

Algoritmos Aproximativos - Definição padrão

Quatro ingredientes de um problema de otimização

- ▶ **Conjunto de Instâncias**
- ▶ **Conjunto de Soluções** $Sol(I)$ para cada instância I
- ▶ **Valor** $val(S) > 0$ para cada solução $S \in Sol(I)$
- ▶ **Tipo**: minimização ou maximização

Objetivo: Dada uma instância I , encontrar uma **solução ótima**
 $S \in Sol(I)$: solução com valor mínimo ou máximo, dependendo do tipo.
Valor ótimo $opt(I)$ é o valor de uma solução ótima de uma instância I .

Um algoritmo $\mathcal{A}$ para um problema de otimização é **α -aproximativo**
se produz uma solução $\mathcal{A}(I)$ tq $val(\mathcal{A}(I)) \leq \alpha \cdot opt(I)$ (se **min.** ($\alpha \geq 1$))
ou $val(\mathcal{A}(I)) \geq \alpha \cdot opt(I)$ (se **max.** ($\alpha \leq 1$))

DEFINIÇÃO PADRÃO

Algoritmos Aproximativos - Def. padrão versus alternativa

Quatro ingredientes de um problema de otimização

- ▶ **Conjunto de Instâncias**
- ▶ **Conjunto de Soluções** $Sol(I)$ para cada instância I
- ▶ **Valor** $val(S) > 0$ para cada solução $S \in Sol(I)$
- ▶ **Tipo**: minimização ou maximização

DEFINIÇÃO: Dada uma instancia I e uma solução $S \in Sol(I)$, seja

$$\mathcal{R}(I, S) = \max \left\{ \frac{opt(I)}{val(S)}, \frac{val(S)}{opt(I)} \right\} \geq 1 \quad (\text{performance ratio})$$

(razão de aproximação)

Um algoritmo $\mathcal{A}$ para um problema de otimização é **$\alpha_{\geq 1}$ -aproximativo** se produz uma solução $\mathcal{A}(I)$ tal que $\mathcal{R}(I, \mathcal{A}(I)) \leq \alpha$ para toda instância I .

DEFINIÇÃO ALTERNATIVA. Embora possa parecer confuso com a “definição antiga”, sempre ficará claro no contexto se estamos lidando com uma definição ou outra. Essa divergência de definições ocorre entre pesquisadores da área (Vazirani (antiga), Ausiello (nova)). No tópico das reduções que preservam aproximação entre problemas de otimização, a definição nova é mais útil. De qualquer modo, elas são intercambiáveis e é importante conhecer e saber lidar com ambas.

Algoritmo Aproximativo (Def. padrão versus alternativa)

DEFINIÇÃO: Dada uma instancia I e uma solução $S \in Sol(I)$, seja

$$\mathcal{R}(I, S) = \max \left\{ \frac{opt(I)}{val(S)}, \frac{val(S)}{opt(I)} \right\} \geq 1 \quad (\text{performance ratio})$$

(razão de aproximação)

Um algoritmo $\mathcal{A}$ para um problema de otimização é **$\alpha_{\geq 1}$ -aproximativo** se produz uma solução $\mathcal{A}(I)$ tal que $\mathcal{R}(I, \mathcal{A}(I)) \leq \alpha$ para toda instância I .

Ou seja, um algoritmo tem **fator de aproximação** α se sempre produz uma **solução** com **razão de aproximação** $\leq \alpha$ (nessa definição alternativa).

- ▶ Problemas de minimização $\Rightarrow$ Definições coincidem (mesmo fator de aproximação)
- ▶ Problemas de maximização $\Rightarrow$ Definições divergem (fatores de aproximação inversos)
- ▶ Alg. **0.5-aprox** MaxSAT-Johnson $\Rightarrow$ **2 - aprox** (definição alternativa)
- ▶ Alg. **0.75-aprox** MaxSAT-GW $\Rightarrow$ **(1.333)-aprox** (definição alternativa)
- ▶ Alg. **0.878-aprox** MaxCut-GW $\Rightarrow$ **(1.138)-aprox** (definição alternativa)
- ▶ Alg. **0.931-aprox** Max2SAT-FG $\Rightarrow$ **(1.074)-aprox** (definição alternativa)

Conclusão (até agora), assumindo $P \neq NP$

- ▶ **Mochila:** FPTAS $O(n^3/\varepsilon)$.
- ▶ **Escalonamento - num m fixo máquinas:** FPTAS $O((n/\varepsilon)^{m+1})$.
- ▶ **Escalon - num qquer máquinas:** PTAS $O(n^{2 \log_{1+\varepsilon}(1/\varepsilon)} \cdot \log_2 \frac{1}{\varepsilon})$.
- ▶ **Bin Packing:** PTAS assintótico. 1.7-aprox. $(1.5 - \varepsilon)$ -inaprox.
- ▶ **TSP-Euclidiano:** PTAS.
- ▶ **TSP-Métrico:** 1.5-aproximativo.
- ▶ **TSP:** $\alpha(n)$ -inaprox. para qualquer função poli $\alpha(n)$.
- ▶ **Set Cover:** $(\ln n + 1)$ -aprox., mas $(\ln n - \varepsilon)$ -inaprox. (Moshkovitz'15)
- ▶ **MaxClique e MinColor:** n -aproximáveis, mas $n^{1-\varepsilon}$ -inaproximáveis em tempo polinomial (Zuckerman'06)

Aqui relembramos que, na notação anterior, a frase acima quer dizer que MaxClique é $(1/n)$ -aproximável, mas $n^{\varepsilon-1}$ -inaproximável em tempo polinomial para qualquer $\varepsilon > 0$ (Zuckerman'06)

Classes de Aproximabilidade

NPO: Problemas de otimização tais que toda **solução tem um tamanho limitado** por um polinômio no tamanho da instância e nos quais podemos, em tempo polinomial, **reconhecer instâncias e soluções** de instâncias, bem como **calcular valores** de soluções.

PO: Problemas NPO com algoritmo exato polinomial.

APX: Problemas NPO com algoritmo poli α -aprox. para α constante.

poly-APX: Problemas NPO com algoritmo de tempo poli $\alpha(n)$ -aprox. para função **polinomial** $\alpha(n) = O(n^k)$, onde $k = \text{const}$ e $n = \langle I \rangle$.

log-APX: idem, mas função **logarítmica** $\alpha(n) = O(\log n)$.

PTAS: Problemas NPO que tem PTAS (**esquema de aproximação em tempo polinomial**): algoritmo $\mathcal{A}_\varepsilon$ polinomial em $\langle I \rangle$ que retorna solução satisf. $\text{val}(\mathcal{A}(I)) = (1 \pm \varepsilon)\text{opt}(I)$, p/ cada racional $\varepsilon > 0$ e instância I .

FPTAS: Problemas NPO que tem FPTAS (**esquema de aproximação em tempo completamente polinomial**): PTAS polinomial também em $1/\varepsilon$.

PO $\subseteq$ FPTAS $\subseteq$ PTAS $\subseteq$ APX $\subseteq$ log-APX $\subseteq$ poly-APX $\subseteq$ NPO

Classes de Aproximabilidade

NPO: Problemas de otimização tais que toda **solução tem um tamanho limitado** por um polinômio no tamanho da instância e nos quais podemos, em tempo polinomial, **reconhecer instâncias e soluções** de instâncias, bem como **calcular valores** de soluções.

PO: Problemas NPO com algoritmo exato polinomial.

APX: Problemas NPO com algoritmo poli α -aprox. para α constante.

poly-APX: Problemas NPO com algoritmo de tempo poli $\alpha(n)$ -aprox. para função **polinomial** $\alpha(n) = O(n^k)$, onde $k = \text{const}$ e $n = \langle I \rangle$.

log-APX: idem, mas função **logarítmica** $\alpha(n) = O(\log n)$.

PTAS: Problemas NPO que tem PTAS (**esquema de aproximação em tempo polinomial**): algoritmo $\mathcal{A}_\varepsilon$ polinomial em $\langle I \rangle$ que retorna solução satisf. $\mathcal{R}(I, \mathcal{A}_\varepsilon(I)) \leq 1 + \varepsilon$, p/ cada racional $\varepsilon > 0$ e instância I .

FPTAS: Problemas NPO que tem FPTAS (**esquema de aproximação em tempo completamente polinomial**): PTAS polinomial também em $1/\varepsilon$.

$$\text{PO} \subseteq \text{FPTAS} \subseteq \text{PTAS} \subseteq \text{APX} \subseteq \text{log-APX} \subseteq \text{poly-APX} \subseteq \text{NPO}$$

Classes de Aproximabilidade - Exemplos

NPO: Caixeiro Viajante (TSP)

PO: Menor Caminho, Árvore Geradora Mínima (MST).

APX: Bin Packing, Vertex Cover, Caixeiro Viajante Métrico (TSPM).

poly-APX: MaxClique, MinColor, pois têm alg. n -aprox.

log-APX: Set Cover, pois tem alg. $(\ln n + 1)$ -aprox.

PTAS: Escalonamento Mínimo, Caixeiro Viajante Euclidiano (TSPE).

FPTAS: Problema da Mochila.

PO $\subseteq$ FPTAS $\subseteq$ PTAS $\subseteq$ APX $\subseteq$ log-APX $\subseteq$ poly-APX $\subseteq$ NPO

Classes de Aproximabilidade - Relações, se $P \neq NP$

NPO: TSP $\notin$ poly-APX (já visto)

PO: Menor Caminho, Árvore Geradora Mínima (MST).

APX: Bin Packing $\notin$ PTAS, pois é $(1.5 - \varepsilon)$ -inaproximável.

poly-APX: MaxClique, MinColor $\notin$ log-APX, pois são $n^{1-\varepsilon}$ -inaprox. (Zuckerman'06)

log-APX: Set Cover $\notin$ APX, pois é $(1-\varepsilon) \ln n$ -inaprox. (Moshkovitz'15)

PTAS: Escalonamento Mínimo $\notin$ FPTAS.

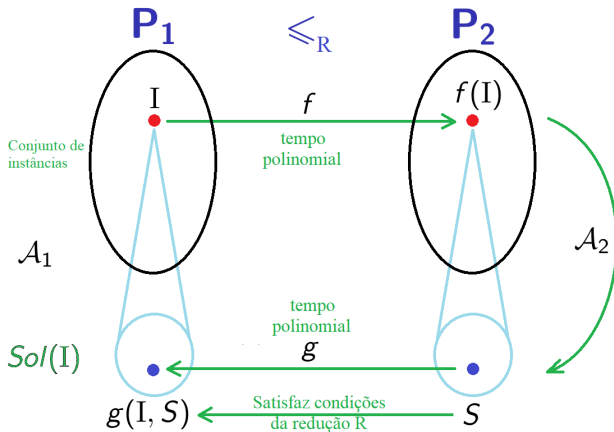
FPTAS: Problema da Mochila $\notin$ PO, pois é NP-Difícil.

PO $\subsetneq$ FPTAS $\subsetneq$ PTAS $\subsetneq$ APX $\subsetneq$ log-APX $\subsetneq$ poly-APX $\subsetneq$ NPO

Redução entre Problemas NPO P_1 e P_2

Redução $P_1 \leq_R P_2$: par (f, g)

- ▶ f e g : funções computáveis tempo polin, no tam de suas instâncias
- ▶ Instância I de $P_1 \Rightarrow f(I)$ é uma instância de P_2
- ▶ $g(I, S)$ é solução de I em $P_1 \Leftarrow$ Solução S de $f(I)$ em P_2



Redução entre Problemas NPO P_1 e P_2

- f e g : funções computáveis tempo polin. no tam de suas instâncias
- Instância I de $P_1 \Rightarrow f(I)$ é uma instância de P_2
- $g(I, S)$ é solução de I em $P_1 \Leftarrow$ Solução S de $f(I)$ em P_2
- tal que, para toda instância I de P_1 e toda solução S de $f(I)$ em P_2

$P_1 \leq_{\text{strict}} P_2$: Redução strict (f, g)

$$\blacktriangleright \mathcal{R}_{P_2}(f(I), S) \leq r \Rightarrow \mathcal{R}_{P_1}(I, g(I, S)) \leq r$$

$P_1 \leq_c P_2$: Redução contínua ($f, g, \alpha_{\geq 1}$)

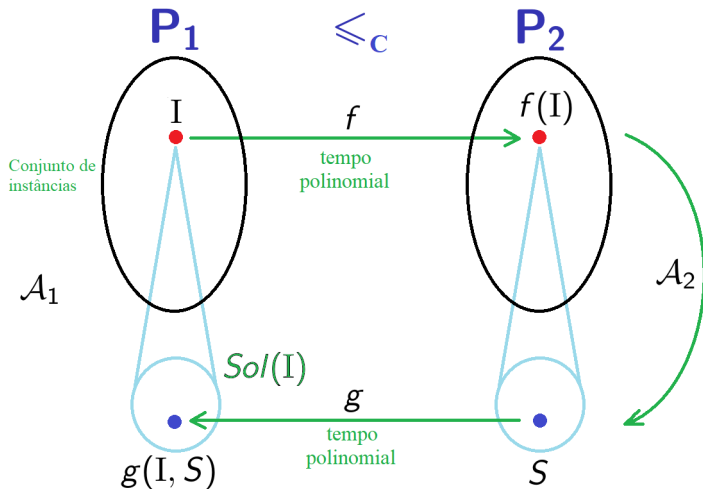
$$\blacktriangleright \mathcal{R}_{P_2}(f(I), S) \leq r \Rightarrow \mathcal{R}_{P_1}(I, g(I, S)) \leq \alpha \cdot r$$

$P_1 \leq_E P_2$: Redução E (relativa ao "erro") ($f, g, \alpha_{\geq 1}$)

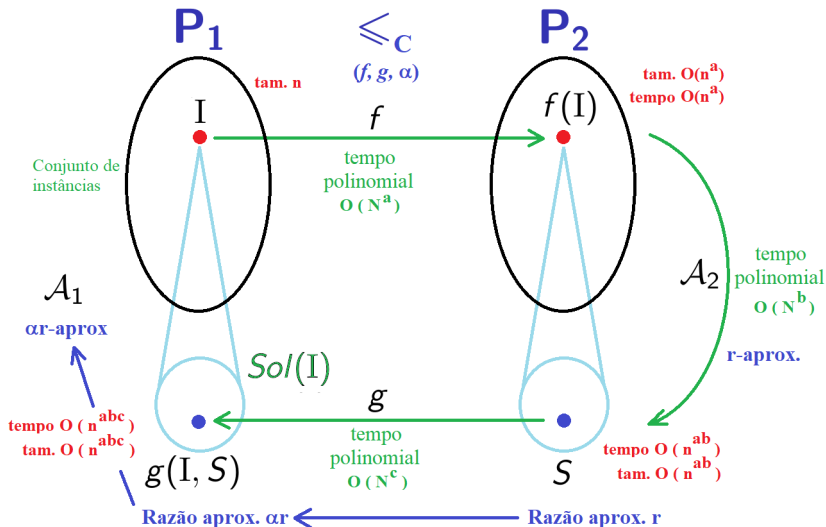
$$\blacktriangleright \mathcal{R}_{P_2}(f(I), S) \leq r \Rightarrow \mathcal{R}_{P_1}(I, g(I, S)) \leq 1 + \alpha \cdot (r - 1)$$

$$\blacktriangleright \mathbf{P_1} \leq_{\text{strict}} \mathbf{P_2} \Rightarrow \mathbf{P_1} \leq_E \mathbf{P_2} \Rightarrow \mathbf{P_1} \leq_c \mathbf{P_2}$$

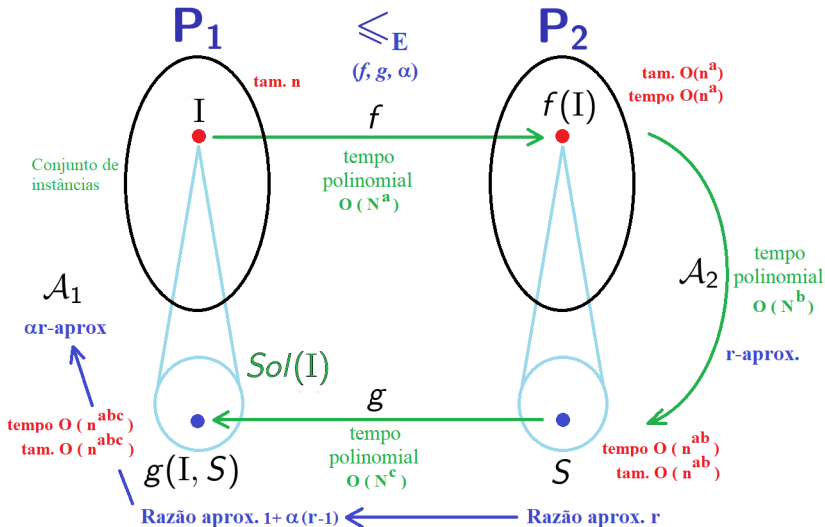
PROVA: $P_1 \leq_c P_2$ e $P_2 \in APX \Rightarrow P_1 \in APX$



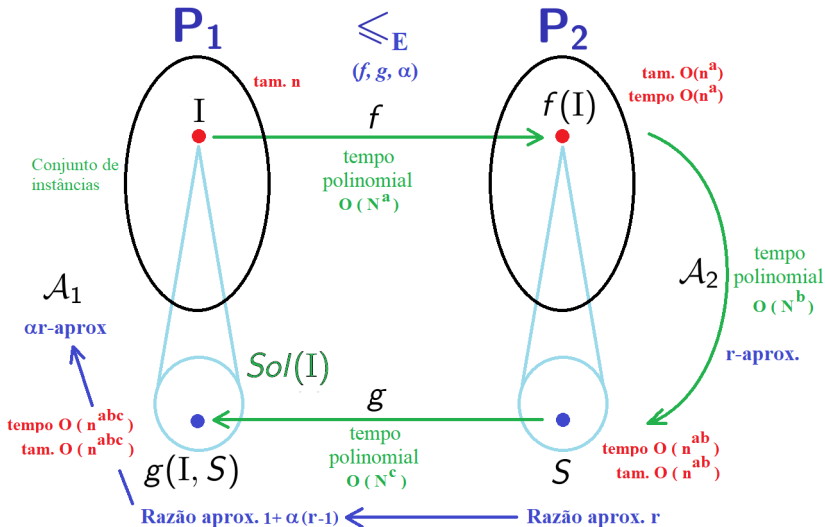
PROVA: $P_1 \leq_c P_2$ e $P_2 \in APX \Rightarrow P_1 \in APX$



PROVA: $P_1 \leq_E P_2$ e $P_2 \in PTAS \Rightarrow P_1 \in PTAS$



PROVA: $P_1 \leq_E P_2$ e $P_2 \in \text{FPTAS} \Rightarrow P_1 \in \text{FPTAS}$



Redução entre Problemas NPO P_1 e P_2

► $P_1 \leq_C P_2$ e $P_2 \in APX \Rightarrow P_1 \in APX$

• $P_1 \leq_C P_2$ e $P_1 \notin APX \Rightarrow P_2 \notin APX$

► $P_1 \leq_E P_2$ e $P_2 \in PTAS \Rightarrow P_1 \in PTAS$

• $P_1 \leq_E P_2$ e $P_1 \notin PTAS \Rightarrow P_2 \notin PTAS$

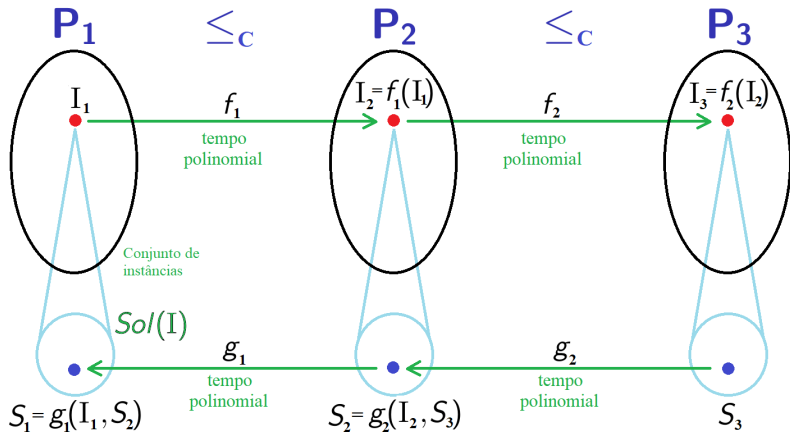
► $P_1 \leq_E P_2$ e $P_2 \in FPTAS \Rightarrow P_1 \in FPTAS$

• $P_1 \leq_E P_2$ e $P_1 \notin FPTAS \Rightarrow P_2 \notin FPTAS$

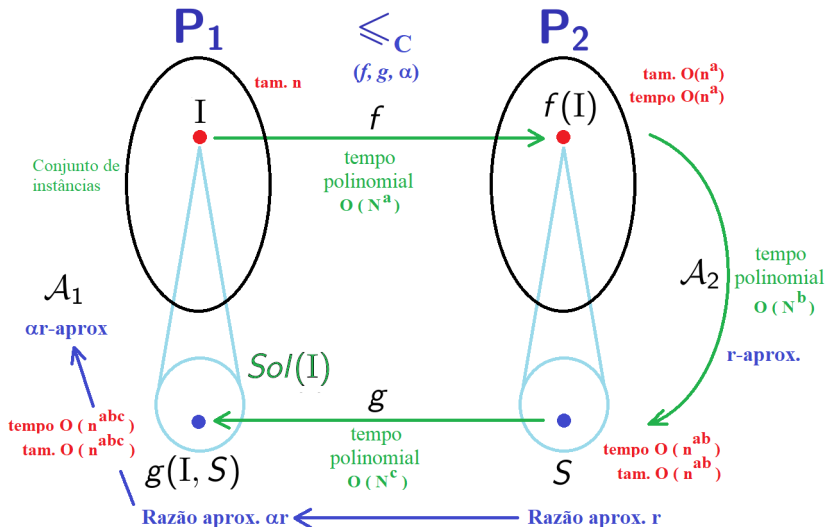
► $P_1 \leq_C P_2$ e $P_2 \leq_C P_3 \Rightarrow P_1 \leq_C P_3$

► $P_1 \leq_E P_2$ e $P_2 \leq_E P_3 \Rightarrow P_1 \leq_E P_3$

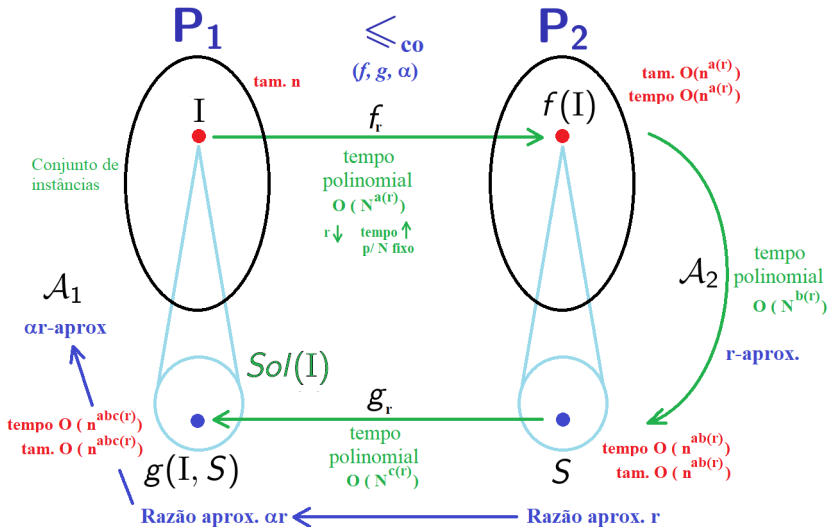
PROVA: $P_1 \leq_c P_2$ e $P_2 \leq_c P_3 \Rightarrow P_1 \leq_c P_3$



PROVA: $P_1 \leq_c P_2$ e $P_2 \in APX \Rightarrow P_1 \in APX$



PROVA: $P_1 \leq_{co} P_2$ e $P_2 \in APX \Rightarrow P_1 \in APX$



Redução entre Problemas NPO P_1 e P_2

Redução (f, g) : para todo $r \geq 1$ racional

- f_r e g_r : funções computáveis tempo polin. no tam de suas instâncias
- Instância I de $P_1 \Rightarrow f_r(I)$ é uma instância de P_2
- $g_r(I, S)$ é solução de I em $P_1 \Leftarrow$ Solução S de $f_r(I)$ em P_2
- tal que, para toda instância I de P_1 e toda solução S de $f_r(I)$ em P_2

$P_1 \leq_{co} P_2$: Redução Co (contínua forte) $(f, g, \alpha_{\geq 1})$

$$\blacktriangleright \mathcal{R}_{P_2}(f_r(I), S) \leq r \Rightarrow \mathcal{R}_{P_1}(I, g_r(I, S)) \leq \alpha \cdot r$$

$P_1 \leq_{ap} P_2$: Redução AP $(f, g, \alpha_{\geq 1})$

$$\blacktriangleright \mathcal{R}_{P_2}(f_r(I), S) \leq r \Rightarrow \mathcal{R}_{P_1}(I, g_r(I, S)) \leq 1 + \alpha \cdot (r - 1)$$

$$\begin{array}{ccccc} P_1 \leq_{\text{strict}} P_2 & \Rightarrow & P_1 \leq_E P_2 & \Rightarrow & P_1 \leq_C P_2 \\ & & \Downarrow & & \Downarrow \\ & & P_1 \leq_{ap} P_2 & \Rightarrow & P_1 \leq_{co} P_2 \end{array}$$

Redução entre Problemas NPO P_1 e P_2

- ▶ $P_1 \leq_{\text{co}} P_2$ e $P_2 \in \text{APX}$ $\Rightarrow P_1 \in \text{APX}$
- ▶ $P_1 \leq_{\text{co}} P_2$ e $P_2 \in \text{poly-APX}$ $\Rightarrow P_1 \in \text{poly-APX}$
- ▶ $P_1 \leq_{\text{co}} P_2$ e $P_2 \in \text{log-APX}$ $\Rightarrow P_1 \in \text{log-APX}$
- ▶ $P_1 \leq_{\text{ap}} P_2$ e $P_2 \in \text{PTAS}$ $\Rightarrow P_1 \in \text{PTAS}$
- ▶ $P_1 \leq_{\text{co}} P_2$ e $P_2 \leq_{\text{co}} P_3 \Rightarrow P_1 \leq_{\text{co}} P_3$
- ▶ $P_1 \leq_{\text{ap}} P_2$ e $P_2 \leq_{\text{ap}} P_3 \Rightarrow P_1 \leq_{\text{ap}} P_3$
- ▶ $P_1 \leq_{\text{strict}} P_2 \Rightarrow P_1 \leq_E P_2 \Rightarrow P_1 \leq_c P_2$
 $\Downarrow$
 $P_1 \leq_{\text{ap}} P_2 \Rightarrow P_1 \leq_{\text{co}} P_2$
 $\Downarrow$
 $P_1 \leq_{\text{ptas}} P_2$

NPO-completude e APX-completude

NPO-difícil e APX-difícil

- ▶ Problema P_2 é **NPO-difícil** se $P_1 \leq_{co} P_2$, $\forall P_1 \in \text{NPO}$
- ▶ Problema P_2 é **poly-APX-difícil** se $P_1 \leq_{co} P_2$, $\forall P_1 \in \text{poly-APX}$
- ▶ Problema P_2 é **log-APX-difícil** se $P_1 \leq_{co} P_2$, $\forall P_1 \in \text{log-APX}$
- ▶ Problema P_2 é **APX-difícil** se $P_1 \leq_{ap} P_2$, $\forall P_1 \in \text{APX}$
- ▶ Problema P_2 é **APX-completo** se é APX e APX-difícil
- ▶ Problema P_2 é **NPO-completo** se é NPO e NPO-difícil

Conclusões (se $P \neq NP$)

- ▶ P_2 é **NPO-difícil** $\Rightarrow P_2 \notin \text{poly-APX}$
- ▶ P_2 é **poly-APX-difícil** $\Rightarrow P_2 \notin \text{log-APX}$
- ▶ P_2 é **log-APX-difícil** $\Rightarrow P_2 \notin \text{APX}$
- ▶ P_2 é **APX-difícil** $\Rightarrow P_2 \notin \text{PTAS}$

NPO-completude e APX-completude

- ▶ Problema P_2 é **NPO-difícil** se $P_1 \leq_{co} P_2$, $\forall P_1 \in \text{NPO}$
- ▶ Problema P_2 é **poly-APX-difícil** se $P_1 \leq_{co} P_2$, $\forall P_1 \in \text{poly-APX}$
- ▶ Problema P_2 é **log-APX-difícil** se $P_1 \leq_{co} P_2$, $\forall P_1 \in \text{log-APX}$
- ▶ Problema P_2 é **APX-difícil** se $P_1 \leq_{ap} P_2$, $\forall P_1 \in \text{APX}$

Conclusões

- ▶ $P_1 \leq_{co} P_2$ e $P_1 \in \text{NPO-difícil} \Rightarrow P_2 \in \text{NPO-difícil}$
- ▶ $P_1 \leq_{co} P_2$ e $P_1 \in \text{poly-APX-difícil} \Rightarrow P_2 \in \text{poly-APX-difícil}$
- ▶ $P_1 \leq_{co} P_2$ e $P_1 \in \text{log-APX-difícil} \Rightarrow P_2 \in \text{log-APX-difícil}$
- ▶ $P_1 \leq_{ap} P_2$ e $P_1 \in \text{APX-difícil} \Rightarrow P_2 \in \text{APX-difícil}$
- ▶ **MaxSAT**, BinPacking, MaxCut, MinCV, TSPM são **APX-completos**
[Papadimitriou, Yannakakis'91]
- ▶ **TSP** é **NPO-completo** [Orponen, Mannila'87]
- ▶ **Clique** é **poly-APX-completo** [Bazgan et al.'05]
- ▶ **Set Cover** é **log-APX-completo** [Escoffier, Paschos'06]

Classes de Aproximabilidade - Relações, se $P \neq NP$

NPO: TSP $\notin$ poly-APX (já visto)

PO: Menor Caminho, Árvore Geradora Mínima (MST).

APX: Bin Packing $\notin$ PTAS, pois é $(1.5 - \varepsilon)$ -inaproximável.

poly-APX: MaxClique, MinColor $\notin$ log-APX, pois são $n^{1-\varepsilon}$ -inaprox. (Zuckerman'06)

log-APX: Set Cover $\notin$ APX, pois é $(1-\varepsilon) \ln n$ -inaprox. (Moshkovitz'15)

PTAS: Escalonamento Mínimo $\notin$ FPTAS.

FPTAS: Problema da Mochila $\notin$ PO, pois é NP-Difícil.

PO $\subsetneq$ FPTAS $\subsetneq$ PTAS $\subsetneq$ APX $\subsetneq$ log-APX $\subsetneq$ poly-APX $\subsetneq$ NPO

Exemplo de Redução: Subset-Sum $\leq_{strict}$ Mochila

$$U = \{ 2, \overset{\bullet}{5}, 5, 8, \overset{\bullet}{11} \}$$
$$T = 17,9$$

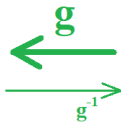
Objetivo: obter subconjunto de U com soma máxima $\leq T$



		1	2	3	4	5		
pesos	=	[	2,	5,	5,	8,	11	]
valores	=	[	2,	5,	5,	8,	11	]
capacidade	=	17,9						

Objetivo: Obter subconjunto de itens com valor máximo e peso $\leq$ capacidade

$$U' = \{ 5, 11 \}$$



$$\text{Itens} = \{2, 5\}$$

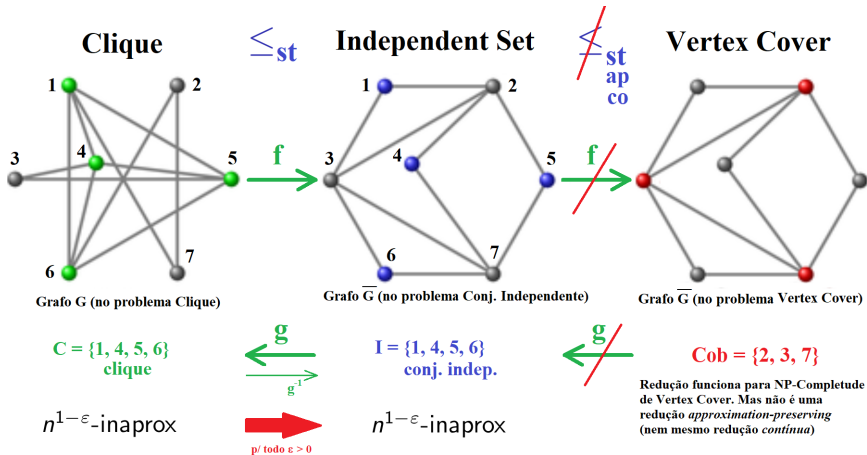
FPTAS



FPTAS

Exemplo de Redução: Clique $\leq_{strict}$ Independent Set

- ▶ Clique $\leq_{ap}$ Independent Set
- ▶ Independent Set $\leq_{ap}$ Clique



Exemplo de Redução: Set Packing $\leq_{strict}$ Independent Set

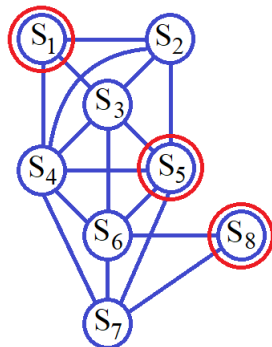
Set Packing

 $\leq_{st}$

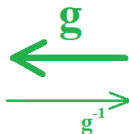
Independent Set

$U = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$

- $S_1 = \{1, 2, 3\}$
- $S_2 = \{1, 2, 3, 4\}$
- $S_3 = \{2, 3, 4, 5\}$
- $S_4 = \{3, 4, 5, 6\}$
- $S_5 = \{4, 5, 6\}$
- $S_6 = \{5, 6, 7, 8\}$
- $S_7 = \{6, 7, 8, 9\}$
- $S_8 = \{7, 8, 9\}$



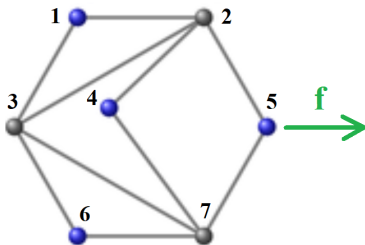
Pack = $\{S_1, S_5, S_8\}$



I = $\{S_1, S_5, S_8\}$

Exemplo de Redução: Independent Set $\leq_{strict}$ Set Packing

Independent Set $\leq_{st}$



Set Packing

$U = \{12, 13, 23, 24, 25, 36, 37, 47, 57, 67\}$

$S_1 = \{12, 13\}$

$S_2 = \{12, 23, 24, 25\}$

$S_3 = \{13, 23, 36, 37\}$

$S_4 = \{24, 47\}$

$S_5 = \{25, 57\}$

$S_6 = \{36, 67\}$

$S_7 = \{37, 47, 57, 67\}$

Grafo G (no problema Conj. Independente)

$I = \{1, 4, 5, 6\}$
conj. indep.

$n^{1-\varepsilon}$ -inaprox

$n = |V(G)|$



Pack = $\{S_1, S_4, S_5, S_6\}$

$n^{1-\varepsilon}$ -inaprox

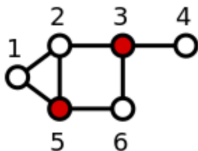
mesmo se interseção de 2 conjuntos
tenha tamanho no máximo 1

n = num. conjuntos



p/ todo $\varepsilon > 0$

Exemplo de Redução: Domination $\leq_{strict}$ Set Cover



$$U = \{1, 2, \dots, 6\}$$

$$S_1 = \{1, 2, 5\}$$

$$S_2 = \{1, 2, 3, 5\}$$

$$S_3 = \{2, 3, 4, 6\}$$

$$S_4 = \{3, 4\}$$

$$S_5 = \{1, 2, 5, 6\}$$

$$S_6 = \{3, 5, 6\}$$

$$D = \{3, 5\}$$



$$C = \{S_3, S_5\}$$

Grau máximo $\Delta = 3$
(no vértice 2, por exemplo)

Frequência máxima $= \Delta + 1 = 4$
(no elemento 2, por exemplo)

Algoritmo $(\Delta + 1)$ -aprox
Algoritmo $(\ln n + 1)$ -aprox



Algoritmo freq-aprox
Algoritmo $(\ln n + 1)$ -aprox

Exemplo de Redução: Set Cover $\leq_{strict}$ Domination

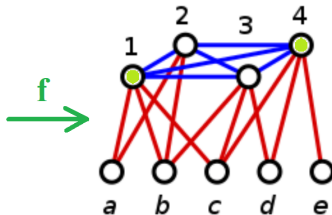
$$U = \{a, b, c, d, e\}$$

$$S_1 = \{a, b, c\}$$

$$S_2 = \{a, b\}$$

$$S_3 = \{b, c, d\}$$

$$S_4 = \{c, d, e\}$$



$$C = \{S_1, S_4\}$$

$$D = \{a, b, 4\}$$

$$D = \{1, 4\}$$



Podemos assumir que não há um elemento x em todos os conjuntos
 $N = |U|$

Não existe algoritmo $(\ln N - \varepsilon)$ - aprox polinomial

Grafo split (clique + indep)
 $N = \alpha(G) = \text{tam. maior conjunto independente}$

Não existe algoritmo $(\ln \alpha(G) - \varepsilon)$ - aprox polinomial

p/ todo $\varepsilon > 0$

Exemplo de Redução: Set Cover $\leq_{strict}$ Hitting Set

$$\begin{array}{l} U = \{1, 2, 3, 4, 5\} \\ \mathcal{S} = \{\{1, 2, 3\}, \\ \quad \{2, 3, 4\}, \\ \quad \{3, 4, 5\}\} \end{array} \xrightarrow{\mathbf{f}} \begin{array}{l} U' = \{a, b, c\} \\ \mathcal{S}' = \{\{a\}, \{a, b\}, \\ \quad \{a, b, c\}, \\ \quad \{b, c\}, \{c\}\} \end{array}$$

$$\begin{array}{l} C = \{\{1, 2, 3\}, \\ \quad \{3, 4, 5\}\} \end{array} \xleftarrow{\mathbf{g}} \mathbf{T} = \{a, c\} \text{ transversal} \xrightarrow{\mathbf{g}^{-1}}$$

$$N = |U|$$

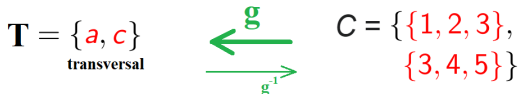
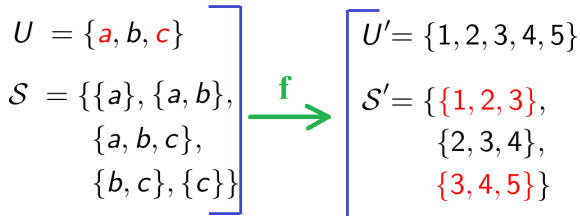
Não existe algoritmo $(\ln N - \varepsilon)$ -aprox
polinomial

p/ todo $\varepsilon > 0$



Não existe algoritmo $(\ln |\mathcal{S}'| - \varepsilon)$ -aprox
polinomial

Exemplo de Redução: Hitting Set $\leq_{strict}$ Set Cover



Tam. s conj máx. = 3



Freq máxima = 3
(do elemento 3)

Algoritmo s-aprox

Algoritmo $(\ln |S| + 1)$ -aprox



Algoritmo freq-aprox

Algoritmo $(\ln |U'| + 1)$ -aprox