Algoritmos Aproximativos - Definição padrão

Quatro ingredientes de um problema de otimização

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

Objetivo: Dada uma instância I, encontrar uma solução ótima $S \in Sol(\mathrm{I})$: 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 $\mathcal{A}(\mathrm{I})$ tq $val(\mathcal{A}(\mathrm{I})) \leq \alpha \cdot opt(\mathrm{I})$ (se **min.** $(\alpha \geq 1)$)
ou $val(\mathcal{A}(\mathrm{I})) \geq \alpha \cdot opt(\mathrm{I})$ (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** $Sol(\mathrm{I})$ para cada instância I
- **Valor** $va\ell(S) > 0$ para cada solução $S \in Sol(\mathrm{I})$
- **Tipo**: minimização ou maximização

$\boxed{\text{DEFINIÇÃO:}}$ Dada uma instancia I e uma solução $S \in Sol(\mathrm{I})$, seja

$$\mathcal{R}(\mathrm{I}, S) = \max\left\{\frac{opt(\mathrm{I})}{va\ell(S)}, \frac{va\ell(S)}{opt(\mathrm{I})}\right\} \geq 1$$

(*performance ratio*)
(*razão de aproximação*)

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

$\boxed{\text{DEFINIÇÃO:}}$ Dada uma instancia I e uma solução $S \in Sol(\mathrm{I})$, seja

$$\mathcal{R}(\mathrm{I}, S) = \max\left\{ \frac{opt(\mathrm{I})}{va\ell(S)}, \frac{va\ell(S)}{opt(\mathrm{I})} \right\} \geq 1$$

(*performance ratio*)

(*razão de aproximação*)

Um algoritmo $\mathcal{A}$ para um problema de otimização é $\boxed{\alpha_{\geq 1}\textbf{-aproximativo}}$ se produz uma solução $\mathcal{A}(\mathrm{I})$ tal que $\mathcal{R}(\mathrm{I}, \mathcal{A}(\mathrm{I})) \leq \alpha$ 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-Metrico:** 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 $k = const$ e $n = \langle \mathrm{I} \rangle$.

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 $\langle \mathrm{I} \rangle$ que retorna solução satisf. $val(\mathcal{A}(\mathrm{I})) = (1 \pm \varepsilon) opt(\mathrm{I})$, 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 $1/\varepsilon$.

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

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 $k = const$ e $n = \langle \mathrm{I} \rangle$.

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 $\langle \mathrm{I} \rangle$ que retorna solução satisf. $\mathcal{R}(\mathrm{I}, \mathcal{A}_\varepsilon(\mathrm{I})) \leq 1 + \varepsilon$, 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 $1/\varepsilon$.

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

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.

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

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-ε) ln n-inaprox. (Moshkovitz'15)

PTAS: Escalonamento Mínimo $\notin$ FPTAS.

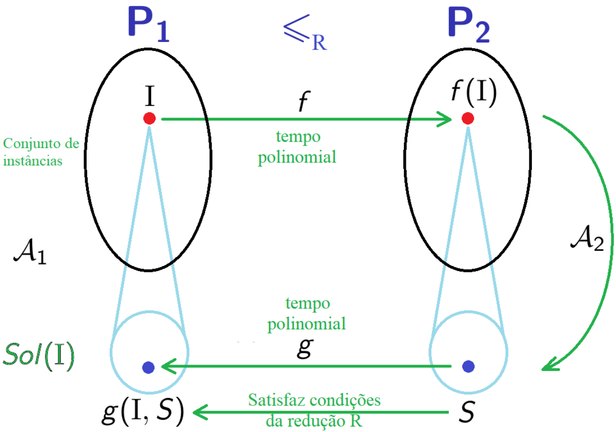
FPTAS: Problema da Mochila $\notin$ PO, pois é NP-Difícil.

$$\boxed{\textbf{PO} \subsetneq \textbf{FPTAS} \subsetneq \textbf{PTAS} \subsetneq \textbf{APX} \subsetneq \textbf{log-APX} \subsetneq \textbf{poly-APX} \subsetneq \textbf{NPO}}$$

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 $\Rightarrow$ $f(\mathrm{I})$ é uma instância de P_2
- $g(\mathrm{I}, S)$ é solução de I em P_1 $\Leftarrow$ Solução S de $f(\mathrm{I})$ 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 $\Rightarrow$ $f(\mathrm{I})$ é uma instância de P_2
- $g(\mathrm{I}, S)$ é solução de I em P_1 $\Leftarrow$ Solução S de $f(\mathrm{I})$ em P_2
- tal que, para toda instância I de P_1 e toda solução S de $f(\mathrm{I})$ em P_2

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

- $\boxed{\mathcal{R}_{P_2}(f(\mathrm{I}), S) \leq r \quad \Rightarrow \quad \mathcal{R}_{P_1}(\mathrm{I}, g(\mathrm{I}, S)) \leq r}$

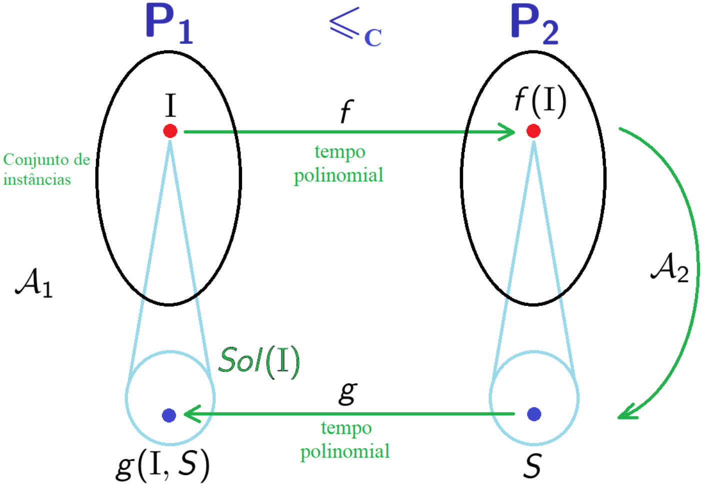
$P_1 \leq_C P_2$: Redução contínua $(f, g, \alpha_{\geq 1})$

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

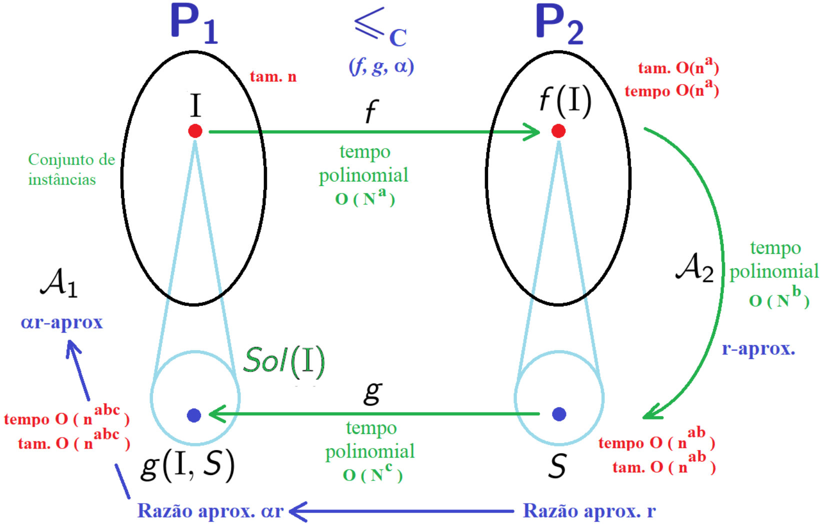
$P_1 \leq_E P_2$: Redução E (relativa ao "*erro*") $(f, g, \alpha_{\geq 1})$

- $\boxed{\mathcal{R}_{P_2}(f(\mathrm{I}), S) \leq r \quad \Rightarrow \quad \mathcal{R}_{P_1}(\mathrm{I}, g(\mathrm{I}, S)) \leq 1 + \alpha \cdot (r - 1)}$
- $\mathbf{P_1 \leq_{strict} P_2 \quad \Rightarrow \quad P_1 \leq_E P_2 \quad \Rightarrow \quad P_1 \leq_C P_2}$

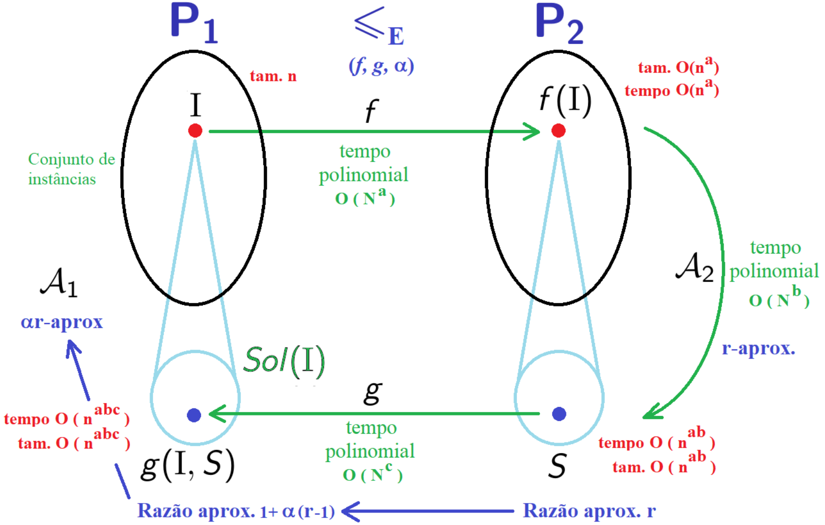
PROVA: $\mathbf{P_1} \leq_{\mathbf{C}} \mathbf{P_2}$ e $\mathbf{P_2} \in \text{APX} \Rightarrow \mathbf{P_1} \in \text{APX}$



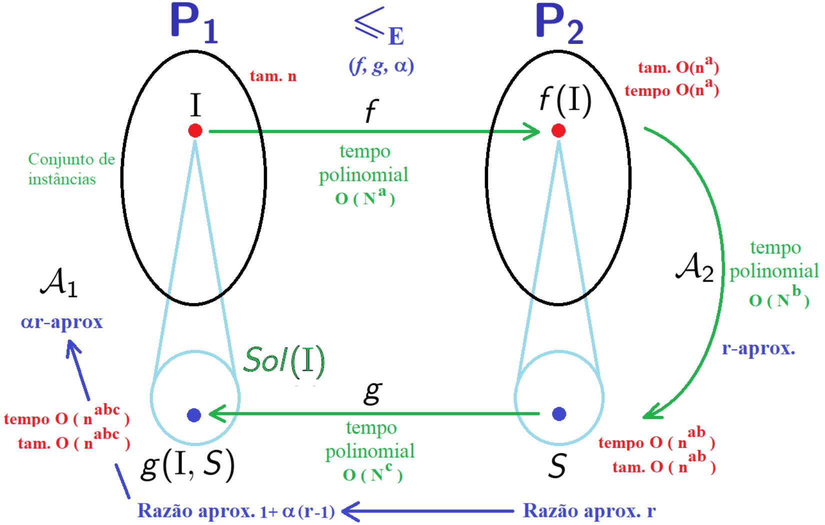
PROVA: $\mathbf{P_1} \leq_C \mathbf{P_2}$ e $\mathbf{P_2} \in$ APX $\Rightarrow$ $\mathbf{P_1} \in$ APX



PROVA: $\mathbf{P_1} \leq_E \mathbf{P_2}$ e $\mathbf{P_2} \in$ PTAS $\Rightarrow$ $\mathbf{P_1} \in$ PTAS



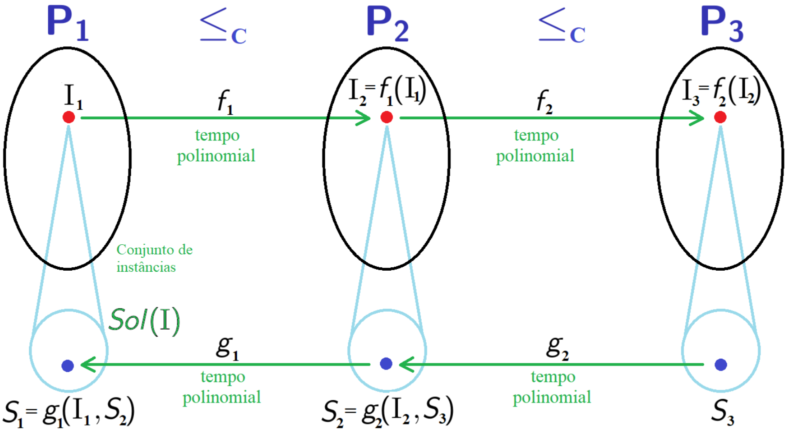
PROVA: $\mathbf{P_1} \leq_E \mathbf{P_2}$ e $\mathbf{P_2} \in$ FPTAS $\Rightarrow$ $\mathbf{P_1} \in$ FPTAS



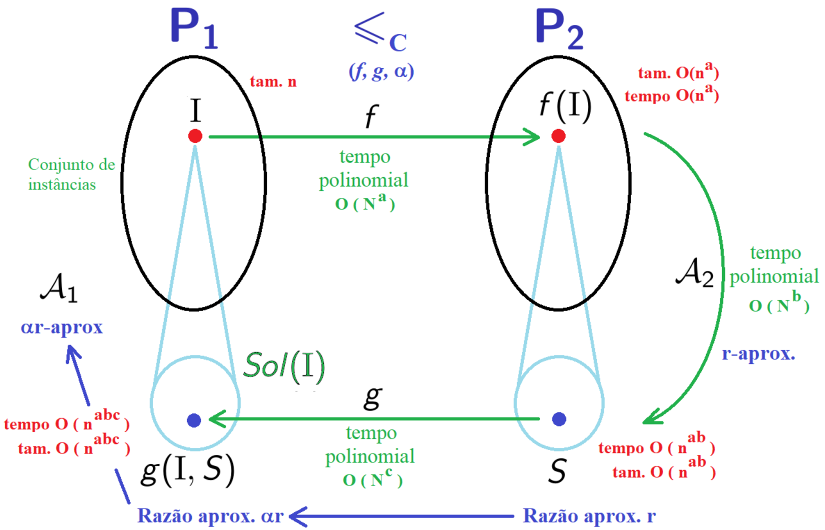
Redução entre Problemas NPO P_1 e P_2

- $\boxed{\mathbf{P_1 \leq_C P_2} \quad \text{e} \quad P_2 \in APX \quad \Rightarrow \quad P_1 \in APX}$
  - $\mathbf{P_1 \leq_C P_2} \quad \text{e} \quad P_1 \notin APX \quad \Rightarrow \quad P_2 \notin APX$
- $\boxed{\mathbf{P_1 \leq_E P_2} \quad \text{e} \quad P_2 \in PTAS \quad \Rightarrow \quad P_1 \in PTAS}$
  - $\mathbf{P_1 \leq_E P_2} \quad \text{e} \quad P_1 \notin PTAS \quad \Rightarrow \quad P_2 \notin PTAS$
- $\boxed{\mathbf{P_1 \leq_E P_2} \quad \text{e} \quad P_2 \in FPTAS \quad \Rightarrow \quad P_1 \in FPTAS}$
  - $\mathbf{P_1 \leq_E P_2} \quad \text{e} \quad P_1 \notin FPTAS \quad \Rightarrow \quad P_2 \notin FPTAS$
- $\mathbf{P_1 \leq_C P_2} \quad \text{e} \quad \mathbf{P_2 \leq_C P_3} \quad \Rightarrow \quad \mathbf{P_1 \leq_C P_3}$
- $\mathbf{P_1 \leq_E P_2} \quad \text{e} \quad \mathbf{P_2 \leq_E P_3} \quad \Rightarrow \quad \mathbf{P_1 \leq_E P_3}$

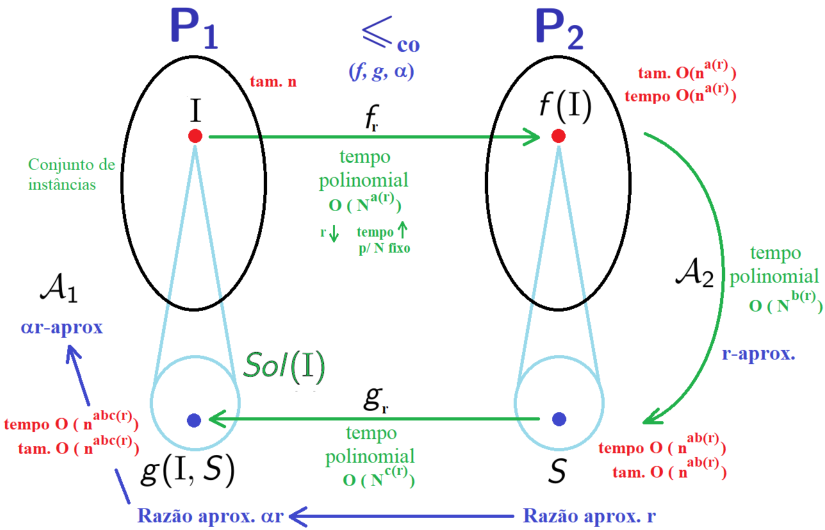
PROVA: $\mathbf{P_1} \leq_C \mathbf{P_2}$ e $\mathbf{P_2} \leq_C \mathbf{P_3} \Rightarrow \mathbf{P_1} \leq_C \mathbf{P_3}$



PROVA: $\mathbf{P_1} \leq_C \mathbf{P_2}$ e $\mathbf{P_2} \in \text{APX} \Rightarrow \mathbf{P_1} \in \text{APX}$



PROVA: $\mathbf{P_1} \leq_{co} \mathbf{P_2}$ e $\mathbf{P_2} \in$ APX $\Rightarrow$ $\mathbf{P_1} \in$ 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 $\Rightarrow$ $f_r(\mathrm{I})$ é uma instância de P_2
- $g_r(\mathrm{I}, S)$ é solução de I em P_1 $\Leftarrow$ Solução S de $f_r(\mathrm{I})$ em P_2
- tal que, para toda instância I de P_1 e toda solução S de $f_r(\mathrm{I})$ em P_2

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

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

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

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

$$
\begin{array}{ccccc}
\mathbf{P_1 \leq_{strict} P_2} & \Rightarrow & \mathbf{P_1 \leq_{E} P_2} & \Rightarrow & \mathbf{P_1 \leq_{C} P_2} \\
 & & \Downarrow & & \Downarrow \\
 & & \mathbf{P_1 \leq_{ap} P_2} & \Rightarrow & \mathbf{P_1 \leq_{co} P_2}
\end{array}
$$

Redução entre Problemas NPO P_1 e P_2

- $\boxed{\mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2} \quad \text{e} \quad P_2 \in \text{APX} \quad \Rightarrow \quad P_1 \in \text{APX}}$
- $\boxed{\mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2} \quad \text{e} \quad P_2 \in \text{poly-APX} \quad \Rightarrow \quad P_1 \in \text{poly-APX}}$
- $\boxed{\mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2} \quad \text{e} \quad P_2 \in \text{log-APX} \quad \Rightarrow \quad P_1 \in \text{log-APX}}$
- $\boxed{\mathbf{P_1} \leq_{\mathbf{ap}} \mathbf{P_2} \quad \text{e} \quad P_2 \in \text{PTAS} \quad \Rightarrow \quad P_1 \in \text{PTAS}}$
- $\mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2} \quad \text{e} \quad \mathbf{P_2} \leq_{\mathbf{co}} \mathbf{P_3} \quad \Rightarrow \quad \mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_3}$
- $\mathbf{P_1} \leq_{\mathbf{ap}} \mathbf{P_2} \quad \text{e} \quad \mathbf{P_2} \leq_{\mathbf{ap}} \mathbf{P_3} \quad \Rightarrow \quad \mathbf{P_1} \leq_{\mathbf{ap}} \mathbf{P_3}$
- $$\begin{array}{ccccc} \mathbf{P_1} \leq_{\mathbf{strict}} \mathbf{P_2} & \Rightarrow & \mathbf{P_1} \leq_{\mathbf{E}} \mathbf{P_2} & \Rightarrow & \mathbf{P_1} \leq_{\mathbf{C}} \mathbf{P_2} \\ & & \Downarrow & & \Downarrow \\ & & \mathbf{P_1} \leq_{\mathbf{ap}} \mathbf{P_2} & \Rightarrow & \mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2} \\ & & \Downarrow & & \\ & & \mathbf{P_1} \leq_{\mathbf{ptas}} \mathbf{P_2} & & \end{array}$$

NPO-completude e APX-completude

NPO-difícil e APX-difícil

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

Conclusões (se P $\neq$ NP)

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

NPO-completude e APX-completude

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

Conclusões

- $\mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2}$ e $\mathbf{P_1} \in$ NPO-difícil $\Rightarrow$ $\mathbf{P_2} \in$ NPO-difícil
- $\mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2}$ e $\mathbf{P_1} \in$ poly-APX-difícil $\Rightarrow$ $\mathbf{P_2} \in$ poly-APX-difícil
- $\mathbf{P_1} \leq_{\mathbf{co}} \mathbf{P_2}$ e $\mathbf{P_1} \in$ log-APX-difícil $\Rightarrow$ $\mathbf{P_2} \in$ log-APX-difícil
- $\mathbf{P_1} \leq_{\mathbf{ap}} \mathbf{P_2}$ e $\mathbf{P_1} \in$ APX-difícil $\Rightarrow$ $\mathbf{P_2} \in$ 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-ε) ln n-inaprox. (Moshkovitz'15)

PTAS: Escalonamento Mínimo $\notin$ FPTAS.

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

$\boxed{\textbf{PO} \subsetneq \textbf{FPTAS} \subsetneq \textbf{PTAS} \subsetneq \textbf{APX} \subsetneq \textbf{log-APX} \subsetneq \textbf{poly-APX} \subsetneq \textbf{NPO}}$

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

$U = \{\, 2, 5, 5, 8, 11 \,\}$

$\mathrm{T} = 17{,}9$

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

$\xrightarrow{\mathbf{f}}$

$$
\begin{array}{ll}
 & \quad\ \ 1 \quad 2 \quad 3 \quad 4 \quad 5 \\
\text{pesos} & = [\, 2, 5, 5, 8, 11 \,] \\
\text{valores} & = [\, 2, 5, 5, 8, 11 \,] \\
\text{capacidade} & = 17{,}9
\end{array}
$$

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

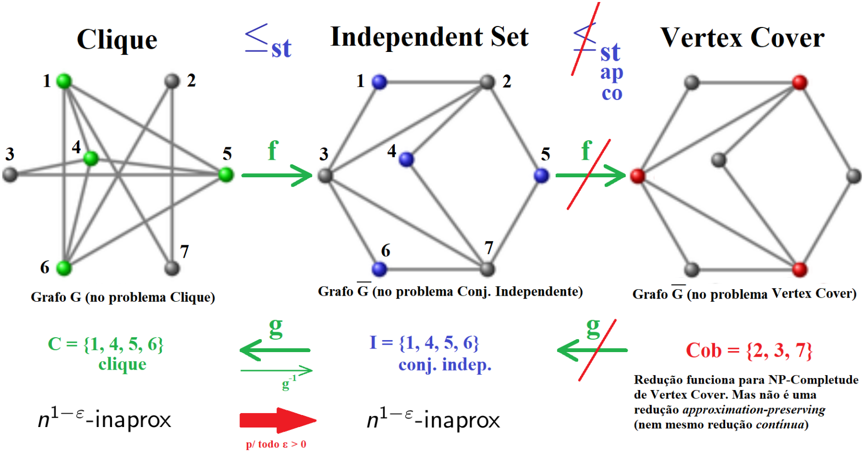
$U' = \{\, 5, 11 \,\}$ $\xleftarrow{\mathbf{g}}$ Itens $= \{2, 5\}$

$\xrightarrow[g^{-1}]{}$

FPTAS $\Longleftarrow$ **FPTAS**

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

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



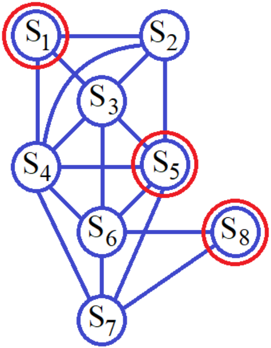
Exemplo de Redução: Set Packing $\leq_{strict}$ Independent Set

Set Packing $\leq_{st}$ **Independent Set**

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

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

$\xrightarrow{f}$

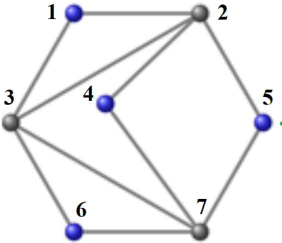


$\text{Pack} = \{S_1, S_5, S_8\} \xleftarrow{g} I = \{S_1, S_5, S_8\}$

$\xrightarrow{g^{-1}}$

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

Independent Set $\leq_{st}$ **Set Packing**



Grafo G (no problema Conj. Independente)

$\xrightarrow{f}$

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}

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

$\xleftarrow{g}$ $\xrightarrow[g^{-1}]{}$

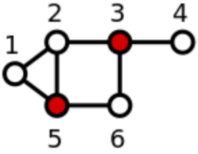
Pack = {S_1, S_4, S_5, S_6}

$n^{1-\varepsilon}$-inaprox
n = |V(G)|

p/ todo ε > 0

$n^{1-\varepsilon}$-inaprox
mesmo se interseção de 2 conjuntos tenha tamanho no máximo 1
n = num. conjuntos

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



$\xrightarrow{\mathbf{f}}$

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

$S_1 = \{1, 2, 5\}$
$S_2 = \{1, 2, 3, 5\}$
$S_3 = \{2, 3, 4, 6\}$
$S_4 = \{3, 4\}$
$S_5 = \{1, 2, 5, 6\}$
$S_6 = \{3, 5, 6\}$

$D = \{3, 5\} \xleftarrow{\mathbf{g}} C = \{S_3, S_5\}$

$\xrightarrow[g^{-1}]{}$

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

Frequência máxima = Δ+1 = 4
(no elemento 2, por exemplo)

Algoritmo (Δ+1) - aprox
Algoritmo (ln n + 1)-aprox

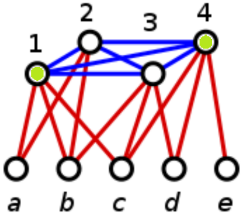
Algoritmo freq-aprox
Algoritmo (ln n + 1)-aprox

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

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

$S_1 = \{a, b, c\}$
$S_2 = \{a, b\}$
$S_3 = \{b, c, d\}$
$S_4 = \{c, d, e\}$

$\xrightarrow{\mathbf{f}}$



$C = \{S_1, S_4\}$ $\xleftarrow{\mathbf{g}}$ $D = \{a, b, 4\}$

$\xrightarrow[g^{-1}]{}$ $D = \{1, 4\}$

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

Grafo split (clique + indep)
N =α(G) = tam. maior conjunto independente

p/ todo ε > 0

Não existe algoritmo (ln N - ε) - aprox polinomial ⟹ **Não existe algoritmo (ln α(G) - ε) - aprox polinomial**

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

$$U = \{1,2,3,4,5\} \qquad \xrightarrow{\ \mathbf{f}\ } \qquad U' = \{a, b, c\}$$

$$\mathcal{S} = \{\{1,2,3\}, \{2,3,4\}, \{3,4,5\}\} \qquad \mathcal{S}' = \{\{a\}, \{a,b\}, \{a,b,c\}, \{b,c\}, \{c\}\}$$

$$C = \{\{1,2,3\}, \{3,4,5\}\} \quad \underset{g^{-1}}{\overset{\mathbf{g}}{\leftrightarrows}} \quad \mathbf{T} = \{a, c\}$$

transversal

$N = |U|$

Não existe algoritmo $(\ln N - \varepsilon)$ - aprox polinomial $\xrightarrow{\text{p/ todo } \varepsilon > 0}$ Não existe algoritmo $\left(\ln \left|\mathcal{S}'\right| - \varepsilon\right)$ - aprox polinomial

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

$$U = \{a, b, c\} \qquad U' = \{1, 2, 3, 4, 5\}$$

$$\mathcal{S} = \{\{a\}, \{a, b\}, \{a, b, c\}, \{b, c\}, \{c\}\} \xrightarrow{\mathbf{f}} \mathcal{S}' = \{\{1, 2, 3\}, \{2, 3, 4\}, \{3, 4, 5\}\}$$

$$\mathbf{T} = \{a, c\} \xleftarrow{\mathbf{g}} C = \{\{1, 2, 3\}, \{3, 4, 5\}\}$$

transversal

$$\xrightarrow[g^{-1}]{}$$

Tam. s conj máx. = 3 $\longrightarrow$ **Freq máxima = 3 (do elemento 3)**

Algoritmo s-aprox $\Longleftarrow$ **Algoritmo freq-aprox**

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