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 consideramos que $(\alpha{>}1)$-aprox quando maximização ou $(\alpha{<}1)$-aprox quando minimização, se refere na verdade a $(1/\alpha)$-aprox., para simplificar a escrita. Ou seja, MaxClique é $(1/n)$-aproximável, mas $n^{\varepsilon-1}$-inaproximável em tempo polinomial para qualquer $\varepsilon > 0$ (Zuckerman'06)

Problemas de Otimização

Quatro ingredientes de um problema de otimização

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

$\boxed{\text{Observações:}}$ Se $\mathit{Sol}(\mathrm{I}) = \emptyset$, dizemos que a instância I é inviável.
Objetivo: dada uma instância I, encontrar uma solução ótima $S \in \mathit{Sol}(\mathrm{I})$: solução com valor mínimo ou máximo, dependendo do tipo do problema. Valor ótimo $\mathbf{opt}(\mathrm{I})$ é o valor de uma solução ótima de uma instância I.

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

Aqui consideramos que $(\alpha{>}1)$-aprox quando maximização ou $(\alpha{<}1)$-aprox quando minimização, se refere na verdade a $(1/\alpha)$-aprox., para simplificar a escrita. Ou seja, $\mathit{va\ell}(\mathcal{A}(\mathrm{I})) \geq [1/\alpha,\, \alpha] \cdot \mathit{opt}(\mathrm{I})$, para $\alpha \geq 1$

Problemas de Otimização - Definições

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

Seja $\boxed{\langle \mathrm{I} \rangle}$ o tamanho da instância I (muitas vezes, usa-se n: $\langle \mathrm{I} \rangle = O(n)$).

Um algoritmo é $\boxed{\text{polinomial}}$ se existe um polinômio p tal que o consumo de tempo do algoritmo seja $O(p(\langle \mathrm{I} \rangle))$ para toda instância I.

Seja $\boxed{\mathit{Max}(\mathrm{I})}$ o maior inteiro (em valor absoluto) que ocorre em I. Se I não tem valores numéricos (como Vertex-Cover), $\mathit{Max}(\mathrm{I}) = 0$. Racionais são frações de 2 inteiros. Por exemplo, $\mathit{Max}(\mathrm{I}) \geq 17$ se $3/17$ ocorre em I. Os bits para representar valores númericos estão sendo contados em $\langle \mathrm{I} \rangle$.

Um algoritmo é $\boxed{\text{pseudo-polinomial}}$ se é poli quando $\mathit{Max}(\mathrm{I}) \leq p(\langle \mathrm{I} \rangle)$ para um polinômio p. **Exemplo:** o algoritmo de tempo $O(n \cdot C)$ do Problema da Mochila é pseudo-polinomial.

$\boxed{\text{NP-Difícil Forte}}$: se NP-Difícil quando $\mathit{Max}(\mathrm{I}) \leq p(\langle \mathrm{I} \rangle)$ para polin. p.
Portanto: NP-Difícil Fraco $\Leftrightarrow$ NPD e tem algoritmo pseudo-polinomial.
Exemplo: NP-Difícil Fraco (Mochila) e NP-Difícil Forte (MaxSAT).

Problemas de Otimização - Exemplos

Suponha que I seja um vetor com n inteiros representados por n bits cada um. Então $\langle \mathrm{I} \rangle = n^2$, mas $Max(\mathrm{I}) \leq 2^n$ (número binário).

Se os valores numéricos fossem representados em unário$^+$ (basicamente o valor é o número de 1's), então $Max(\mathrm{I}) \leq n$.

Definições alternativas para pseudo-polinomial e NP-Difícil Forte:
polinomial ou NP-Difícil quando os valores numéricos são representados em unário$^+$.

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 $\mathcal{A}$ polinomial em $\langle \mathrm{I} \rangle$ que retorna solução satisf. $val(\mathcal{A}(\mathrm{I}, \varepsilon)) = (1 \pm \varepsilon) opt(\mathrm{I})$, para cada racional ε 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 e Min Color, 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 e MinColor $\notin$ log-APX, pois são $n^{1-\varepsilon}$-inaprox. (Zuckerman'06)

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

PTAS: Escalonamento Mínimo $\notin$ FPTAS.

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

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

FPTAS $\subsetneq$ PTAS, se P $\neq$ NP

Teorema [Garey, Johnson'78]: Seja Π um problema NP-Difícil Forte tal que, para toda instância I, o valor de qualquer solução é um **inteiro** não-negativo e $opt(\mathrm{I}) \leq p(\langle \mathrm{I} \rangle, Max(\mathrm{I}))$ para algum polinômio p.
Se P $\neq$ NP, então $\Pi \notin$ FPTAS.

PROVA: Suponha que Π é de maximização. Por contradição, seja $\mathcal{A}$ um FPTAS para Π. Seja $\varepsilon = (p(\langle \mathrm{I} \rangle, Max(\mathrm{I})) + 1)^{-1}$. Logo:
$val(\mathcal{A}(\mathrm{I}, \varepsilon)) \;\geq\; (1 - \varepsilon) opt(\mathrm{I})$.

$$opt(\mathrm{I}) - val(\mathcal{A}(\mathrm{I}, \varepsilon)) \;\leq\; \varepsilon \cdot opt(\mathrm{I}) \;=\; \frac{opt(\mathrm{I})}{p(\langle \mathrm{I} \rangle, Max(\mathrm{I})) + 1} \;<\; 1$$

Como os valores são números inteiros, $val(\mathcal{A}(\mathrm{I}, \varepsilon)) = opt(\mathrm{I})$ e então $\mathcal{A}$ resolve Π exatamente em tempo polinomial em $\langle \mathrm{I} \rangle$ e em $1/\varepsilon$ (ou seja, polinomial em $Max(\mathrm{I})$). Ou seja, $\mathcal{A}$ é pseudo-polinomial. Se P $\neq$ NP, temos uma contradição, pois Π é NP-Difícil Forte.

FPTAS $\subsetneq$ PTAS, se P $\neq$ NP

Problema Escalonamento-Int: Idêntico a Escalonamento (n tarefas, m máquinas), mas todos os valores numéricos (tempos das tarefas) são **inteiros positivos**.

Escalonamento e Escalonamento-Int são polinomialmente equivalentes, pois os valores númericos de Escalonamento são racionais e podemos traduzir um problema para o outro. Além disso, um algoritmo α-aprox. de um pode ser transformado em um α-aprox. do outro. Portanto, **Escalonamento-Int $\in$ PTAS**.

Escalonamento-Int é NP-Difícil Forte: O clássico problema **3-Partition** é um caso especial (dados $n = 3m$ inteiros positivos, decidir se é possível particioná-los em m triplas com a mesma soma). **3-Partition** é NP-Difícil Forte [Livro de Garey e Johnson]: mesmo se os inteiros são limitados por um polinômio em n.

Escalonamento-Int satisfaz condições do Teo [Garey,Johnson'78], então:

Escalonamento-Int e **Escalonamento** não pertencem a FPTAS.

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

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

PTAS: Escalonamento Mínimo $\notin$ FPTAS.

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

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