

# Single Source Fixed Charge Network Flow Problem

Artur Rodrigues

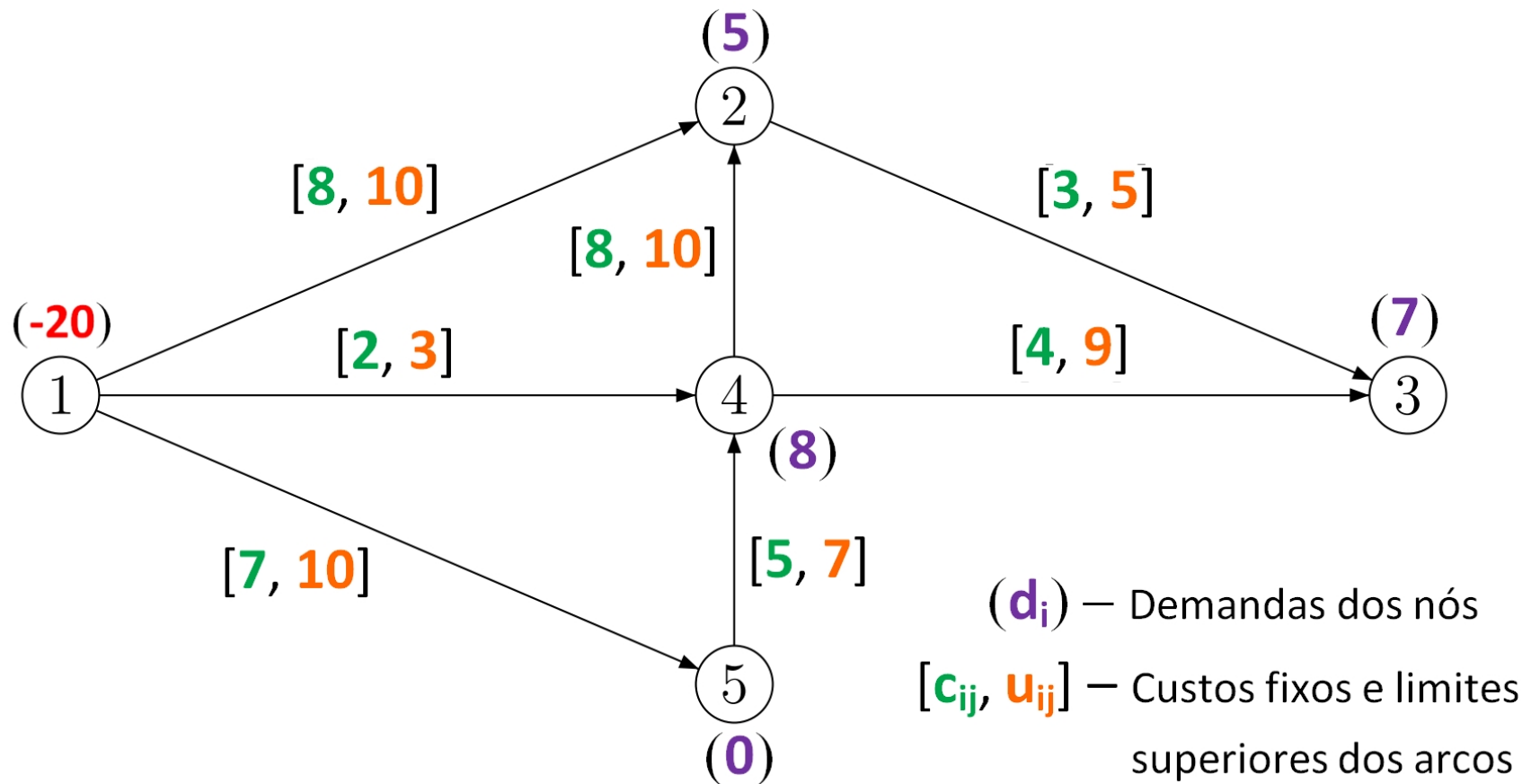
Luciana Maroun

Thanis Paiva

# Modelagem

- Baseia-se num grafo direcionado  $G = (N, A)$  com 1 único nó de oferta e os demais nós de demanda/transbordo.
- Os arcos são capacitados(limitados superiormente) e apresentam custos fixos.
- Objetivo: Selecionar um subconjunto de arcos tais que o fluxo que passe por eles atenda às demandas com um custo mínimo.

# Modelagem



# Formulação Matemática

Nó de *oferta*,  $b_f < 0$  e  $b_f = -\sum_{i \in N \setminus \{f\}} b_i$

Nós de *demanda*,  $b_i > 0$

Nós de *transbordo*,  $b_i = 0$

*Arcos*,  $\forall (i, j) \in A, c_{ij} \geq 0, u_{ij} \geq 0$

*Fluxo* definido pela variável  $x_{ij}$  e pela variável binária  $y_{ij}$  tal que:

$$y_{ij} = \begin{cases} 1 & \text{se } x_{ij} > 0 \\ 0 & \text{se } x_{ij} = 0 \end{cases}$$

# Formulação Matemática

$$\min \sum_{(i,j) \in A} c_{ij} y_{ij} \quad (1)$$

$$\sum_{j \in V^-(i)} x_{ij} - \sum_{j \in V^+(i)} x_{ij} = b_i, i \in N \quad (2)$$

$$0 \leq x_{ij} \leq u_{ij} y_{ij}, (i, j) \in A \quad (3)$$

$$y \in \{0, 1\} \quad (4)$$

# Heurística do Caminho Mínimo

$\forall (i,j) \in A$  faça

$$x_{ij} \leftarrow 0$$

enquanto  $\exists i \in V \mid b_i > 0$  faça

$$C \leftarrow \text{caminho\_minimo}(f, i)$$

$$fluxo \leftarrow \min(\text{capacidade}_C, b_i)$$

se  $fluxo = 0$  faça

retorne “Solução Inválida”

$\forall (i,j) \in A$  faça

$$u_{ij} \leftarrow u_{ij} - fluxo$$

$$c_{ij} \leftarrow 0$$

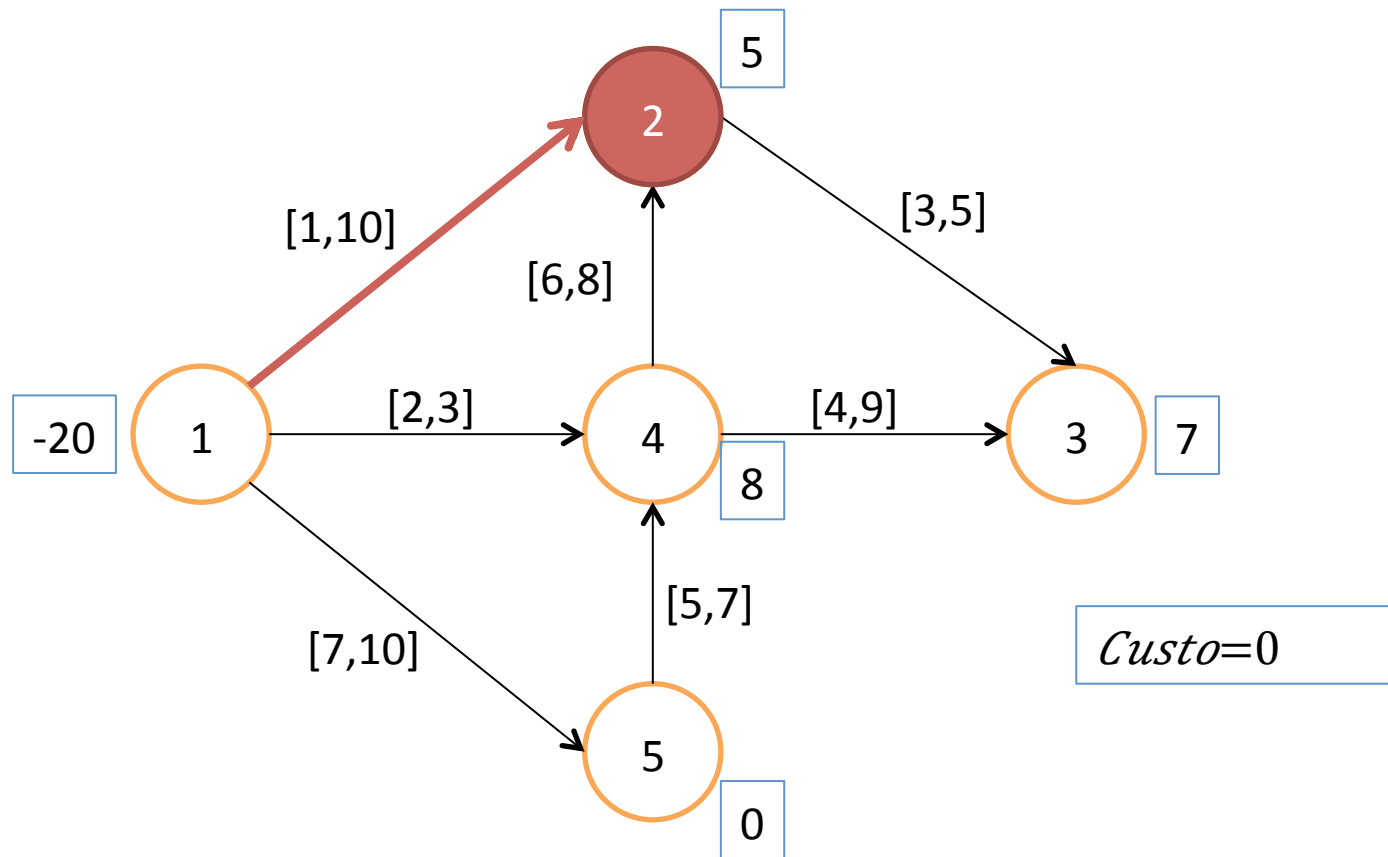
Complexidade:

$$O(kn^3)$$

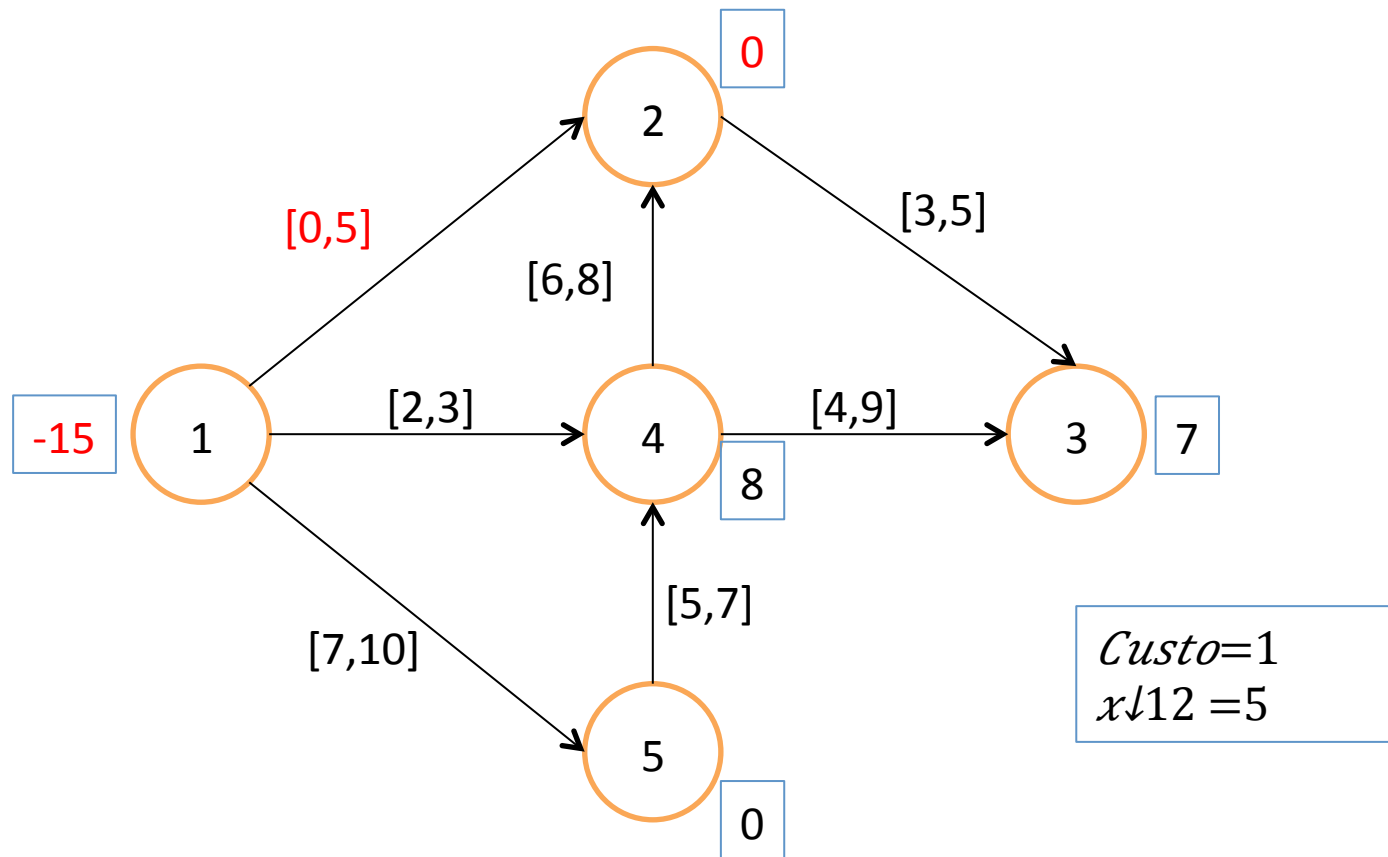
k: valor máximo de  
demanda

n: número de nós

# Exemplo

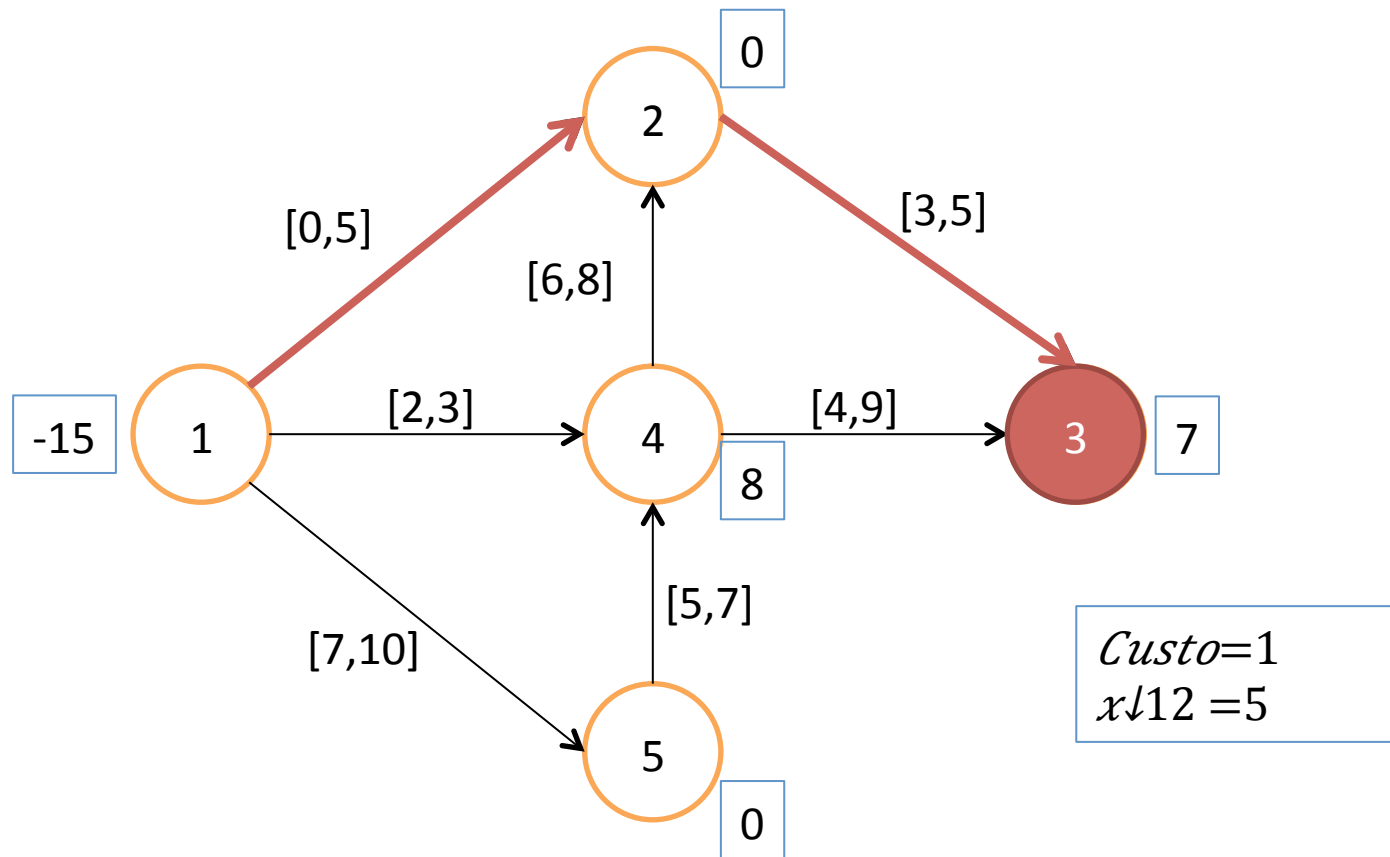


# Exemplo

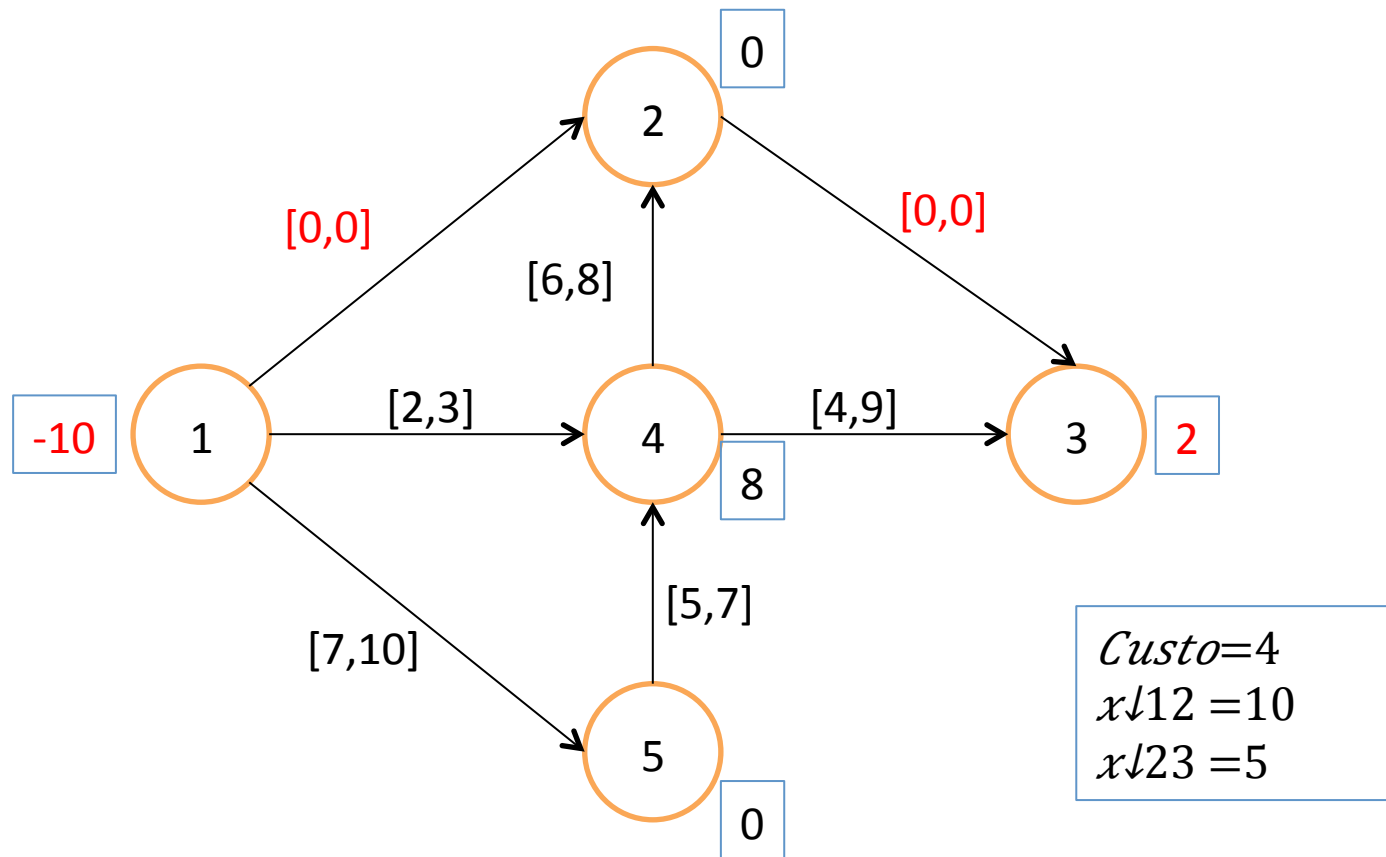




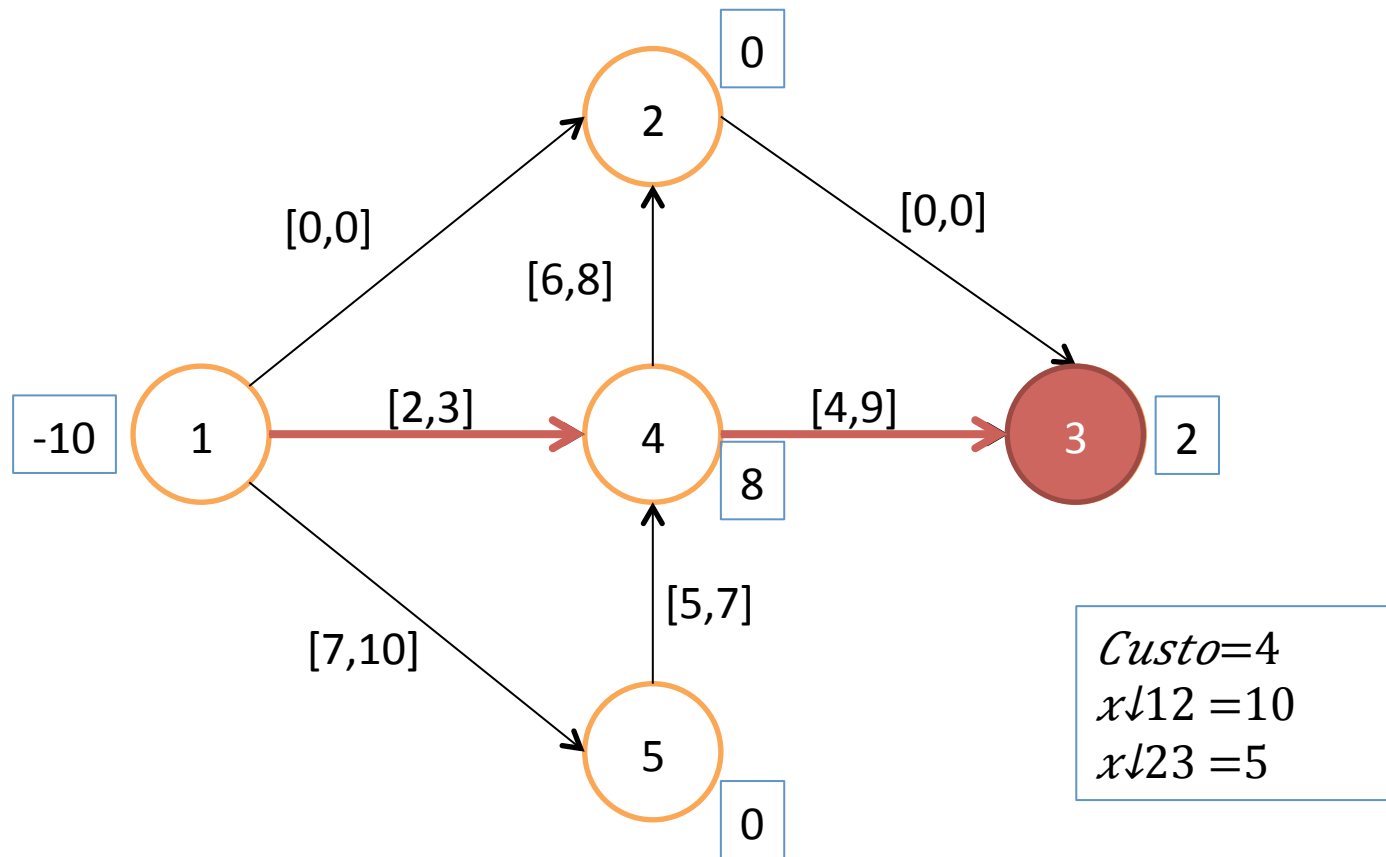
# Exemplo



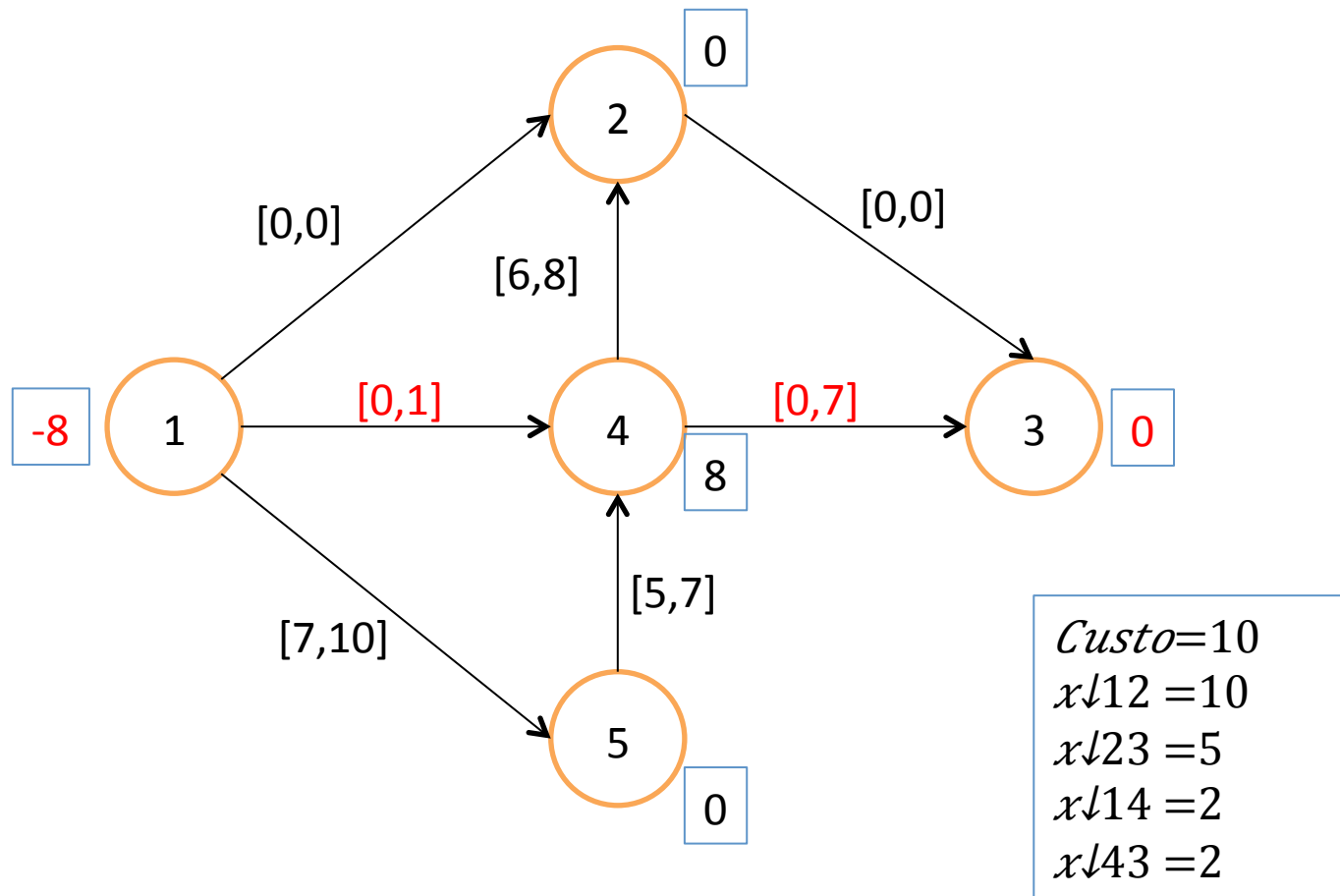
# Exemplo



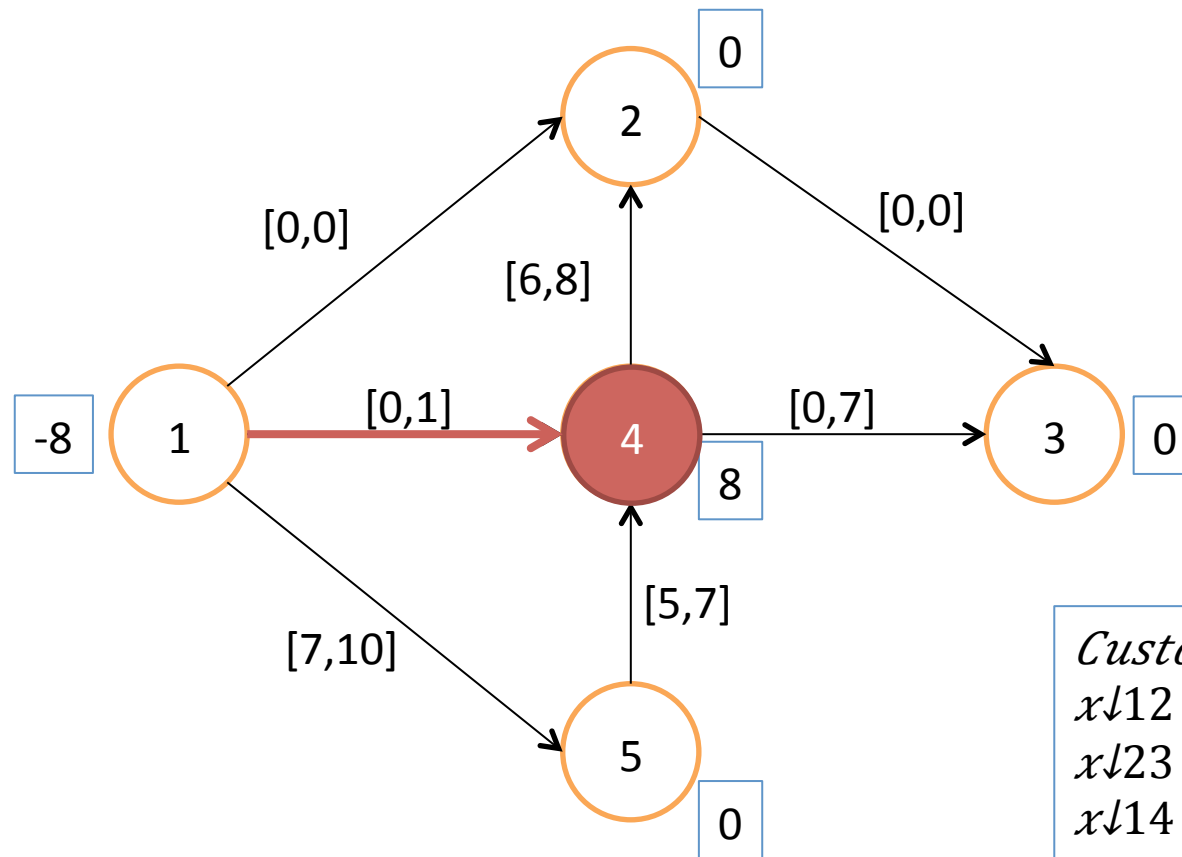
# Exemplo



# Exemplo

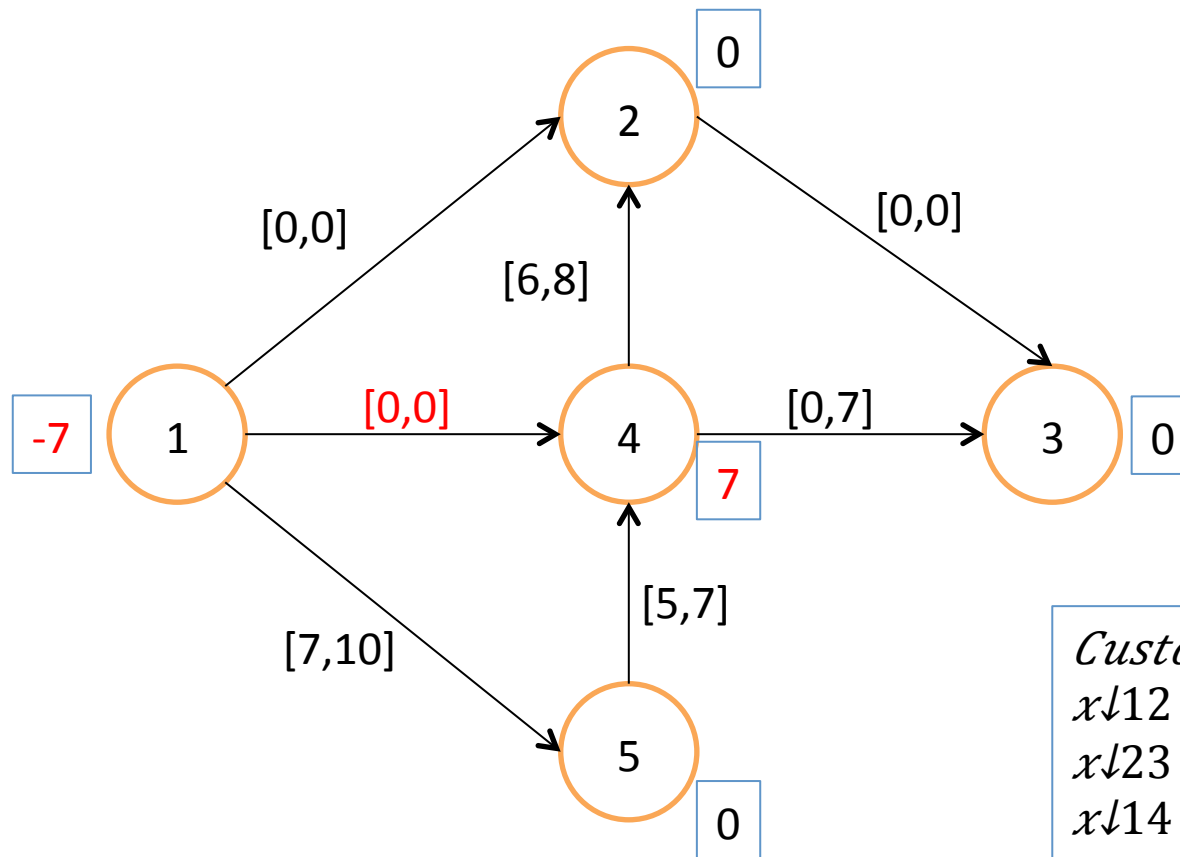


# Exemplo



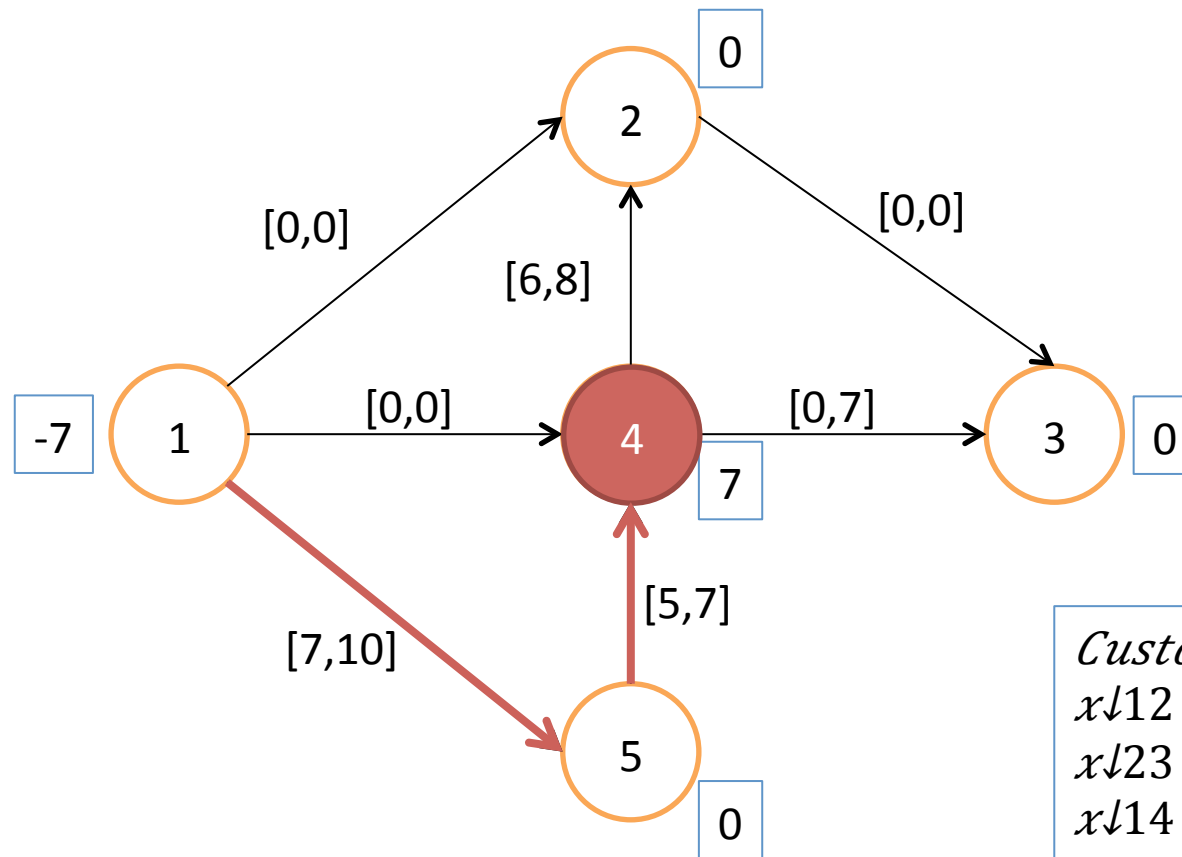
*Custo* = 10  
 $x \downarrow 12 = 10$   
 $x \downarrow 23 = 5$   
 $x \downarrow 14 = 2$   
 $x \downarrow 43 = 2$

# Exemplo



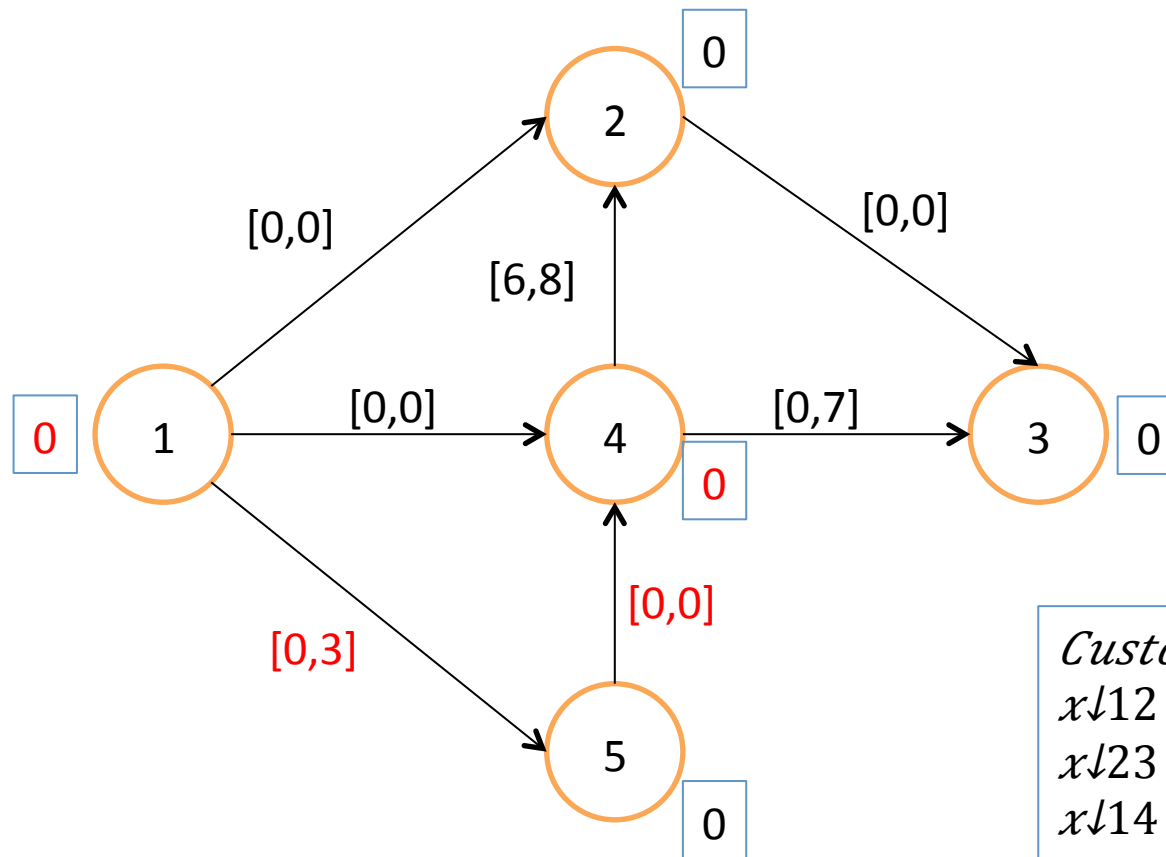
*Custo* = 10  
 $x_{12} = 10$   
 $x_{23} = 5$   
 $x_{14} = 3$   
 $x_{43} = 2$

# Exemplo



*Custo*=10  
 $x \downarrow 12 = 10$   
 $x \downarrow 23 = 5$   
 $x \downarrow 14 = 3$   
 $x \downarrow 43 = 2$

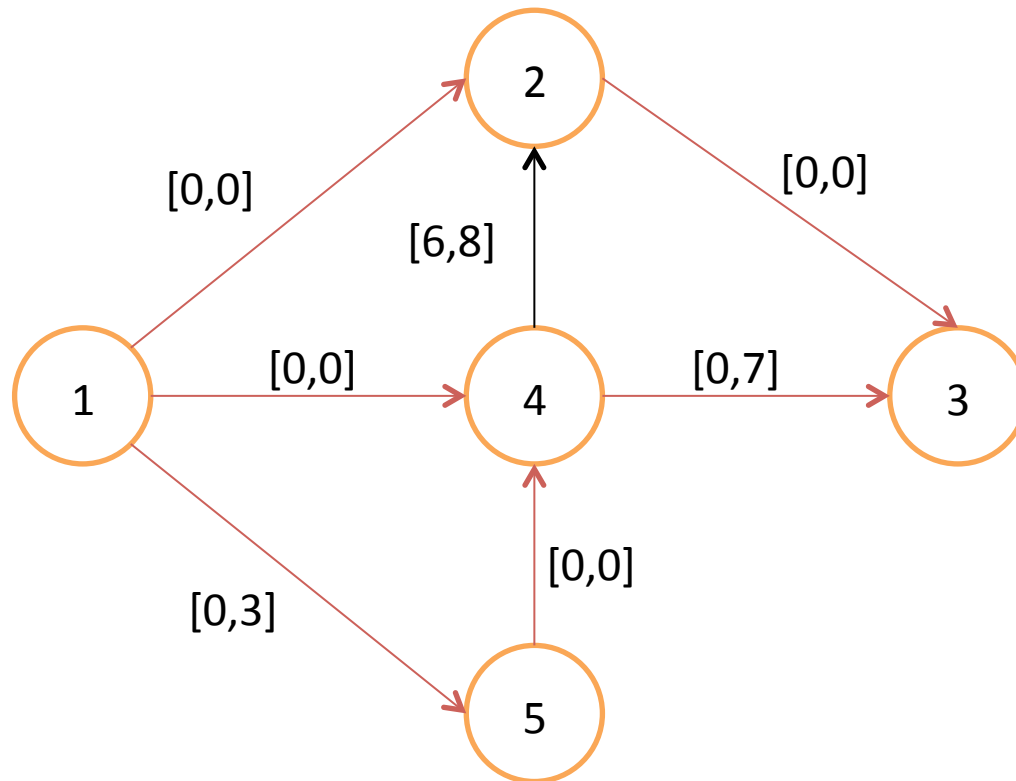
# Exemplo



*Custo* = 22  
 $x \downarrow 12 = 10$   
 $x \downarrow 23 = 5$   
 $x \downarrow 14 = 3$   
 $x \downarrow 43 = 2$   
 $x \downarrow 15 = 7$   
 $x \downarrow 54 = 7$



# Exemplo



Solução encontrada.

*Custo*=22  
 $x_{12} = 10$   
 $x_{23} = 5$   
 $x_{14} = 3$   
 $x_{43} = 2$   
 $x_{15} = 7$   
 $x_{54} = 7$

# Comportamento da Heurística

- Prioriza custos e não considera capacidades para escolher caminhos
- Caminhos escolhidos inicialmente podem não compor a solução ótima
- Um caminho não escolhido pode comportar um fluxo que na heurística percorre vários caminhos mínimos, estes totalizando um maior custo

# Implementação

- Python
  - *Container types* (listas e dicionários)
  - Vasto número de bibliotecas
- Biblioteca *NetworkX*
  - Estruturas e funções de grafos já prontas (ex.: *dijkstra-path*)
- Ainda mais rápido com PyPy (*just-in-time compiler*)

# Gerador de Instâncias

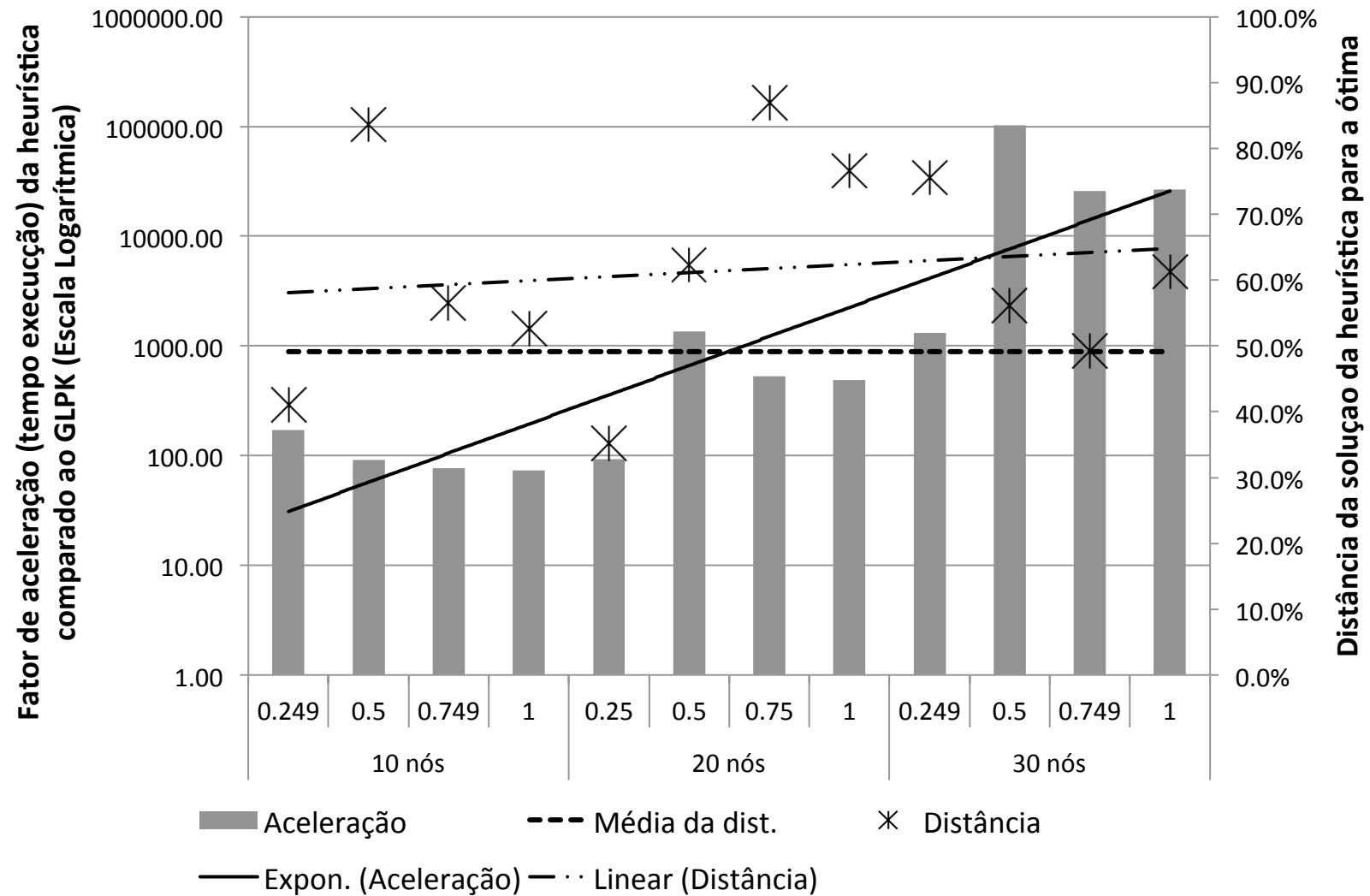
- Gera instâncias com base nos seguintes parâmetros:
  - Número de nós de demanda e transbordo
  - Densidade
  - Valor máximo para demanda
  - Valor máximo para capacidades nos arcos
  - Valor máximo para custos nos arcos
- Viabilidade garantida através do solver
- Objeto *python* e entrada para o *GLPSOL*

# Experimentos – Resultados

Número de nós	Densidade	Tempo (s)			Função Objetivo		
		GLPK	Heurística	Aceleração	GLPK	Heurística	Distância
10 nós	0.249	0.1	0.001	172.12	112	158	41.1%
	0.5	0.1	0.001	91.66	153	281	83.7%
	0.749	0.1	0.001	77.34	161	252	56.5%
	1	0.1	0.001	73.10	114	174	52.6%
20 nós	0.25	0.2	0.002	92.94	108	146	35.2%
	0.5	8.3	0.006	1355.10	355	576	62.3%
	0.75	3.3	0.006	524.14	161	301	87.0%
	1	5.1	0.010	487.99	94	166	76.6%
30 nós	0.249	10.7	0.008	1318.06	315	553	75.6%
	0.5	1390.7	0.014	102611.97	370	578	56.2%
	0.749	453.2	0.017	25966.88	205	306	49.3%
	1	925.7	0.035	26493.99	212	342	61.3%
Média							49.1%
DPA							16.4%

*Valor máximo para as demandas = 10*

# Experimentos – Resultados



# Experimentos – Resultados

Demanda Max	Tempo (s)			Função Objetivo		
	GLPK	Heurística	Aceleração	GLPK	Heurística	Distância
10	0.1	0.0007	153.14	51	56	9.8%
20	0.1	0.0005	216.45	40	45	12.5%
30	0.1	0.0008	128.53	39	59	51.3%
40	0.1	0.0006	168.35	77	81	5.2%
50	0.1	0.0005	200.00	45	60	33.3%
60	0.1	0.0006	177.94	46	52	13.0%
70	0.1	0.0005	202.84	49	56	14.3%

*Densidade = 0.249; Número de vértices = 10*

# Dúvidas?

<https://github.com/arturhoo/single-source-fcnf-heuristic>