A pricing game

Dois vendedores, 1 e 2, de um mesmo produto e três compradores, A, B, e C.

A pricing game

Dois vendedores, 1 e 2, de um mesmo produto
e três compradores, A, B, e C.

- A só pode comprar de 1 e C só pode comprar de 2;
- B pode comprar de 1 ou 2;

A pricing game

Dois vendedores, 1 e 2, de um mesmo produto
e três compradores, A, B, e C.

- A só pode comprar de 1 e C só pode comprar de 2;
- B pode comprar de 1 ou 2;

- 1 e 2 são os jogadores;
- cada estratégia é um preço p_i em $[0, 1]$;

A pricing game

Dois vendedores, 1 e 2, de um mesmo produto
e três compradores, A, B, e C.

- A só pode comprar de 1 e C só pode comprar de 2;
- B pode comprar de 1 ou 2;

- 1 e 2 são os jogadores;
- cada estratégia é um preço p_i em $[0, 1]$;

- B compra do mais barateiro, e de 1 em caso de empate.

A pricing game

Dois vendedores, 1 e 2, de um mesmo produto
e três compradores, A, B, e C.

- A só pode comprar de 1 e C só pode comprar de 2;
- B pode comprar de 1 ou 2;
- 1 e 2 são os jogadores;
- cada estratégia é um preço p_i em $[0, 1]$;
- B compra do mais barateiro, e de 1 em caso de empate.

Há equilíbrio de Nash para esse jogo?

A pricing game

Dois vendedores, 1 e 2, de um mesmo produto
e três compradores, A, B, e C.

- A só pode comprar de 1 e C só pode comprar de 2;
- B pode comprar de 1 ou 2;

- 1 e 2 são os jogadores;
- cada estratégia é um preço p_i em $[0, 1]$;

- B compra do mais barateiro, e de 1 em caso de empate.

Há equilíbrio de Nash para esse jogo?

Há um equilíbrio misto... Veja exercício 8 da lista 2.

Representações sucintas de jogos

Jogo de dois jogadores,
um com m estratégias, outro com n.

$2mn$ números são necessários para representar tal jogo.

Representações sucintas de jogos

Jogo de dois jogadores,
um com m estratégias, outro com n.

$2mn$ números são necessários para representar tal jogo.

Jogo com n jogadores, cada um com s estratégias.

ns^n números são necessários para representar tal jogo.

Representações sucintas de jogos

Jogo de dois jogadores,
um com m estratégias, outro com n.

$2mn$ números são necessários para representar tal jogo.

Jogo com n jogadores, cada um com s estratégias.

ns^n números são necessários para representar tal jogo.

Efeito prático: problema intratável...
Efeito teórico: problema trivial...

Representações sucintas de jogos

Jogo de dois jogadores,
um com m estratégias, outro com n.

$2mn$ números são necessários para representar tal jogo.

Jogo com n jogadores, cada um com s estratégias.

ns^n números são necessários para representar tal jogo.

Efeito prático: problema intratável...
Efeito teórico: problema trivial...

Se $s = 2$, a entrada tem tamanho $n2^n$ e
pode-se testar todos os $(2^2)^n$ suportes.

Isso é polinomial no tamanho da entrada...

Representações sucintas de jogos

Temos interesse em jogos com representação sucinta.

Representações sucintas de jogos

Temos interesse em jogos com representação sucinta.

Exemplos:

- Jogos gráficos

Representações sucintas de jogos

Temos interesse em jogos com representação sucinta.

Exemplos:

- Jogos gráficos
- Jogos esparsos

Representações sucintas de jogos

Temos interesse em jogos com representação sucinta.

Exemplos:

- Jogos gráficos
- Jogos esparsos
- Jogos simétricos

Representações sucintas de jogos

Temos interesse em jogos com representação sucinta.

Exemplos:

- Jogos gráficos
- Jogos esparsos
- Jogos simétricos
- Jogos anônimos

Representações sucintas de jogos

Temos interesse em jogos com representação sucinta.

Exemplos:

- Jogos gráficos
- Jogos esparsos
- Jogos simétricos
- Jogos anônimos
- Jogos de congestionamento

Representações sucintas de jogos

Temos interesse em jogos com representação sucinta.

Exemplos:

- Jogos gráficos
- Jogos esparsos
- Jogos simétricos
- Jogos anônimos
- Jogos de congestionamento
- Jogos de formação de rede

Jogo de balanceamento de carga

Dados:

- n tarefas
- m máquinas
- w_i: peso da tarefa i
- s_j: velocidade da máquina j

Jogo de balanceamento de carga

Dados:

- n tarefas
- m máquinas
- w_i: peso da tarefa i
- s_j: velocidade da máquina j

Cada jogador controla uma tarefa.

Conjuntos de estratégias $S_i = [m]$:
o jogador escolhe a qual máquina atribuir sua tarefa.

Jogo de balanceamento de carga

Dados:

- n tarefas
- m máquinas
- w_i: peso da tarefa i
- s_j: velocidade da máquina j

Cada jogador controla uma tarefa.

Conjuntos de estratégias $S_i = [m]$:
o jogador escolhe a qual máquina atribuir sua tarefa.

Vetor de estratégia: atribuição $A : [n] \rightarrow [m]$.

Jogo de balanceamento de carga

Dados:

- n tarefas
- m máquinas
- w_i: peso da tarefa i
- s_j: velocidade da máquina j

Cada jogador controla uma tarefa.

Conjuntos de estratégias $S_i = [m]$:
o jogador escolhe a qual máquina atribuir sua tarefa.

Vetor de estratégia: atribuição $A : [n] \to [m]$.

Custo de A para i, onde $j = A(i)$: $\text{custo}_i(A) = \sum_{\ell : A(\ell)=j} \frac{w_\ell}{s_j}$.

Jogo de balanceamento de carga

Jogo: n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$

É um jogo finito, logo tem equilíbrio (misto) de Nash.

Jogo de balanceamento de carga

Jogo: n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$

É um jogo finito, logo tem equilíbrio (misto) de Nash.

Estamos interessados apenas em estratégias puras:
balanceamento de carga com estratégias puras.

Jogo de balanceamento de carga

Jogo: n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$

É um jogo finito, logo tem equilíbrio (misto) de Nash.

Estamos interessados apenas em estratégias puras:
balanceamento de carga com estratégias puras.

Consideramos ainda dois casos:

- máquinas idênticas ($s_1 = \cdots = s_m$)
- máquinas relacionadas (caso geral)

Jogo de balanceamento de carga

Jogo: n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$

É um jogo finito, logo tem equilíbrio (misto) de Nash.

Estamos interessados apenas em estratégias puras:
balanceamento de carga com estratégias puras.

Consideramos ainda dois casos:

- máquinas idênticas ($s_1 = \cdots = s_m$)
- máquinas relacionadas (caso geral)

O jogo com estratégias puras tem equilíbrio?

O jogo tem equilíbrio?

Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

O jogo tem equilíbrio?

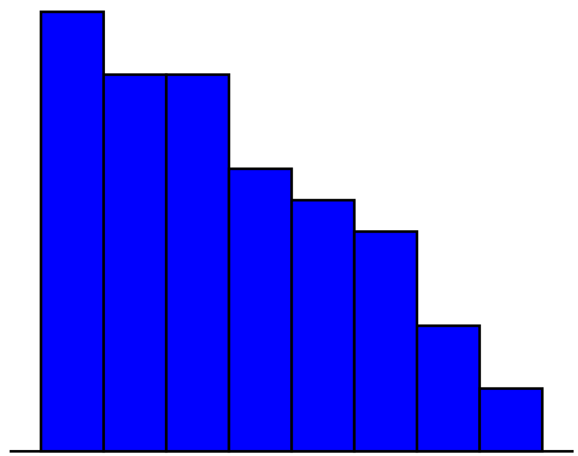
Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

Dada uma atribuição $A : [n] \to [m]$, considere o vetor com a carga de cada máquina, em ordem decrescente.

O jogo tem equilíbrio?

Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

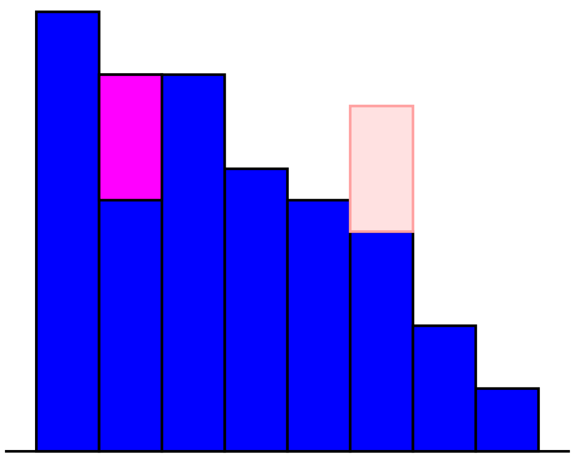
Dada uma atribuição $A : [n] \to [m]$, considere o vetor com a carga de cada máquina, em ordem decrescente.

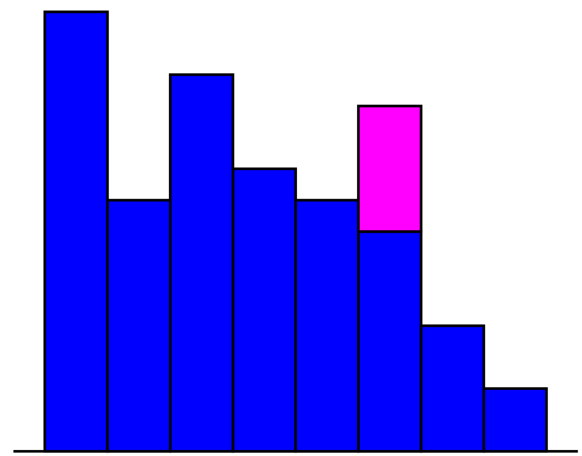


O jogo tem equilíbrio?

Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

Dada uma atribuição $A : [n] \to [m]$, considere o vetor com a carga de cada máquina, em ordem decrescente.

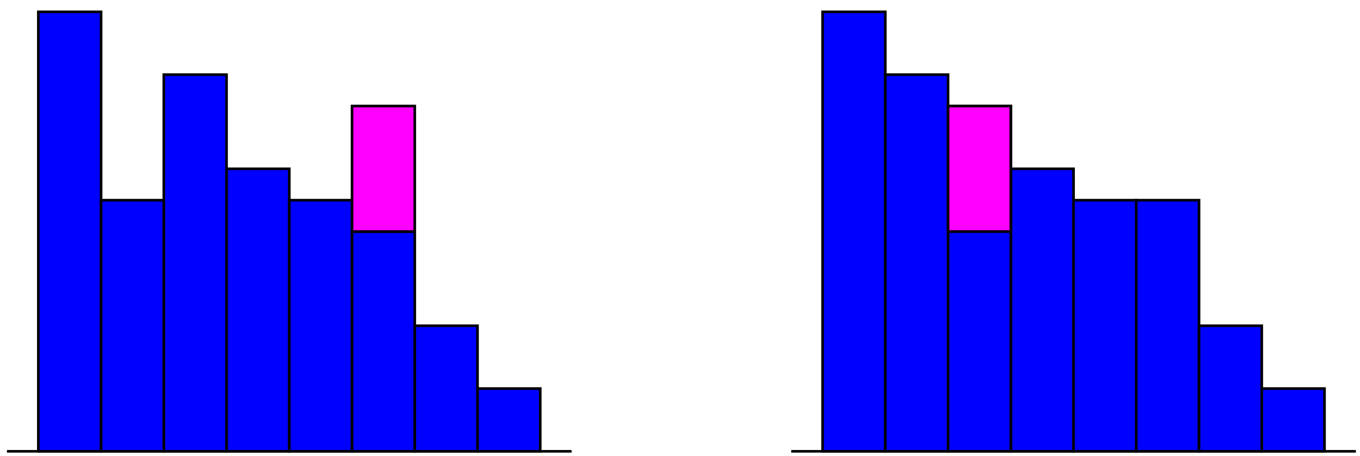




O jogo tem equilíbrio?

Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

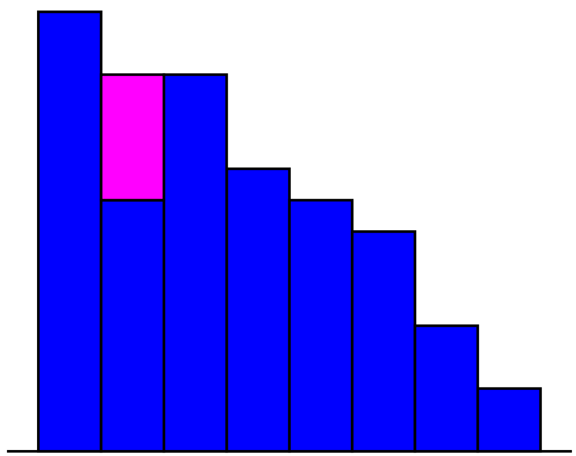
Dada uma atribuição $A : [n] \to [m]$, considere o vetor com a carga de cada máquina, em ordem decrescente.

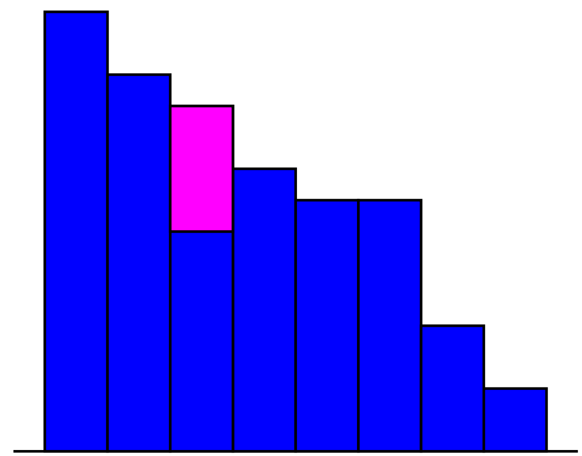


O jogo tem equilíbrio?

Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

Dada uma atribuição $A : [n] \to [m]$, considere o vetor com a carga de cada máquina, em ordem decrescente.

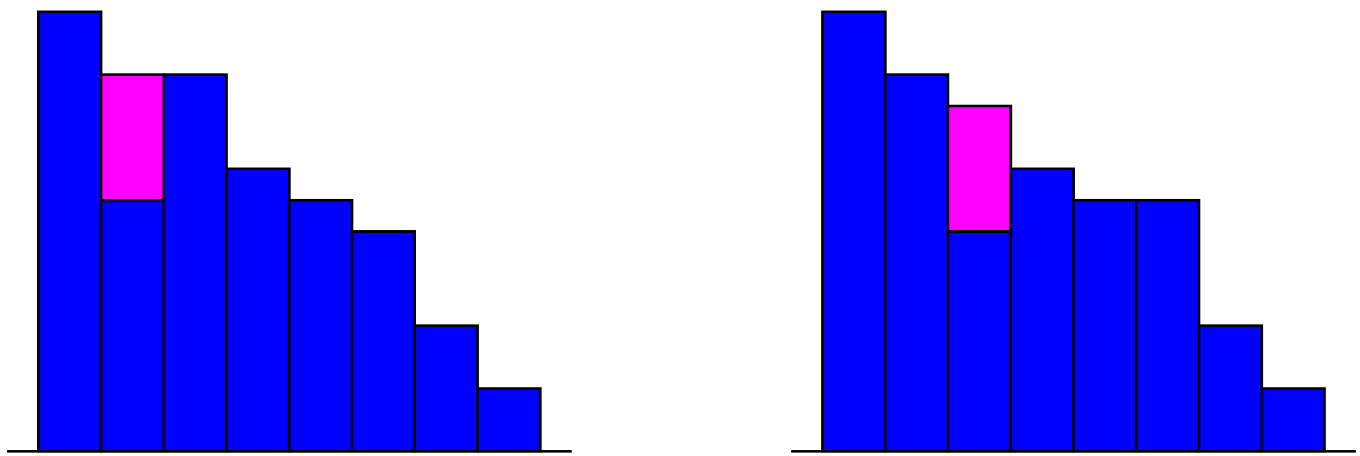




O jogo tem equilíbrio?

Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

Dada uma atribuição $A : [n] \to [m]$, considere o vetor com a carga de cada máquina, em ordem decrescente.

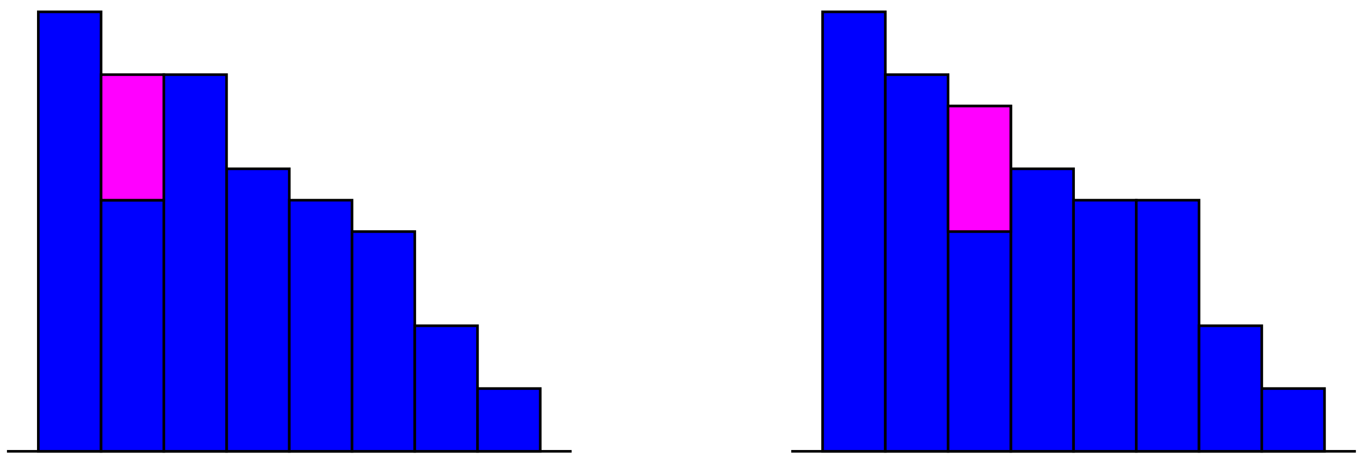


Migração vai para vetor lexicograficamente menor.

O jogo tem equilíbrio?

Proposição: O jogo de balanceamento de cargas dado por n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$ com estratégias puras tem pelo menos um equilíbrio.

Dada uma atribuição $A : [n] \to [m]$, considere o vetor com a carga de cada máquina, em ordem decrescente.



Migração vai para vetor lexicograficamente menor.

Processo termina e num equilíbrio.

Duas perguntas

Jogo de balanceamento de cargas com estratégias puras

- Quão ruim pode ser um equilíbrio em comparação com o chamado ótimo social?

Duas perguntas

Jogo de balanceamento de cargas com estratégias puras

- Quão ruim pode ser um equilíbrio em comparação com o chamado ótimo social?
- Quanto tempo para chegar a um equilíbrio?

Duas perguntas

Jogo de balanceamento de cargas com estratégias puras

- Quão ruim pode ser um equilíbrio em comparação com o chamado ótimo social?
- Quanto tempo para chegar a um equilíbrio?

Aqui o custo social de uma atribuição A é a carga da máquina mais carregada, ou seja, é o makespan.

(função social não utilitária)

Duas perguntas

Jogo de balanceamento de cargas com estratégias puras

- Quão ruim pode ser um equilíbrio em comparação com o chamado ótimo social?
- Quanto tempo para chegar a um equilíbrio?

Aqui o custo social de uma atribuição A é a carga da máquina mais carregada, ou seja, é o makespan.

(função social não utilitária)

O makespan mínimo é o ótimo social, denotado por OPT.

Duas perguntas

Jogo de balanceamento de cargas com estratégias puras

- Quão ruim pode ser um equilíbrio em comparação com o chamado ótimo social?
- Quanto tempo para chegar a um equilíbrio?

Aqui o custo social de uma atribuição A é a carga da máquina mais carregada, ou seja, é o makespan.

(função social não utilitária)

O makespan mínimo é o ótimo social, denotado por OPT.

Dados n, m, $w_1, \ldots, w_n$, $s_1, \ldots, s_m$,
determinar o makespan mínimo é um problema NP-difícil.

Duas medidas de qualidade

Preço da anarquia PoA(m):
maior razão entre o custo social de um equilíbrio de um jogo com m máquinas e o ótimo social.

Duas medidas de qualidade

Preço da anarquia PoA(m):
maior razão entre o custo social de um equilíbrio de um jogo com m máquinas e o ótimo social.

Preço da estabilidade PoE(m):
menor razão entre o custo social de um equilíbrio de um jogo com m máquinas e o ótimo social.

Duas medidas de qualidade

Preço da anarquia PoA(m):
maior razão entre o custo social de um equilíbrio de um jogo com m máquinas e o ótimo social.

Preço da estabilidade PoE(m):
menor razão entre o custo social de um equilíbrio de um jogo com m máquinas e o ótimo social.

Cada um vale pelo menos 1.

Duas medidas de qualidade

Preço da anarquia PoA(m):
maior razão entre o custo social de um equilíbrio de um jogo com m máquinas e o ótimo social.

Preço da estabilidade PoE(m):
menor razão entre o custo social de um equilíbrio de um jogo com m máquinas e o ótimo social.

Cada um vale pelo menos 1.

Preço da estabilidade é 1 para o balanceamento de carga.

Duas medidas de qualidade

Preço da anarquia PoA(m):
maior razão entre o custo social de um equilíbrio
de um jogo com m máquinas e o ótimo social.

Preço da estabilidade PoE(m):
menor razão entre o custo social de um equilíbrio
de um jogo com m máquinas e o ótimo social.

Cada um vale pelo menos 1.

Preço da estabilidade é 1 para o balanceamento de carga.

Prova anterior ajustada:

Duas medidas de qualidade

Preço da anarquia PoA(m):
maior razão entre o custo social de um equilíbrio
de um jogo com m máquinas e o ótimo social.

Preço da estabilidade PoE(m):
menor razão entre o custo social de um equilíbrio
de um jogo com m máquinas e o ótimo social.

Cada um vale pelo menos 1.

Preço da estabilidade é 1 para o balanceamento de carga.

Prova anterior ajustada:

Comece de configuração com makespan mínimo.
O makespan nunca aumenta, e termina num equilíbrio.

Preço da anarquia

Exemplo: $m = 2$, $s_1 = s_2 = 1$ e $w_1 = w_2 = 2$, $w_3 = w_4 = 1$.

Preço da anarquia

Exemplo: $m = 2$, $s_1 = s_2 = 1$ e $w_1 = w_2 = 2$, $w_3 = w_4 = 1$.

Preço da anarquia

Exemplo: $m = 2$, $s_1 = s_2 = 1$ e $w_1 = w_2 = 2$, $w_3 = w_4 = 1$.

Makespan mínimo: 3 (da esquerda)

Na direita, equilíbrio com makespan 4.

Preço da anarquia

Exemplo: $m = 2$, $s_1 = s_2 = 1$ e $w_1 = w_2 = 2$, $w_3 = w_4 = 1$.

Makespan mínimo: 3 (da esquerda)

Na direita, equilíbrio com makespan 4.

Preço da anarquia pelo menos 4/3.

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \left(2 - \frac{2}{m+1}\right)$.

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \left(2 - \frac{2}{m+1}\right)$.

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \left(2 - \frac{2}{m+1}\right)$.

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A

i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \left(2 - \frac{2}{m+1}\right)$.

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A

i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A

Se só i^* em j^*, então $\mathrm{custo}(A) = \mathrm{OPT}(J)$ e nada a provar.
Senão $w_{i^*} \leq \frac{1}{2}\,\mathrm{custo}(A)$.

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \left(2 - \frac{2}{m+1}\right)$.

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A

i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A

Se só i^* em j^*, então $\text{custo}(A) = \text{OPT}(J)$ e nada a provar.
Senão $w_{i^*} \leq \frac{1}{2}\,\text{custo}(A)$.

Não há máquina com carga menor que $\ell(j^*) - w_{i^*}$
(ou i^* teria incentivo para ir para tal máquina).

Caso de máquinas idênticas

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A
i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A
$w_{i^*} \leq \frac{1}{2}\,\mathrm{custo}(A)$.

Não há máquina com carga menor que $\ell(j^*) - w_{i^*}$.

Caso de máquinas idênticas

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A
i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A
$w_{i^*} \leq \frac{1}{2}\,\text{custo}(A)$.

Não há máquina com carga menor que $\ell(j^*) - w_{i^*}$.
Ou seja, para todo $j \in [m]$,

$$\ell(j) \;\geq\; \ell(j^*) - w_{i^*} \;\geq\; \text{custo}(A) - \frac{1}{2}\,\text{custo}(A) \;=\; \frac{1}{2}\,\text{custo}(A)$$

Caso de máquinas idênticas

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A
i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A
$w_{i^*} \leq \frac{1}{2}\,\text{custo}(A)$.

Não há máquina com carga menor que $\ell(j^*) - w_{i^*}$.
Ou seja, para todo $j \in [m]$,

$$\ell(j) \;\geq\; \ell(j^*) - w_{i^*} \;\geq\; \text{custo}(A) - \frac{1}{2}\,\text{custo}(A) \;=\; \frac{1}{2}\,\text{custo}(A)$$

e

$$\begin{aligned} \text{OPT}(J) \;&\geq\; \frac{\sum_{i\in[n]} w_i}{m} \;=\; \frac{\sum_{j\in[m]} \ell(j)}{m} \\ &\geq\; \frac{\text{custo}(A) + (m-1)\text{custo}(A)/2}{m} \;=\; \frac{(m+1)\text{custo}(A)}{2m} \end{aligned}$$

Caso de máquinas idênticas

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A
i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A
$w_{i^*} \leq \frac{1}{2}\,\mathrm{custo}(A)$.

Não há máquina com carga menor que $\ell(j^*) - w_{i^*}$.

Ou seja, $\ell(j) \;\geq\; \frac{1}{2}\,\mathrm{custo}(A)$ para todo $j \in [m]$ e

Caso de máquinas idênticas

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A
i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A
$w_{i^*} \leq \frac{1}{2}\,\mathrm{custo}(A)$.

Não há máquina com carga menor que $\ell(j^*) - w_{i^*}$.

Ou seja, $\ell(j) \;\geq\; \frac{1}{2}\,\mathrm{custo}(A)$ para todo $j \in [m]$ e

$$\mathrm{OPT}(J) \geq \frac{(m+1)\mathrm{custo}(A)}{2m},$$

donde se conclui que

Caso de máquinas idênticas

Prova: Jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$.

j^*: máquina com carga máxima em A
i^*: tarefa mais leve em j^*

$\ell(j)$: carga da máquina j em A
$w_{i^*} \leq \frac{1}{2}\,\mathrm{custo}(A)$.

Não há máquina com carga menor que $\ell(j^*) - w_{i^*}$.

Ou seja, $\ell(j) \;\geq\; \frac{1}{2}\,\mathrm{custo}(A)$ para todo $j \in [m]$ e

$$\mathrm{OPT}(J) \geq \frac{(m+1)\mathrm{custo}(A)}{2m},$$

donde se conclui que

$$\mathrm{custo}(A) \;\leq\; \Big(2 - \frac{2}{m+1}\Big)\mathrm{OPT}(J). \;\blacksquare$$

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \left(2 - \frac{2}{m+1}\right)$.

Prova:
Para todo jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$,

$$\text{custo}(A) \;\leq\; \left(2 - \frac{2}{m+1}\right)\text{OPT}(J). \;\blacksquare$$

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \big(2 - \frac{2}{m+1}\big)$.

Prova:
Para todo jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$,

$$\text{custo}(A) \;\leq\; \Big(2 - \frac{2}{m+1}\Big)\text{OPT}(J). \;\blacksquare$$

Para $m = 2$, análise é justa para o exemplo pois

$$\frac{4}{3} \;=\; \Big(2 - \frac{2}{2+1}\Big).$$

Caso de máquinas idênticas

Teorema: Para o jogo de balanceamento de carga em m máquinas idênticas, PoA$(m) = \left(2 - \frac{2}{m+1}\right)$.

Prova:
Para todo jogo $J = (n, m, w)$ e atribuição $A : [n] \to [m]$,

$$\mathrm{custo}(A) \;\leq\; \left(2 - \frac{2}{m+1}\right)\mathrm{OPT}(J). \;\blacksquare$$

Para $m = 2$, análise é justa para o exemplo pois

$$\frac{4}{3} \;=\; \left(2 - \frac{2}{2+1}\right).$$

Exercício: Mostre um exemplo para um m arbitrário que prove que a análise é justa para todo m.