Resumo

Título: Como decidir melhor em 10^{156} passos fáceis.
Estamos sobrecarregados com escolhas. Nas tarefas mais cotidianas, como montar uma refeição ou no planejamento de um roteiro de viagens, precisamos definir critérios para tomarmos boas decisões. Neste seminário, conversaremos sobre ferramentas matemáticas que podem nos ajudar a tomar a melhor decisão possível (!) em problemas que podem ter um número astronômico de escolhas.

Como decidir melhor em 10^{156} passos fáceis

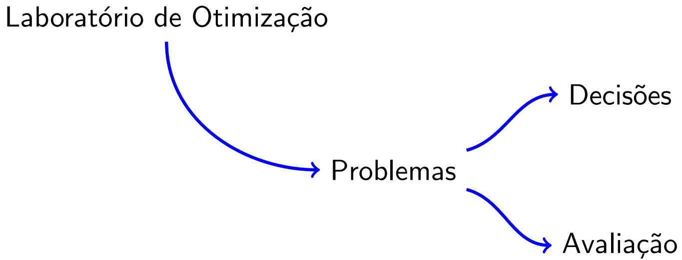
Pedro Belin Castellucci

@pbcastellucci
lot.icmc.usp.br

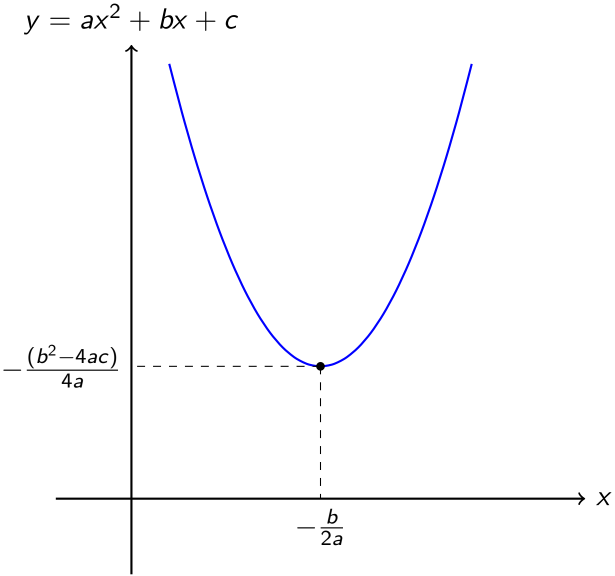
Laboratório de Otimização

Laboratório de Otimização

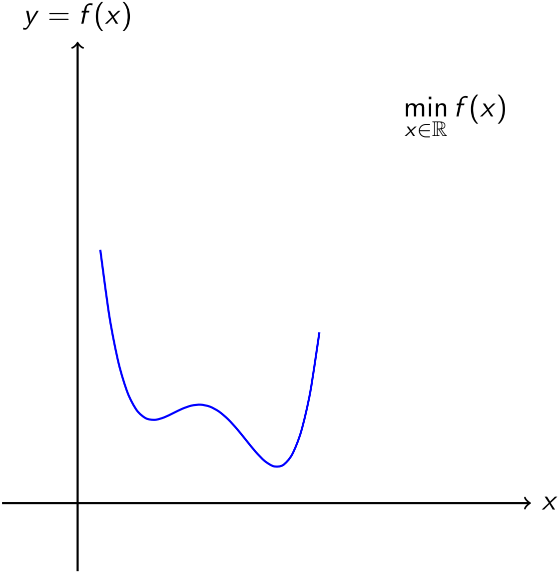
Problemas


Laboratório de Otimização
Problemas
Decisões
Avaliação

Minimizando

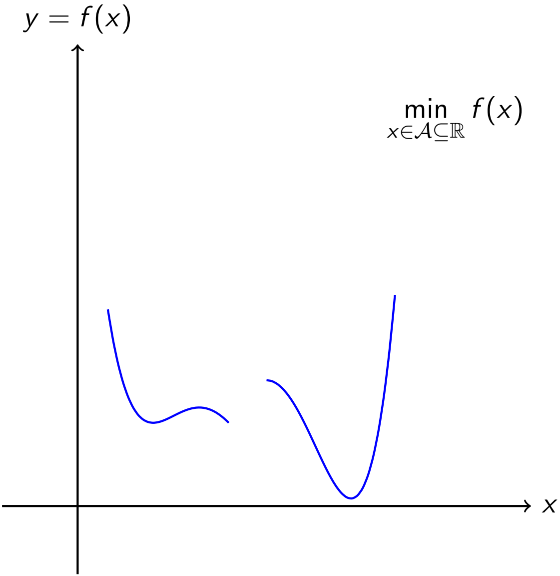


Minimizando



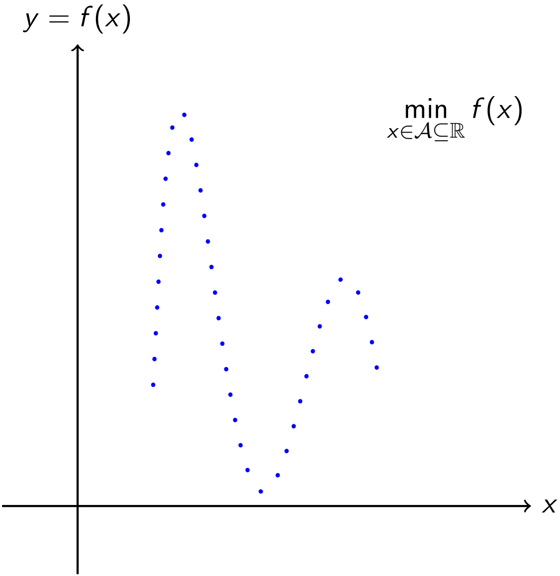
$$\min_{x \in \mathbb{R}} f(x)$$

Minimizando

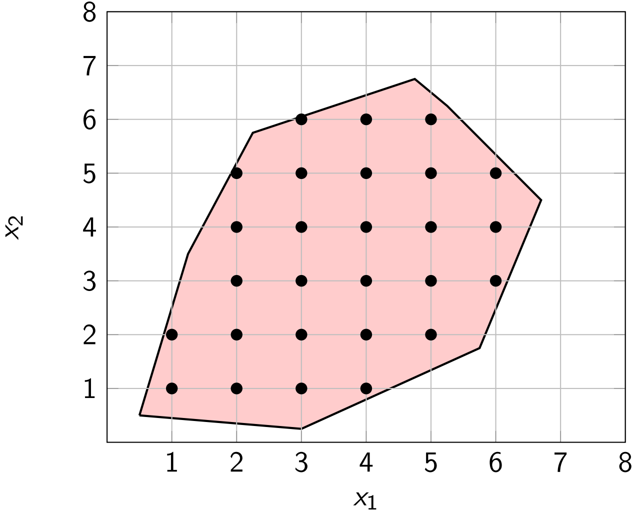


$$\min_{x \in \mathcal{A} \subseteq \mathbb{R}} f(x)$$

Minimizando



$$
\begin{aligned}
&\text{Opt } c_1x_1 + c_2x_2 + \ldots + c_nx_n,\\
&a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n \le b_1,\\
&a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n \le b_2,\\
&\vdots\\
&a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \le b_m,\\
&x_1, x_2, \ldots, x_n \in \mathbb{Z}_+.
\end{aligned}
$$



8
7
6
5
4
3
2
1
x_2
1
2
3
4
5
6
7
8
x_1

Problema da mochila

Item (i)	1	2	3	4	5
Valor (v_i)	9	10	14	3	4
Peso (w_i)	3	4	7	2	5
C	13				

Problema da mochila

Item (i)	1	2	3	4	5
Valor (v_i)	9	10	14	3	4
Peso (w_i)	3	4	7	2	5
C	13				

x_i : 1 se o item i é escolhido, 0 caso contrário.

Problema da mochila

Item (i)	1	2	3	4	5
Valor (v_i)	9	10	14	3	4
Peso (w_i)	3	4	7	2	5
C	13				

x_i : 1 se o item i é escolhido, 0 caso contrário.

Max $\quad 9x_1 + 10x_2 + 14x_3 + 3x_4 + 4x_5$

sujeito a:

$$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13$$

$$x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}$$

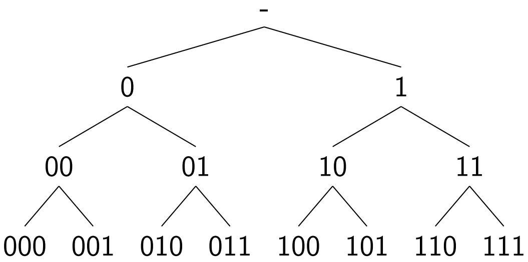
Problema da mochila

Item (i)	1	2	3	4	5
Valor (v_i)	9	10	14	3	4
Peso (w_i)	3	4	7	2	5
C	13				

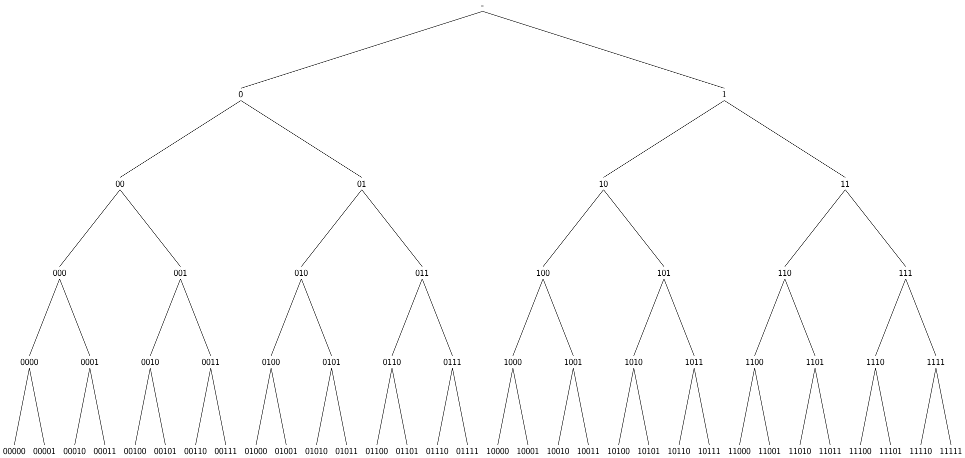
x_i : 1 se o item i é escolhido, 0 caso contrário.

$$
\begin{aligned}
\text{Max} \quad & 9x_1 + 10x_2 + 14x_3 + 3x_4 + 4x_5 \\
\text{sujeito a:} \quad & \\
& 3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \\
& x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{aligned}
$$

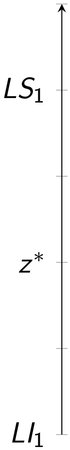
$$
\begin{aligned}
\text{Max} \quad & \sum_{i \in I} v_i x_i \\
\text{sujeito a:} \quad & \\
& \sum_{i \in I} w_i x_i \leq C \\
& x_i \in \{0, 1\}, i \in I.
\end{aligned}
$$


-
0
1
00
01
10
11
000
001
010
011
100
101
110
111

Enumeração explícita

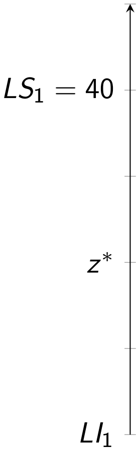


Limitantes



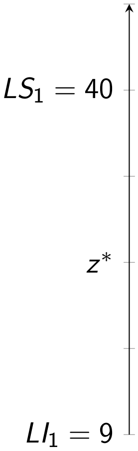
$$
\begin{aligned}
\text{Max} \quad & 9x_1 + 10x_2 + 14x_3 + 3x_4 + 4x_5 \\
\text{sujeito a:} \quad & \\
& 3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \\
& x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{aligned}
$$

Limitantes



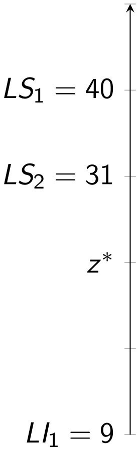
$$
\begin{aligned}
\text{Max} \quad & 9x_1 + 10x_2 + 14x_3 + 3x_4 + 4x_5 \\
\text{sujeito a:} \quad & \\
& 3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \\
& x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{aligned}
$$

Limitantes



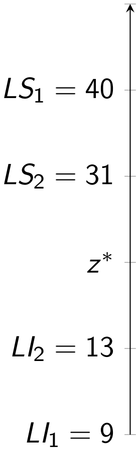
$$
\begin{aligned}
\text{Max} \quad & 9x_1 + 10x_2 + 14x_3 + 3x_4 + 4x_5 \\
\text{sujeito a:} \quad & \\
& 3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \\
& x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{aligned}
$$

Limitantes



$$
\begin{aligned}
\text{Max} \quad & 9x_1 + 10x_2 + 14x_3 + 3x_4 + 4x_5 \\
\text{sujeito a:} \quad & \\
& 3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \\
& x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{aligned}
$$

Limitantes



$$\begin{aligned}
\text{Max} \quad & 9x_1 + 10x_2 + 14x_3 + 3x_4 + 4x_5 \\
\text{sujeito a:} \quad & \\
& 3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \\
& x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{aligned}$$

ECONOMETRICA

VOLUME 28 July, 1960 NUMBER 3

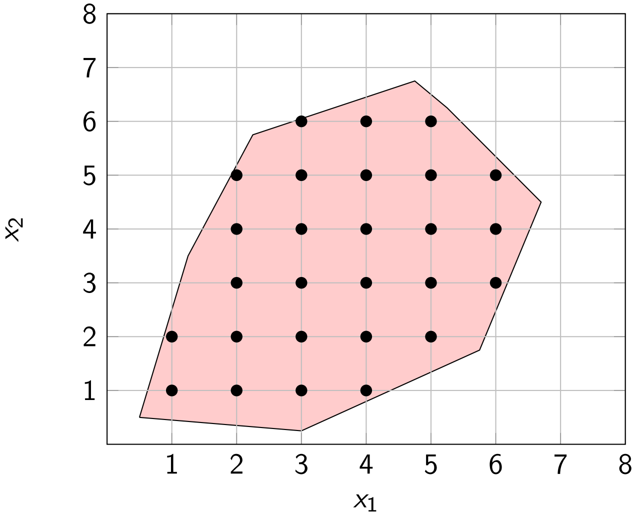
AN AUTOMATIC METHOD OF SOLVING DISCRETE PROGRAMMING PROBLEMS

BY A. H. LAND AND A. G. DOIG

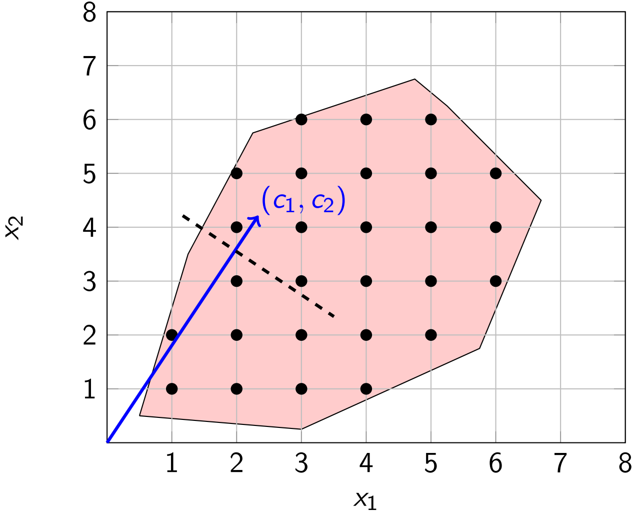
In the classical linear programming problem the behaviour of continuous, nonnegative variables subject to a system of linear inequalities is investigated. One possible generalization of this problem is to relax the continuity condition on the variables. This paper presents a simple numerical algorithm for the solution of programming problems in which some or all of the variables can take only discrete values. The algorithm requires no special techniques beyond those used in ordinary linear programming, and lends itself to automatic computing. Its use is illustrated on two numerical examples.

$$
\begin{aligned}
&\text{Opt } c_1x_1 + c_2x_2 + \ldots + c_nx_n,\\
&a_{11}x_1 + a_{12}x_2 + \ldots + a_{13}x_3 \leq b_1,\\
&a_{21}x_1 + a_{22}x_2 + \ldots + a_{23}x_3 \leq b_2,\\
&\vdots\\
&a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{m3}x_3 \leq b_m,\\
&x_1, x_2, \ldots, x_n \in \mathbb{Z}_+.
\end{aligned}
$$

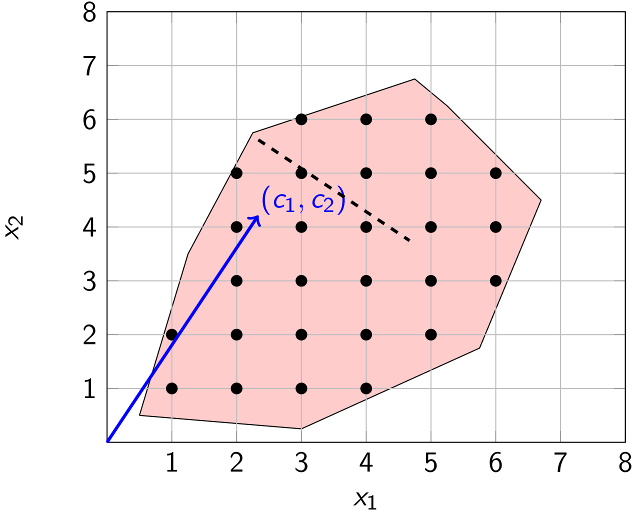
$$
\begin{aligned}
&\text{Opt } c_1x_1 + c_2x_2 + \ldots + c_nx_n,\\
&a_{11}x_1 + a_{12}x_2 + \ldots + a_{13}x_3 \leq b_1,\\
&a_{21}x_1 + a_{22}x_2 + \ldots + a_{23}x_3 \leq b_2,\\
&\vdots\\
&a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{m3}x_3 \leq b_m,\\
&\boldsymbol{x_1, x_2, \ldots, x_n \in \mathbb{R}_+}.
\end{aligned}
$$



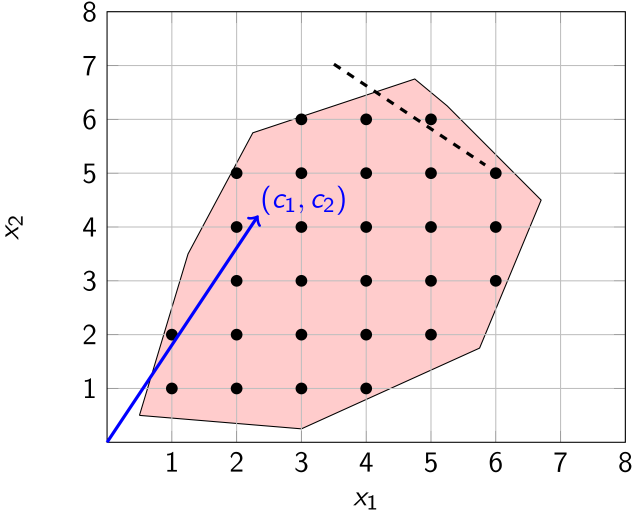
8
7
6
5
4
3
2
1
x_2
1 2 3 4 5 6 7 8
x_1



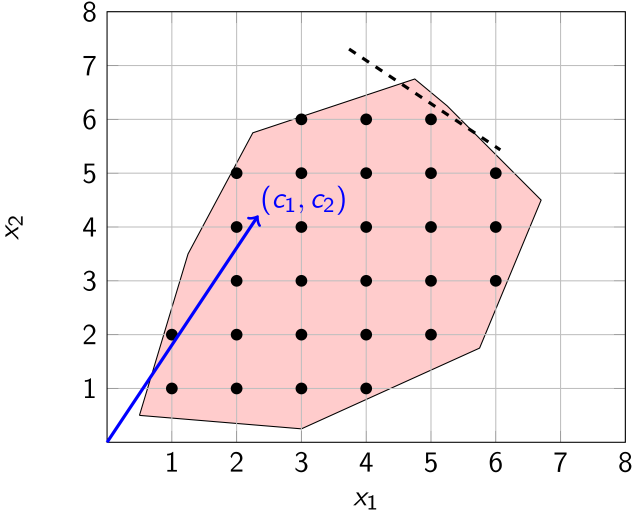
(c_1, c_2)
8
7
6
5
4
3
2
1
1
2
3
4
5
6
7
8
x_1
x_2



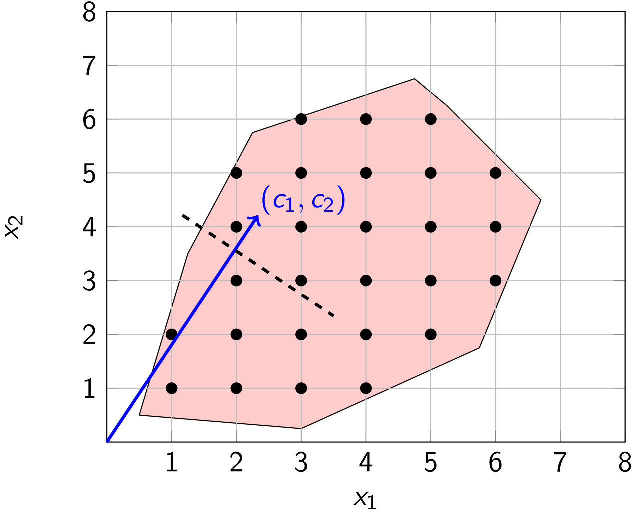
8
7
6
5
4
3
2
1
(c_1, c_2)
1
2
3
4
5
6
7
8
x_1
x_2



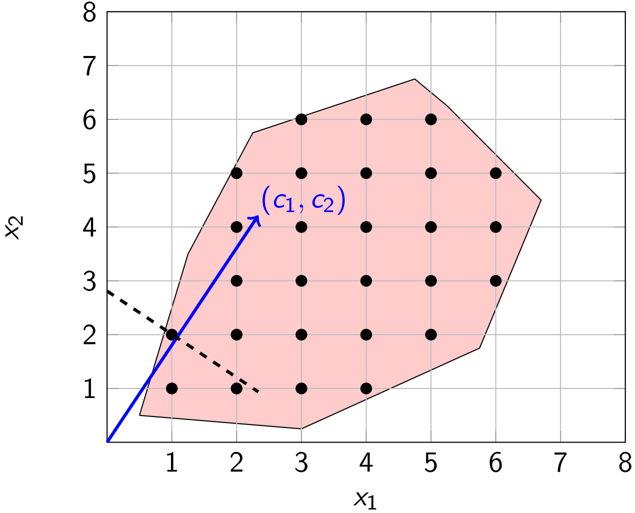
8
7
6
5
4
3
2
1
x_2
(c_1, c_2)
1 2 3 4 5 6 7 8
x_1



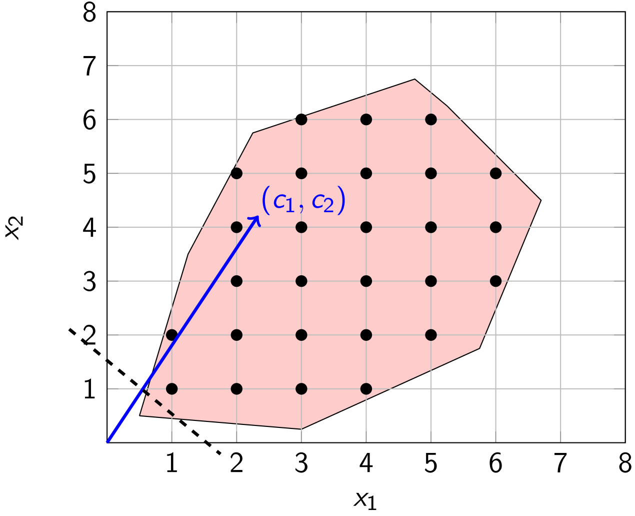
(c_1, c_2)
x_1
x_2
1
2
3
4
5
6
7
8



8
7
6
5
4
3
2
1
x_2
(c_1, c_2)
1 2 3 4 5 6 7 8
x_1



8
7
6
5
4
3
2
1
x_2
(c_1, c_2)
1
2
3
4
5
6
7
8
x_1



(c_1, c_2)
x_1
x_2
1
2
3
4
5
6
7
8

Problema da mochila

$$\text{Max} \quad 9x_1 + 10x_2 + 14x_3 + 3x_4 + 5x_5 \tag{1}$$

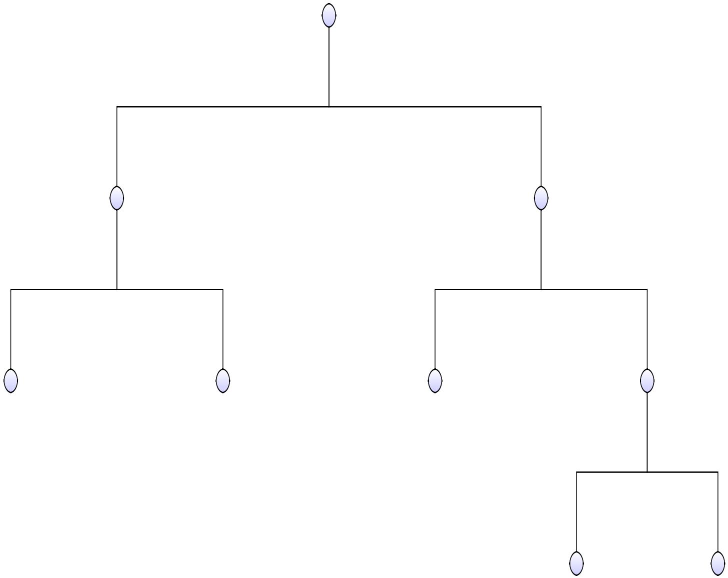
sujeito a:

$$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \tag{2}$$

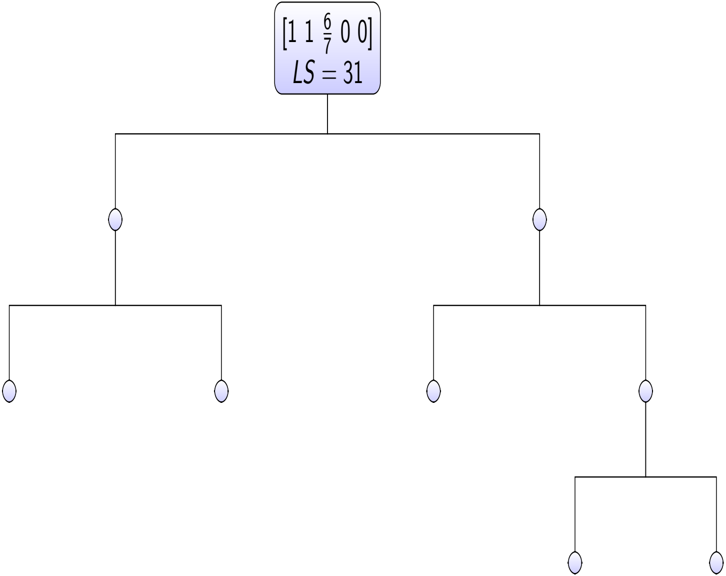
$$x_1, x_2, x_3, x_4, x_5 \in \{0, 1\} \tag{3}$$

$z = \text{Max } \{9x_1 + 10x_2 + 14x_3 + 3x_4 + 5x_5 :$

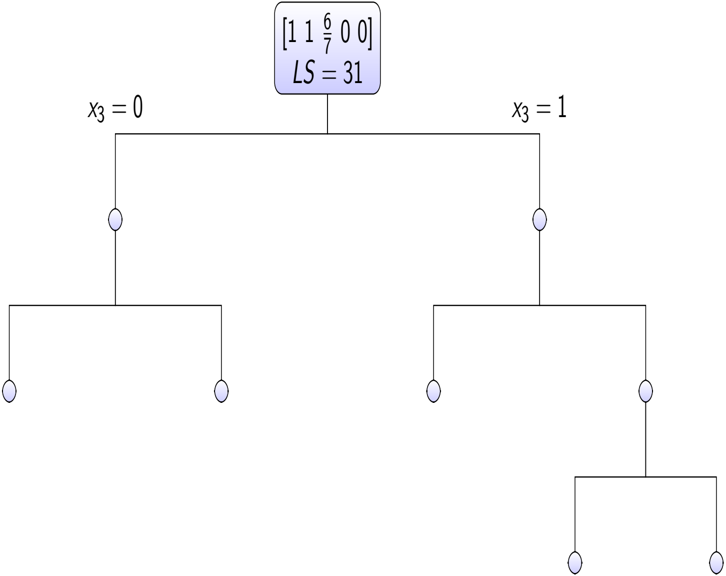
$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \ e \ x_1, \ldots, x_5 \in \{0, 1\}\}$



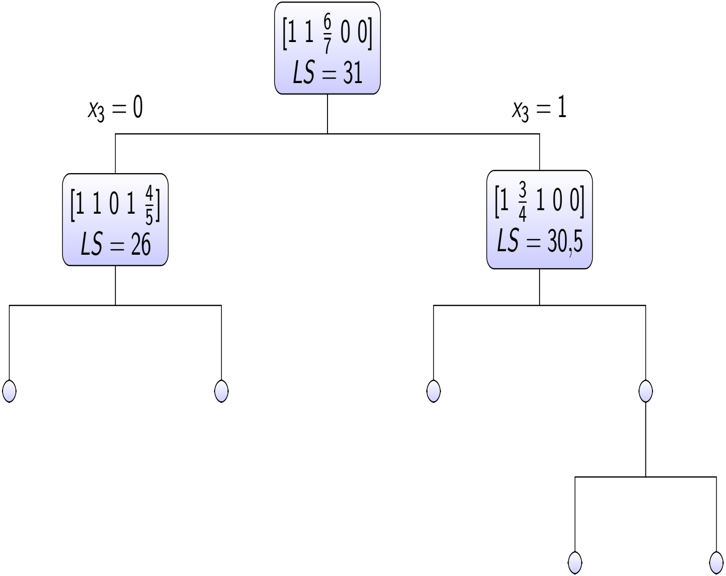
$$z = \text{Max } \{9x_1 + 10x_2 + 14x_3 + 3x_4 + 5x_5 :$$
$$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \text{ } e \text{ } x_1, \ldots, x_5 \in \{0, 1\}\}$$



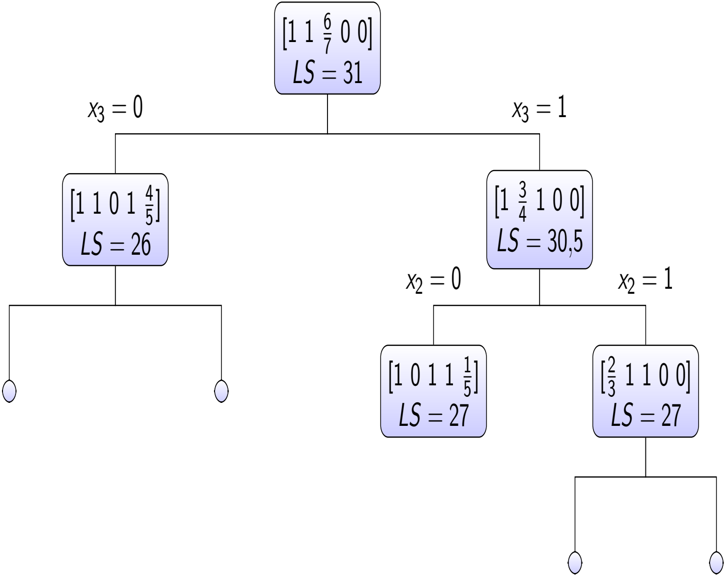
$$z = \text{Max } \{9x_1 + 10x_2 + 14x_3 + 3x_4 + 5x_5 :$$
$$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \ e \ x_1, \ldots, x_5 \in \{0, 1\}\}$$



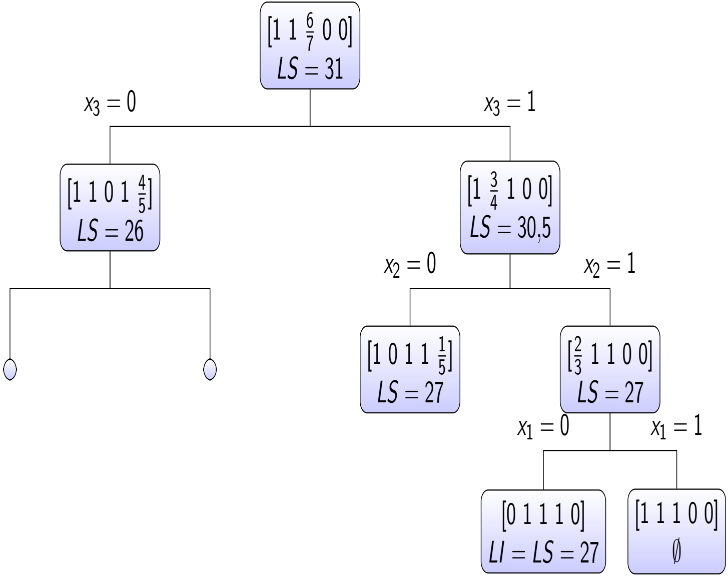
$$z = \text{Max } \{9x_1 + 10x_2 + 14x_3 + 3x_4 + 5x_5 :$$
$$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \ e \ x_1, \ldots, x_5 \in \{0, 1\}\}$$

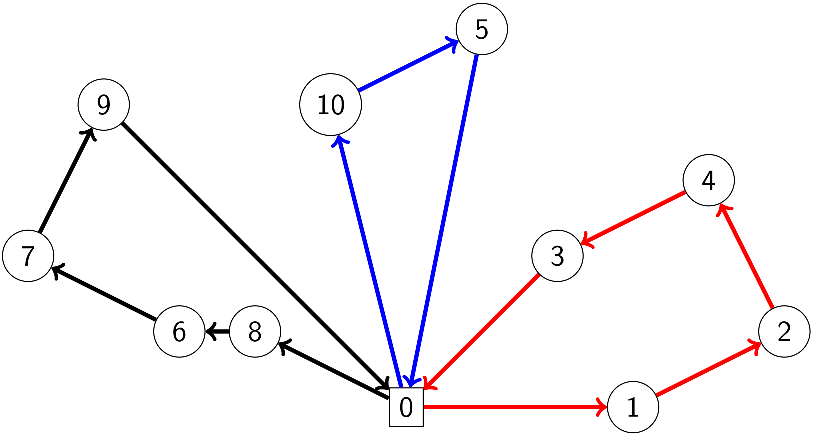


$$z = \text{Max } \{9x_1 + 10x_2 + 14x_3 + 3x_4 + 5x_5 :$$
$$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \ e \ x_1, \ldots, x_5 \in \{0, 1\}\}$$



$$z = \text{Max } \{9x_1 + 10x_2 + 14x_3 + 3x_4 + 5x_5 :$$
$$3x_1 + 4x_2 + 7x_3 + 2x_4 + 5x_5 \leq 13 \ e \ x_1, \ldots, x_5 \in \{0, 1\}\}$$




5
10
9
4
3
7
6
8
2
0
1

n	*Soluções/Caminhos*
5	

n	*Soluções/Caminhos*
5	24

n	*Soluções/Caminhos*
5	24
10	362880

n	*Soluções/Caminhos*
5	24
10	362880
20	$1{,}2 \cdot 10^{17}$

n	*Soluções/Caminhos*
5	24
10	362880
20	$1{,}2 \cdot 10^{17}$
100	$\approx 10^{156}$

$$
\begin{aligned}
99! = \quad & 933262154439441526816992388562667004907 \\
& 159682643816214685929638952175999932299 \\
& 156089414639761565182862536979208272237 \\
& 582511852109168640000000000000000000000
\end{aligned}
$$

$$99! \cdot 10^{-43} \approx 10^{113} s$$

n	*Soluções/Caminhos*
5	24
10	362880
20	$1{,}2 \cdot 10^{17}$
100	$\approx 10^{156}$

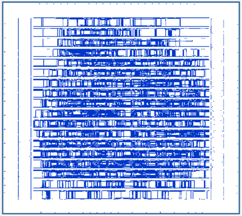
$$
\begin{aligned}
99! = \quad & 93326215443944152681699238856266700490 7\\
& 15968264381621468592963895217599993229 9\\
& 15608941463976156518286253697920827223 7\\
& 58251185210916864000000000000000000000 0
\end{aligned}
$$

$$99! \cdot 10^{-43} \approx 10^{113} s$$

Idade de universo $4{,}32 \cdot 10^{17} s$.

Optimal Tours

Information on the largest TSP instances solved to date can be found by following the links given below.



85,900 Locations in a VLSI Application
Solved in 2006

24,978 Cities in Sweden
Solved in 2004

15,112 Cities in Germany
Solved in 2001 [1]

[1] http://www.math.uwaterloo.ca/tsp/optimal/

Problema de roteamento de veículos

Objetivo: Planejar rotas dos veículos $1, \ldots, K$ minimizando a distância percorrida.

Decisão: Trajeto de cada veículo partindo do depósito (nó 0) e voltando ao depósito.

Restrições: Atender a demanda q_i de cada cliente $i = 1, \ldots, n$, respeitando a capacidade Q dos veículos.

Problema de roteamento de veículos – Variáveis

$$x_{ijk} = \begin{cases} 1, & \text{se veículo } k \text{ usa o trajeto de } i \text{ para } j, \\ 0, & \text{caso contrário.} \end{cases}$$

$0 \leq q_{ik} \leq Q$: Carga do veículo k ao sair do nó i.

Modelo

$$\text{Min} \sum_{k=1}^{K}\sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}x_{ijk} \tag{4}$$

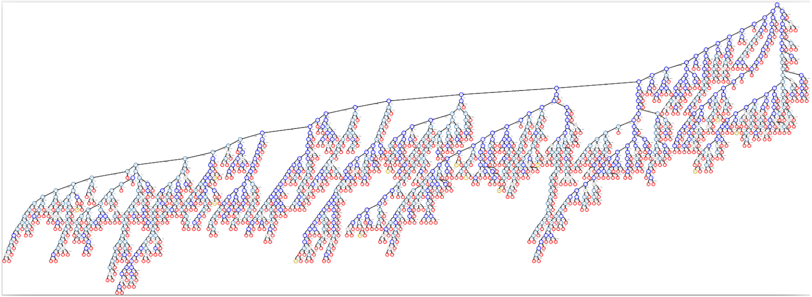
$$\sum_{k=1}^{K}\sum_{i=0}^{n} x_{ijk} = 1, \qquad j = 1, \dots, n, \tag{5}$$

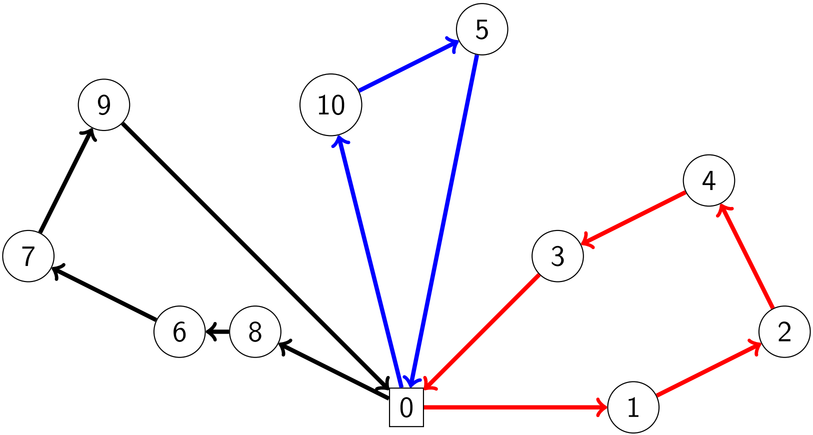
$$\sum_{i=0}^{n} x_{ihk} = \sum_{j=0}^{n} x_{hjk}, \qquad h = 1, \dots n, k = 1, \dots, K, \tag{6}$$

$$u_{jk} \geq u_{ik} + d_j - Q(1 - x_{ijk}), \qquad i = 0, \dots, n, j = 1, \dots, n, \quad k = 1, \dots, K, \tag{7}$$

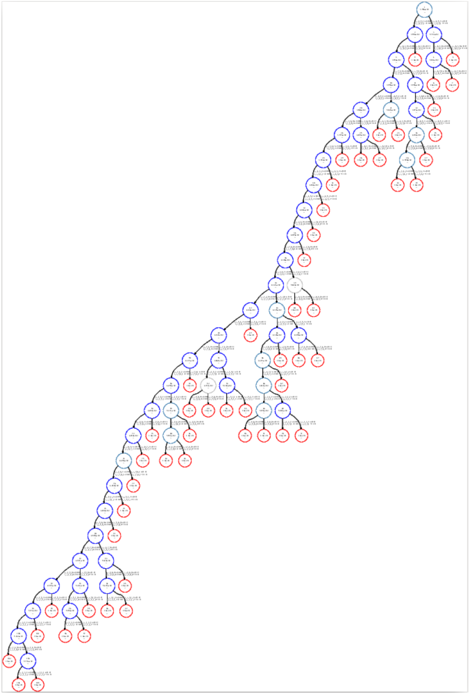
$$0 \leq u_{ik} \leq Q \qquad i = 1, \dots n, k = 1, \dots, K, \tag{8}$$

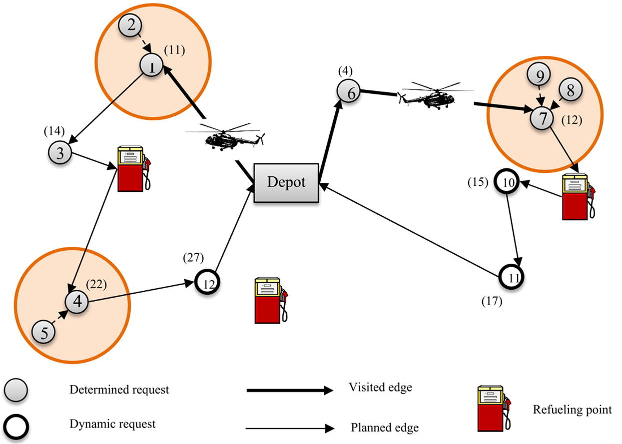
$$x_{ijk} \in \{0, 1\}, \qquad i = 0, \dots, n, j = 0, \dots n, \quad k = 1, \dots, K. \tag{9}$$



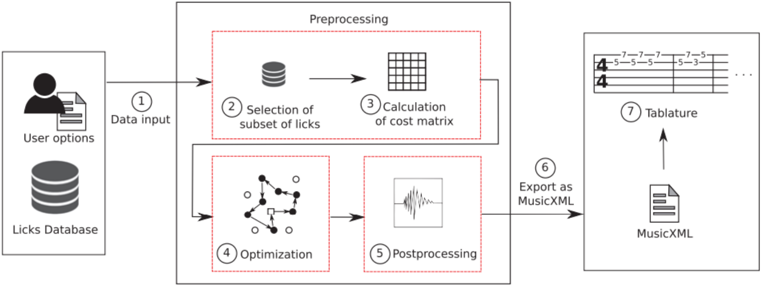


0
1
2
3
4
5
6
7
8
9
10

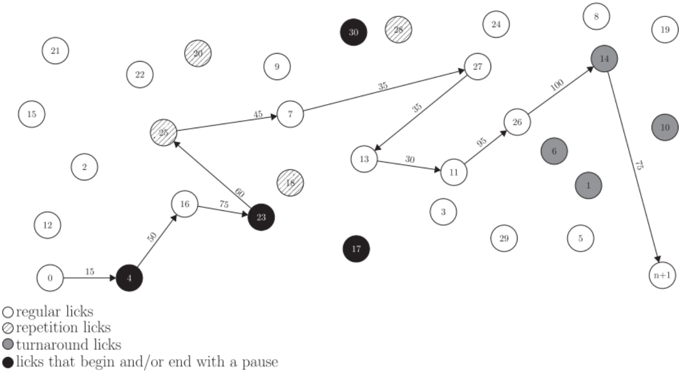




M. Alinaghian, M. Aghaie e M. S. Sabbagh. *A mathematical model for location of temporary relief centers and dynamic routing of aerial rescue vehicles*. Computers & Industrial Engineering, 2019.



S. Cunha, A. Subramanian e D. Herremans. *Generating guitar solos by integer programming*. Journal of the Operational Research Society, 69: 971–985, 2018.

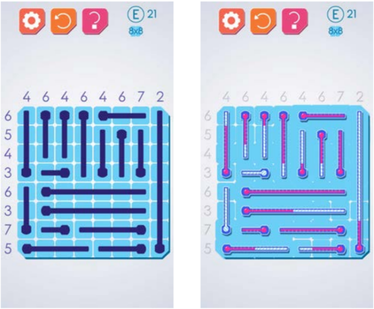


S. Cunha, A. Subramanian e D. Herremans. *Generating guitar solos by integer programming*. Journal of the Operational Research Society, 69: 971–985, 2018.

$$x_{ij} = \begin{cases} 1, & \text{if cell } (i,j) \text{ is filled,} \\ 0, & \text{otherwise.} \end{cases}$$

As is often the case with puzzles, we are just looking for a feasible solution, , so there is no optimization and

Figure 1. Smartphone App *Thermometer Puzzles*



(a) Puzzle (b) Solution

$$\sum_{j=1}^{n} x_{ij} = R_i, \quad i = 1, \ldots, n,$$

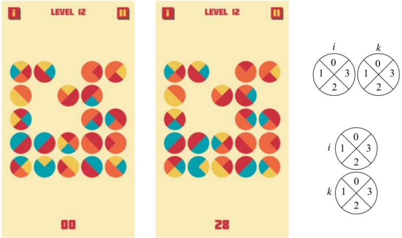
$$\sum_{i=1}^{n} x_{ij} = C_j, \quad j = 1, \ldots, n,$$

$$x_{ij} \geq x_{kl}, \quad (i,j,k,l) \in D,$$

$$x_{ij} \in \{0,1\}, \quad i,j = 1, \ldots, n.$$

Hartmann, S. *Puzzle – Solving Smartphone Puzzle Apps by Mathematical Programming. INFORMS Transactions on Education 18(2): 127-141, 2018*

$$x_{ij} = \begin{cases} 1, & \text{if } j \text{ left-turns should be applied to disk } i, \\ 0, & \text{otherwise.} \end{cases}$$



(a) Puzzle (b) Solution (c) Modeling approach

$$\text{Minimize} \quad \sum_{i=1}^{n} \sum_{j=0}^{3} j x_{ij}$$

$$\text{subject to} \quad \sum_{j=0}^{3} x_{ij} = 1, \quad i = 1, \ldots, n,$$

$$\sum_{j=0}^{3} d_{i,(3-j) \bmod 4} \cdot x_{ij} = \sum_{j=0}^{3} d_{k,(1-j) \bmod 4} \cdot x_{kj}, \quad (i,k) \in H,$$

$$\sum_{j=0}^{3} d_{i,(2-j) \bmod 4} \cdot x_{ij} = \sum_{j=0}^{3} d_{k,(0-j) \bmod 4} \cdot x_{kj}, \quad (i,k) \in V,$$

$$x_{ij} \in \{0,1\} \quad i, = 1, \ldots, n; \; j = 0, \ldots, 3.$$

Hartmann, S. *Puzzle – Solving Smartphone Puzzle Apps by Mathematical Programming. INFORMS Transactions on Education 18(2): 127-141, 2018*

Como decidir melhor em 10^{156} passos fáceis

Pedro Belin Castellucci

@pbcastellucci
lot.icmc.usp.br