x)$  ou  $f_2(x)$  em (2), e executa o pacote CVXPY com o solucionador ECOS, obtendo o resultado da otimização, que corresponde ao modelo  $\mathcal{R}'$ . Em seguida, o algoritmo obtém o modelo  $\mathcal{R}$  a partir de  $\mathcal{R}'$  pelo aumento de todos os intervalos que se encontram na primeira e última cliques de  $\mathcal{R}'$  (cliques mais à esquerda e mais à direita, respectivamente) até que os tamanhos destes intervalos sejam iguais ao maior presente em  $\mathcal{R}'$ . Por fim, o algoritmo retorna como contagem de intervalo de  $P$  o número de tamanhos distintos de  $\mathcal{R}$ . Naturalmente,  $IC(P)$  é no máximo o número de tamanhos presentes em  $\mathcal{R}$ . Portanto, o retorno do algoritmo é de fato um limite superior para  $IC(P)$ .

**Lema 1.** A matriz  $F$  associada à função  $f_1(x)$  ou  $f_2(x)$  em (2) é semidefinida positiva.

### 3. Experimentos

O objetivo é comparar os resultados de cada função objetivo, que fornece um limite superior para a contagem de intervalo, aos valores reais. O limite superior fornecido pelo algoritmo é justo para ordens de intervalo unitário (semiordens), de acordo com o Lema 2.

**Lema 2.** *Seja  $k^*$  o resultado obtido pelo algoritmo com respeito a alguma das funções objetivos formuladas em (2) para dada semiordem  $P$ . Então  $k^* = 1 = IC(P)$ .*

Descreveremos algumas classes de grafos e ordens com instâncias usadas nos experimentos. Todas as instâncias usadas nos experimentos se relacionam a ordens.

**Classe 1.** A Figura 1(a) ilustra um grafo  $G$  com  $IC(G) = 2$  para o qual um vértice ( $c$ ) que não é centro de  $K_{1,3}$  não pode possuir o menor tamanho. Isto contraria a intuição ingênua que o problema de contagem de intervalo 2 se resume a atribuir o maior tamanho exclusivamente àqueles vértices que são centro de  $K_{1,3}$ .



Figura 1. Ordens definidas por seus modelos.

**Classe 2.** Leibowich estudou a classe de grafos quase livres de  $K_{1,3}$ , definidos como aqueles que possuem um vértice cuja remoção torna-o livre de  $K_{1,3}$  e provou que  $IC(G) \leq 2$  para grafos  $G$  nesta classe [Leibowitz et al. 1982]. A Figura 1(b) é um exemplo de um grafo nesta classe.

**Classe 3.** Fishburn introduziu o problema extremal de determinar

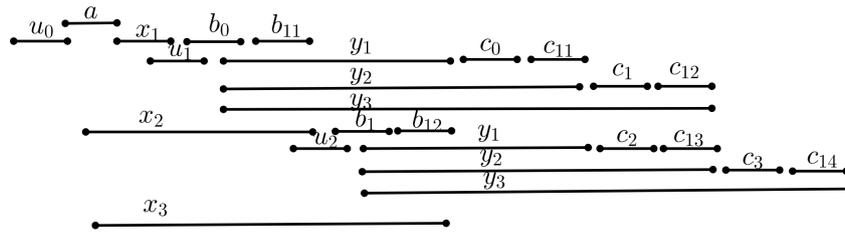
$$\sigma_{\mathcal{C}}(k) = \min\{|X| \mid P = (X, \prec) \text{ is in class } \mathcal{C} \text{ and } IC(P) \geq k\}$$

e  $\tilde{\sigma}_{\mathcal{C}}(k)$  que é definido analogamente para grafos [Fishburn 1985]. Fishburn conjecturou que  $\sigma_{\mathcal{C}}(k) = 3k - 2$  para  $\mathcal{C}$  como a classe geral de ordens de intervalo, tendo provado a conjectura para todo  $k \leq 7$ . No entanto, nenhum valor de  $\tilde{\sigma}_{\mathcal{C}}(k)$  é conhecido. Um grafo de intervalo  $G$  é *trivialmente perfeito* (na classe TP) se  $G$  é livre de  $P_4$ . Um grafo de intervalo  $G$  é um grafo *split* (na classe SPLIT) se existe uma partição  $(K, I)$  de  $V(G)$  na qual  $K$  é uma clique e  $I$  é um conjunto independente. Estudamos  $\sigma_{\mathcal{C}}(k)$  e  $\tilde{\sigma}_{\mathcal{C}}(k)$  para  $\mathcal{C} \in \{TP, SPLIT\}$  [Medeiros et al. 2019]. Entre outros resultados, provamos que  $\sigma_{TP}(k) = 3k - 2$ ,  $\sigma_{SPLIT}(k) = 3k - 2$ ,  $\tilde{\sigma}_{TP}(k) = \frac{(3^k - 1)}{2}$ ,  $\tilde{\sigma}_{SPLIT}(k) = 3k - 1$ .

**Classe 4.** Fishburn define a família de ordens  $\mathcal{D}_2$ , cada uma com cadeia de inclusão própria entre intervalos limitada a 2 [Fishburn 1985], provando que esta família requer um número de tamanhos distintos arbitrariamente grande, estabelecendo o seguinte teorema.

**Teorema 1.** ([Fishburn 1985]) *Para todo natural  $k \geq 1$ , existe uma ordem de intervalo  $P_k \in \mathcal{D}_2$  tal que  $IC(P) \geq k + 1$ .*

A Tabela 1 sumariza os resultados dos experimentos. Os resultados descritos aqui são simplificados devido a limitações de espaço. Ao todo, foram testadas 54 ordens. Nos resultados obtidos, o algoritmo com função objetivo  $f_1(x)$  reportou 92% dos limites



**Figura 2. Modelo de  $P_2$  da família  $\mathcal{D}_2$  (Classe 4).**

superiores que coincidem com a contagem de intervalo real e 8% dos limites superiores maiores que o valor da contagem de intervalo real. A média da diferença entre o valor reportado e o valor real foi de 1,5 com o uso de  $f_1(x)$ . O algoritmo com função objetivo  $f_2(x)$  reportou 28% dos limites superiores que coincidem com a contagem de intervalo real, 48% das ordens testadas com o uso de  $f_2(x)$  não foram computados por nenhum solucionador e 24% dos limites superiores maiores que o valor da contagem de intervalo real. A média da diferença do valor reportado e o real foi de 1,2 com o uso de  $f_2(x)$ .

**Tabela 1. Resultados parciais sumarizados.**

Classe de $P$	$n$	$IC(P)$	Limite Superior	
			$f_1(x)$	$f_2(x)$
1	13	2	3	4
2	9	2	2	2
3 (TP)	7	3	3	3
3 (TP)	43	15	15	15
3 (TP)	40	4	4	4
3 (TP)	121	5	5	6

Classe de $P$	$n$	$IC(P)$	Limite Superior	
			$f_1(x)$	$f_2(x)$
3 (SPLIT)	14	5	5	6
3 (SPLIT)	74	25	25	26
3 (SPLIT)	75	25	25	25
3 (SPLIT)	120	40	40	40
4	25	$\geq 3$	7	8

#### 4. Conclusão

Elaboramos um limite superior algorítmico para a contagem de intervalo de ordens. Observa-se que, em geral, os valores encontrados pela função objetivo  $f_1(x)$  aproximam-se dos resultados reais. Como trabalho futuro, investigaremos outras funções objetivo e se o limite superior aqui proposto é justo para classes restritas de ordens de intervalo.

#### Referências

- Fishburn, P. (1985). *Interval Orders and Interval Graphs*. John Wiley and Sons.
- Joos, F., Lowenstein, C., Oliveira, F. S., Rautenbach, D., and Szwarcfiter, J. L. (2014). Graphs of interval count two with a given partition. *Inf. Process. Lett.*, 114:542–546.
- Klavík, P., Otachi, Y., and Sejnoha, J. (2019). On the classes of interval graphs of limited nesting and count of lengths. *Algorithmica*, 81:1490–1511.
- Kozlov, M. K., Tarasov, S. P., and Khachiyan, L. G. (1981). The polynomial solvability of convex quadratic programming. *Comput. Maths. Math. Phys.*, 20(5):223–228.
- Leibowitz, R., Assmann, S. F., and Peck, G. W. (1982). The interval count of a graph. *SIAM J. Alg. Discr. Meth.*, 3(4):485–494.
- Medeiros, L. S., Oliveira, F. S., and Szwarcfiter, J. L. (2019). Two problems on interval counting. *Electron. Notes Theor. Comput. Sci.*, 346:625–643.