MO829
Tópicos em Teoria da Computação
Teoria dos Jogos Algorítmica

Rafael C. S. Schouery
rafael@ic.unicamp.br

Universidade Estadual de Campinas

1º semestre/2017

Jogo de balanceamento de carga

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
- Escolhe em qual máquina aloca a tarefa
- Conjuntos de estratégias do jogador i é $S_i = [m]$

Jogo de balanceamento de carga

As escolhas dos jogadores geram uma *atribuição* de tarefas às maquinas:

$$A : [n] \to [m]$$

A carga de uma máquina j é:

$$\ell_j = \sum_{\substack{i \in [n] \\ j = A(i)}} \frac{w_i}{s_j}$$

O custo de A para um jogador i é ℓ_j tal que $j = A(i)$

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

Consideramos ainda dois casos:

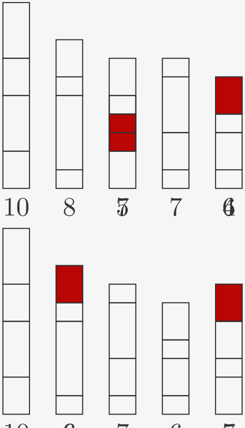
- máquinas *idênticas* ($s_1 = \cdots = s_m$)
- máquinas *relacionadas* (caso geral)

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

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



Vetores encontrados:

- $(10, 8, 7, 7, 4)$
- $(10, 8, 7, 6, 5)$
- $(10, 7, 7, 6, 6)$

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*

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

Seja $\mathcal{J}(m)$ o conjunto de todos jogos de balanceamento de carga com m máquinas

- Nash(J) é o conjunto de equilíbrios de um jogo J
- custo(A) é o custo da atribuição A para J
- opt(J) é o custo mínimo de uma atribuição para J

Preço da anarquia PoA(m):

- É o valor máximo da razão entre o pior custo de um equilíbrio e o custo da solução ótima

$$\text{PoA}(m) = \max_{J \in \mathcal{J}(m)} \max_{A \in \text{Nash}(J)} \frac{\text{custo}(A)}{\text{opt}(J)}$$

Duas medidas de qualidade

Seja $\mathcal{J}(m)$ o conjunto de todos jogos de balanceamento de carga com m máquinas

- Nash(J) é o conjunto de equilíbrios de um jogo J
- custo(A) é o custo da atribuição A para J
- opt(J) é o custo mínimo de uma atribuição para J

Preço da estabilidade PoS(m):

- É o valor máximo da razão entre o melhor custo de um equilíbrio e o custo da solução ótima

$$\text{PoS}(m) = \max_{J \in \mathcal{J}(m)} \min_{A \in \text{Nash}(J)} \frac{\text{custo}(A)}{\text{opt}(J)}$$

Duas medidas de qualidade

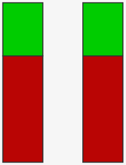
O valores de PoA(m) e de PoS(m) valem pelo menos 1

Preço da estabilidade é 1:

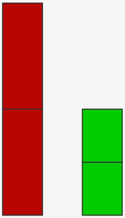
- Comece com a configuração de makespan mínimo e vá aplicando melhores respostas
- O makespan nunca aumenta e termina num equilíbrio

Preço da anarquia

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



Makespan mínimo: 3



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$(m) = \left(2 - \frac{2}{m+1}\right)$

Prova: Jogo $J = (n, m, w)$

- Atribuição em equilíbrio $A : [n] \to [m]$
- j^*: máquina com carga máxima em A
- i^*: tarefa de menor peso em j^*
- $\ell(j)$: carga da máquina j em A

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

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

Caso de máquinas idênticas

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

Usando o fato que

$$\text{opt}(J) \geq \frac{(m+1)\text{custo}(A)}{2m}$$

temos que

$$\text{custo}(A) \;\leq\; \left(2 - \frac{2}{m+1}\right)\text{opt}(J) \quad \square$$

Caso de máquinas idênticas

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

Tempo de convergência

Para máquinas idênticas, há sequência curta de melhoras de qualquer atribuição inicial para um equilíbrio

Política da resposta ótima de peso máximo:

- Ative apenas uma tarefa insatisfeita de peso máximo por vez
- Uma tarefa ativada muda para a melhor máquina

Tempo de convergência

Teorema: A política de resposta ótima de peso máximo atinge um equilíbrio depois de no máximo n passos

Prova:

- ℓ_j^t: carga da máquina j no tempo t
- A tarefa i migra de j para j^* no tempo t

Se a tarefa k em j^* se torna insatisfeita em $t+1$ e quer migrar para a máquina j

$$\ell_j^{t+1} + w_k < \ell_{j^*}^{t+1} = \ell_{j^*}^t + w_i < \ell_j^t = \ell_j^{t+1} + w_i$$

(pois k prefere a máquina j no tempo $t+1$) (pois k prefere a máquina j no tempo $t+1$) (pois i migrou para j^* no tempo t) (pois i prefere a máquina j^* no tempo t) (pois i migrou de j no tempo t)

Isto é, $w_k < w_i$

Se a tarefa k em j^* se torna insatisfeita em $t+1$ e quer migrar para a máquina $j' \neq j$, suponha que $w_k \geq w_i$

$$\ell_{j'}^t + w_i \leq \ell_{j'}^t + w_k = \ell_{j'}^{t+1} + w_k < \ell_{j^*}^{t+1} = \ell_{j^*}^t + w_i$$

(pois $w_k \geq w_i$) (pois a carga de j' não mudou) (pois k ficou insatisfeita) (pois i migrou para j^*)

Mas então, i prefere j' do que j^*. Contradição pela escolha de

Tempo de convergência

Teorema: A política de resposta ótima de peso máximo atinge um equilíbrio depois de no máximo n passos

Prova:

- Quando uma tarefa migra, torna insatisfeitas apenas tarefas com peso menor
- Isto é, após migrar uma tarefa nunca mais fica insatisfeita
- Cada tarefa migra no máximo uma vez
- O equilíbrio é atingido em no máximo n passos $\square$

Caso de máquinas relacionadas

Teorema: Para o jogo de balanceamento de carga em m máquinas relacionadas, $\text{PoA}(m) = \Theta\left(\frac{\lg m}{\lg \lg m}\right)$

Primeira parte:
Mostrar que para jogos $J = (n, m, w, s)$ e atribuições $A : [n] \to [m]$ em equilíbrio vale que

$$\text{custo}(A) = \text{O}\left(\frac{\lg m}{\lg \lg m}\right) \text{opt}(J)$$

Segunda parte:
Apresentar jogo $J = (n, m, w, s)$ e atribuição $A : [n] \to [m]$ em equilíbrio tal que

$$\text{custo}(A) = \Omega\left(\frac{\lg m}{\lg \lg m}\right) \text{opt}(J)$$

Primeira parte

Lema: Para jogos $J = (n, m, w, s)$ e atribuições $A : [n] \to [m]$ em equilíbrio, vale que

$$\text{custo}(A) = \mathrm{O}\left(\frac{\lg m}{\lg \lg m}\right) \text{opt}(J)$$

Prova: Seja $c = \lfloor \text{custo}(A)/\text{opt}(J) \rfloor$

Vamos mostrar que $c \leq \Gamma^{-1}(m)$, onde Γ é a função gama

- $\Gamma(k) = (k-1)!$ para todo natural k
- $\Gamma^{-1}(m) = \Theta\left(\frac{\lg m}{\lg \lg m}\right)$

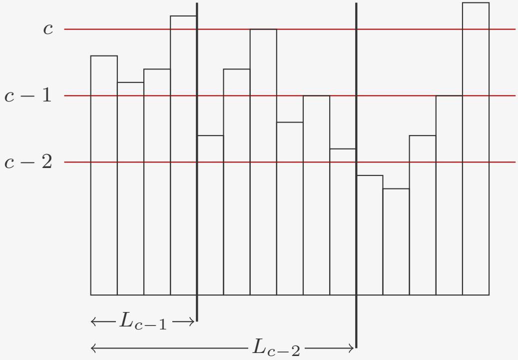
SPG, $s_1 \geq s_2 \geq \cdots \geq s_m$

$L = (1, 2, \ldots, m)$: máquinas em ordem de velocidade

Primeira parte

$\Gamma(k) = (k-1)!$ $\qquad s_1 \geq \cdots \geq s_m \qquad L = (1, 2, \ldots, m)$

Para $k \in \{0, \ldots, c-1\}$, L_k é o maior prefixo de L de máquinas com carga $\geq k\text{opt}(J)$



Primeira parte

Vamos provar que:

- $|L_{c-1}| \geq 1$
- $|L_k| \geq (k+1)|L_{k+1}|$ para todo $0 \leq k \leq c-2$

Disso segue que $|L_0| \geq (c-1)! = \Gamma(c)$

Como $L_0 = L$ e $|L| = m$

- $m \geq \Gamma(c)$
- $c \leq \Gamma^{-1}(m)$

Resta provar a recorrência

Prova da recorrência

Vamos provar que:

- $|L_k| \geq (k+1)|L_{k+1}|$ para todo $0 \leq k \leq c-2$
- $|L_{c-1}| \geq 1$

Primeiro, mostremos que $|L_{c-1}| \geq 1$

Suponha que não, ou seja, que $L_{c-1} = \emptyset$

Então a carga $\ell_1 < (c-1)\text{opt}(J)$

Seja i tarefa em uma máquina com carga $\geq c\,\text{opt}(J)$

$$\ell_1 + \frac{w_i}{s_1} \;<\; (c-1)\,\text{opt}(J) + \text{opt}(J) \;=\; c\,\text{opt}(J)$$

Ou seja, i tem incentivo para mudar para a máquina 1

Contradição pois A é um equilíbrio

Prova de que $|L_k| \geq (k+1)|L_{k+1}|$

Seja A^* uma atribuição ótima (de makespan mínimo)

Lema: Se i é tal que $A(i) \in L_{k+1}$ então $A^*(i) \in L_k$

Prova: Se $L_k = L$, nada a provar

Seja q máquina de menor índice em $L \setminus L_k$

Temos que $\ell_q < k\,\mathsf{opt}(J)$

$\ell_{A(i)} \geq (k+1)\,\mathsf{opt}(J)$ pois i é tq $A(i) \in L_{k+1}$

Se $w_i \leq s_q\,\mathsf{opt}(J)$, então i tem incentivo para ir para q:

$$\ell_q + \frac{w_i}{s_q} < k\,\mathsf{opt}(J) + \mathsf{opt}(J) = (k+1)\mathsf{opt}(J) \leq \ell_{A(i)},$$

contradição

Prova de que $|L_k| \geq (k+1)|L_{k+1}|$

Para toda tarefa i tq $A(i) \in L_{k+1}$:

- $\ell_{A(i)} \geq (k+1)\,\mathrm{opt}(J)$
- $w_i > s_q\,\mathrm{opt}(J)$

Por contradição, suponha que $j = A^*(i) \notin L_k$

Então, como $s_j \leq s_q$, a carga de j em A^* seria

$$\geq \frac{w_i}{s_j} > \frac{s_q\,\mathrm{opt}(J)}{s_j} \geq \mathrm{opt}(J)$$

contradição $\square$

Prova de que $|L_k| \geq (k+1)|L_{k+1}|$

Lema: Se i é tal que $A(i) \in L_{k+1}$ então $A^*(i) \in L_k$ $\square$

Pela definição de L_{k+1}, para cada j em L_{k+1},

$$\ell_j = \sum_{i:A(i)=j} \frac{w_i}{s_j} \;\geq\; (k+1)\,\mathsf{opt}(J),$$

e o total de peso atribuído por A a máquinas em L_{k+1} é

$$\geq \sum_{j \in L_{k+1}} (k+1)\,\mathsf{opt}(J)\,s_j$$

Pelo Lema, A^* atribui tudo isso a L_k sem estourar $\mathsf{opt}(J)$, logo

$$\sum_{j \in L_{k+1}} (k+1)\,\mathsf{opt}(J)\,s_j \leq \sum_{j \in L_k} \mathsf{opt}(J)\,s_j,$$

que implica que $\sum_{j \in L_k} s_j \;\geq\; \sum_{j \in L_{k+1}} (k+1) s_j$

Prova de que $|L_k| \geq (k+1)|L_{k+1}|$

Relembrando:

$$\sum_{j \in L_k} s_j \;\geq\; \sum_{j \in L_{k+1}} (k+1) s_j$$

Seja s^* a velocidade da máquina mais lenta em L_{k+1}

$$\begin{aligned}
(|L_k| - |L_{k+1}|)s^* = |L_k \setminus L_{k+1}|s^* &= \sum_{j \in L_k \setminus L_{k+1}} s^* \\
&\geq \sum_{j \in L_k \setminus L_{k+1}} s_j = \sum_{j \in L_k} s_j - \sum_{j \in L_{k+1}} s_j \\
&\geq \sum_{j \in L_{k+1}} k s_j \geq \sum_{j \in L_{k+1}} k s^* = k|L_{k+1}|s^*
\end{aligned}$$

de onde concluímos que $|L_k| \geq (k+1)|L_{k+1}|$ $\square$

Caso de máquinas relacionadas

Teorema: Para o jogo de balanceamento de carga em m máquinas relacionadas, $\text{PoA}(m) = \Theta\left(\frac{\lg m}{\lg \lg m}\right)$

Primeira parte:

Mostrar que para jogos $J = (n, m, w, s)$ e atribuições $A : [n] \to [m]$ em equilíbrio vale que

$$\text{custo}(A) = \text{O}\left(\frac{\lg m}{\lg \lg m}\right) \text{opt}(J)$$

Segunda parte:

Apresentar jogo $J = (n, m, w, s)$ e atribuição $A : [n] \to [m]$ em equilíbrio tal que

$$\text{custo}(A) = \Omega\left(\frac{\lg m}{\lg \lg m}\right) \text{opt}(J)$$

Segunda parte

Lema: Existe um jogo $J = (n, m, w, s)$ e atribuição $A : [n] \to [m]$ em equilíbrio tal que

$$\text{custo}(A) \;\geq\; \frac{1}{2}\Big(\Gamma^{-1}(m) - 2 - \text{o}(1)\Big)\text{opt}(J)$$

Prova: Seja $q = \lfloor \Gamma^{-1}(m/3) - 1 \rfloor$ e sejam $G_0, \ldots, G_q$ grupos disjuntos de máquinas

Grupo G_k:

- $\frac{q!}{k!}$ máquinas
- com velocidade 2^k
- cada uma com k tarefas (atribuídas por A)
- de peso 2^k
- Carga de cada máquina: k

Segunda parte

Como

$$\sum_{k=0}^{\infty} \frac{1}{k!} = e < 3 \qquad q = \lfloor \Gamma^{-1}(m/3) - 1 \rfloor$$

Total de máquinas:

$$\sum_{k=0}^{q} |G_k| = \sum_{k=0}^{q} \frac{q!}{k!} = q! \sum_{k=0}^{q} \frac{1}{k!} < 3\,\Gamma(q+1) \leq m$$

Complete m com máquinas sem grupo e vazias de velocidade 1 (como as de G_0)

A está em equilíbrio?

Tarefa em máquina de G_k não quer mudar para máquina de G_j

- Se $j \geq k$
  - a máquina tem carga $j \geq k$
- Se $j < k$,
  - como $2^t \geq t + 1$ pra todo $t \geq 1$ temos que

$$j + \frac{2^k}{2^j} = j + 2^{k-j} \geq j + (k - j + 1) = k + 1$$

Portanto, A está em equilíbrio

Jogo e equilíbrio

A é um equilíbrio de custo social q

Vamos mostrar que $\text{opt}(J) \leq 2$

Considere atribuição de tarefas de A^*:

- As de G_q serão atribuídas as máquinas de G_{q-1}
- As de G_{q-1} serão atribuídas as máquinas de G_{q-2}
- Assim sucessivamente...

Número de tarefas que A atribuiu a G_k:

$$k|G_k| = \frac{q!k}{k!} = \frac{q!}{(k-1)!} = |G_{k-1}|$$

Uma tarefa em cada máquina, com custo de $\frac{2^k}{2^{k-1}} = 2$

Conclusão

Preço da anarquia deste jogo: $\geq \frac{q}{2} = \frac{\lfloor \Gamma^{-1}(m/3)-1 \rfloor}{2}$

Como $\Gamma^{-1}(m) = \Theta\left(\frac{\lg m}{\lg\lg m}\right)$, existem c e m_0 tq

$$\Gamma^{-1}(m) \geq c\left(\frac{\lg m}{\lg\lg m}\right),$$

para $m \geq m_0$

Então, para $m \geq 3m_0$, o preço da anarquia é

$$\geq \frac{\lfloor \Gamma^{-1}(m/3) - 1 \rfloor}{2} \geq \frac{\lfloor c\left(\frac{\lg m/3}{\lg\lg m/3}\right) - 1 \rfloor}{2} = \Omega\left(\frac{\lg m}{\lg\lg m}\right)$$

Tempo de convergência

No caso de máquinas relacionadas, não se conhece resultado como o para máquinas idênticas

Mas podemos computar um equilíbrio eficientemente

Tempo de convergência

Algoritmo LPT (*largest processing time*):

- Atribua tarefas em ordem decrescente de peso
- pondo-as em máquinas que minimizem o seu custo

Tempo de convergência

Teorema: A atribuição calculada por LPT é um equilíbrio

Prova: Por indução no número de tarefas (colocadas)

Sejam:

- t a última tarefa colocada
- j^* a máquina a que foi atribuída

Apenas tarefas alocadas a j^* podem estar insatisfeitas

Seja $i < t$ uma tarefa alocada a j^*

Para toda máquina j

$$\frac{\displaystyle\sum_{k:\ A(k)=j^*} w_k}{s_{j^*}} \leq \frac{w_t + \displaystyle\sum_{k:\ A(k)=j} w_k}{s_j} \leq \frac{w_i + \displaystyle\sum_{k:\ A(k)=j} w_k}{s_j}$$

Logo i está satisfeita e a atribuição está em equilíbrio $\square$

Recapitulação

- Jogos de balanceamento de carga
  - com máquinas idênticas
  - com máquinas relacionadas
- Ambas as versões tem equilíbrio puro
- Preço da estabilidade de ambos é 1
- Preço da anarquia:
  - máquinas idênticas: $2 - \frac{2}{m+1}$
  - máquinas relacionadas: $\Theta\left(\frac{\lg m}{\lg \lg m}\right)$
- Tempo de convergência:
  - máquinas idênticas: convergência em n passos
  - máquinas relacionadas:
    - não conhecemos uma política boa
    - existe algoritmo eficiente que encontra equilíbrio

Estratégias mistas

Até o momento, consideramos apenas equilíbrios puros

- Vamos considerar agora equilíbrios mistos

Uma estratégia mista para o jogador i é um vetor p_i onde:

- p_i^j é a probabilidade da tarefa i ser alocada na máquina j
- Note que, A se torna uma atribuição aleatória

Seja x_i^j a variável aleatória binária que indica se a tarefa i é alocada na máquina j

$p_i^j = \mathbb{P}[x_i^j = 1]$ ou equivalentemente $p_i^j = \mathbb{P}[A(i) = j]$

Estratégias mistas

Assim, a carga esperada da máquina j é

$$\mathbb{E}\left[\ell_j\right] = \mathbb{E}\left[\sum_{i\in[n]} \frac{w_i x_i^j}{s_j}\right] = \sum_{i\in[n]} \frac{w_i \mathbb{E}\left[x_i^j\right]}{s_j} = \sum_{i\in[n]} \frac{w_i p_i^j}{s_j}$$

$P = (p_i^j)_{i\in[n],j\in[m]}$ é um perfil de estratégias

O custo de P é o makespan esperado, isto é,

$$\text{custo}(P) = \mathbb{E}[\text{custo}(A)] = \mathbb{E}\left[\max_{j\in[m]}(\ell_j)\right]$$

Estratégias mistas

O custo de uma máquina j para a tarefa i é $c_i^j = \mathbb{E}[\ell_j | A(i) = j]$

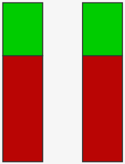
Para todo perfil de estratégias P, vale que

$$c_i^j = \frac{w_i + \sum_{k \neq i} w_k p_k^j}{s_j} = \mathbb{E}(\ell_j) + (1 - p_i^j)\frac{w_i}{s_j}$$

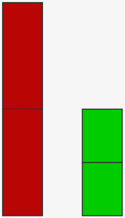
Proposição: Um perfil de estratégias P é um equilíbrio de Nash se e somente se para todo jogador i e para toda máquina j, se $p_i^j > 0$ então, para toda máquina k vale que $c_i^j \leq c_i^k$

Preço da anarquia para equilíbrio puro

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



Makespan mínimo: 3



Equilíbrio com makespan 4

Preço da anarquia pelo menos 4/3

Preço da anarquia para equilíbrio misto

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

- $p_i^j = 1/2$ para toda tarefa i e máquina j

Qual é a carga esperada de uma máquina?

$$\mathbb{E}[\ell_j] = \sum_{1 \le i \le 4} w_i p_i^j = 2 \times 2 \times \frac{1}{2} + 2 \times 1 \times \frac{1}{2} = 3$$

Qual é o custo para uma tarefa de peso 2?

$$c_i^j = \mathbb{E}[\ell_j] + (1 - p_i^j) w_i = 3 + \frac{1}{2} \times 2 = 4$$

Qual é o custo para uma tarefa de peso 1?

$$c_i^j = \mathbb{E}[\ell_j] + (1 - p_i^j) w_i = 3 + \frac{1}{2} \times 1 = 3.5$$

Note que P é um equilíbrio misto

Preço da anarquia para equilíbrio misto

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

- $p_i^j = 1/2$ para toda tarefa i e máquina j

Qual é o makespan esperado?

- Existem 4 formas do makespan ser 3
- Existem 6 formas do makespan ser 4
- Existem 4 formas do makespan ser 5
- Existem 2 formas do makespan ser 6

Assim,

$$\begin{aligned}\text{custo}(P) &= \mathbb{E}[\text{custo}(A)] \\ &= \frac{1}{16}\left(3 \times 4 + 4 \times 6 + 5 \times 4 + 6 \times 2\right) = 4.25\end{aligned}$$

Que é pior do que o equilíbrio puro que tínhamos visto (com custo 4)

Resultados para equilíbrio misto

Teorema: Para todo $m \in \mathbb{N}$, existe uma instância J do jogo do balanceamento de carga com m máquinas idênticas e $n = m$ tarefas que tem um equilíbrio de Nash com perfil de estratégias P onde

$$\text{custo}(P) = \Omega\left(\frac{\lg m}{\lg \lg m}\right) \text{opt}(J)$$

Teorema: Seja J uma instância do jogo de balanceamento de carga com n tarefas e m máquinas idênticas. Seja P qualquer perfil de estratégias que forme um equilíbrio de Nash. Vale que

$$\text{custo}(P) = O\left(\frac{\lg m}{\lg \lg m}\right) \text{opt}(J)$$

Resultados para equilíbrio misto

Teorema: Para todo $m \in \mathbb{N}$, existe uma instância J do jogo do balanceamento de carga com m máquinas idênticas e $n \leq m$ tarefas que tem um equilíbrio de Nash com perfil de estratégias P onde

$$\text{custo}(P) = \Omega\left(\frac{\lg m}{\lg \lg \lg m}\right) \text{opt}(J)$$

Teorema: Seja J uma instância do jogo de balanceamento de carga com n tarefas e m máquinas relacionadas. Seja P qualquer perfil de estratégias que forme um equilíbrio de Nash. Vale que

$$\text{custo}(P) = O\left(\frac{\lg m}{\lg \lg \lg m}\right) \text{opt}(J)$$

Mecanismos de Coordenação

"Mecanismos" que levem a bons valores de PoA

- Estabelecer regras no jogo para diminuir o PoA
- Custo para jogadores na mesma máquina pode diferir
- Custo $c^j : 2^{[n]} \times \{1, \ldots, n\} \to \mathbb{R}$
  - $c^j(S, i)$ é o tempo de término da tarefa i na máquina j quando o conjunto S de tarefas é atribuído à maquina j
  - depende apenas dos pesos das tarefas
  - e da ordem (pré-fixada) das tarefas
- Mecanismo de Coordenação: $\mathcal{C} = (c^1, c^2, \ldots, c^m)$
- As funções de custo são de conhecimento dos jogadores
- No modelo anterior, custo $c^j(S, i) = \sum_{i' \in S} w_{ij}$
  - não diferencia tarefas na mesma máquina

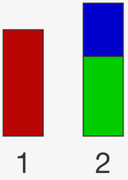
Mecanismos de Coordenação

Exemplo 1:

- Duas máquinas uniformes
- três tarefas tal que $w_1 \succ w_2 \succ w_3$
  - ordenação decrescente pelo peso, com desempate pela ordem de entrada
- Máquina 1 executa em ordem decrescente de pesos
- Máquina 2 executa em ordem crescente de pesos

Em um equilíbrio:

- Tarefa 1 escolhe máquina 1
- Tarefa 3 escolhe máquina 2
- Tarefa 2 escolhe máquina 2
- A configuração é ótima



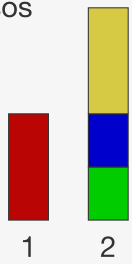
Mecanismos de Coordenação

Exemplo 2:

- Duas máquinas uniformes e quatro tarefas tal que
  - $w_1 = 2 + \varepsilon$, $w_2 = 2$, $w_3 = 1 - \varepsilon$ e $w_4 = 1$
- Máquina 1 executa em ordem decrescente de pesos
- Máquina 2 executa em ordem crescente de pesos

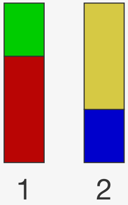
Custo social do equilíbrio: 4

- Tarefa 1 escolhe máquina 1
- Tarefa 3, 4 e 2 escolhem máquina 2



Configuração ótima: Custo social 3

- Tarefas 1 e 3 na máquina 1
- Tarefas 2 e 4 na máquina 2



$$\text{PoA} = \frac{4 - \varepsilon}{3}$$

Mecanismos de Coordenação

Mecanismo LPT*:

- Cada máquina escalona na ordem decrescente, usando $\succ$
- Cada máquina j acrescenta um atraso na tarefa, se necessário, para que termine em tempo $t \equiv j \mod m + 1$
- Atraso pode ser escolhido tão pequeno quanto se queira, usando discretização em ε/mn

Teorema: O Mecanismo LPT*, para máquinas uniformes, possui preço da anarquia limitado a $4/3 - \varepsilon$. Além disso, essa razão é justa.

Teorema: O Mecanismo LPT*, para máquinas relacionadas, possui preço da anarquia entre $1{,}54$ e $1 + \sqrt{3}/3 \approx 1{,}5773$.