Una tercera aplicación de la técnica

Sea $G = (N, A)$ un grafo (dirigido) sin loops.

Decimos que $(v_1, \ldots, v_n)$ es un ciclo simple en G si:

1. $n \geq 2$
2. $(v_i, v_{i+1}) \in A$ para cada $1 \leq i \leq n-1$, y $(v_n, v_1) \in A$
3. $v_i \neq v_j$ para cada $1 \leq i < j \leq n$

Una tercera aplicación de la técnica

Un ciclo simple $(v_1, \ldots, v_n)$ con n vértices puede ser visto como un conjunto con n arcos:

$$\{(v_1, v_2), \ldots (v_{n-1}, v_n), (v_n, v_1)\}$$

En este capítulo utilizamos esta representación de los ciclos simples, y decimos que el largo de este ciclo es n

Una tercera aplicación de la técnica

Un ciclo simple $(v_1, \ldots, v_n)$ con n vértices puede ser visto como un conjunto con n arcos:

$$\{(v_1, v_2), \ldots (v_{n-1}, v_n), (v_n, v_1)\}$$

En este capítulo utilizamos esta representación de los ciclos simples, y decimos que el largo de este ciclo es n

Nótese que dada la representación de ciclos simples tenemos que $(a, b, c) = (b, c, a) = (c, a, b)$, puesto que:

$$\{(a, b), (b, c), (c, a)\} = \{(b, c), (c, a), (a, b)\} = \{(c, a), (a, b), (b, c)\}$$

Una tercera aplicación de la técnica

Sea #CicloSimple una función que, dado un grafo G, retorna el número de ciclos simples en G

Una tercera aplicación de la técnica

Sea #CicloSimple una función que, dado un grafo G, retorna el número de ciclos simples en G

Ejercicio

Muestre que $L_{\#\text{CicloSimple}} \in \text{PTIME}$

Una tercera aplicación de la técnica

Sea #CicloSimple una función que, dado un grafo G, retorna el número de ciclos simples en G

Ejercicio

Muestre que $L_{\#\text{CicloSimple}} \in \text{PTIME}$

Teorema

Si existe un FPRAS para #CicloSimple, entonces $NP = RP$

#CicloSimple no admite un FPRAS: demostración

Sea HAM $= \{G \mid G$ tiene un ciclo hamiltoniano$\}$

Vamos a demostrar que si existe un FPRAS para #CicloSimple, entonces HAM $\in$ BPP

- Dado que HAM es NP-completo, se concluye que NP $\subseteq$ BPP
- Al igual que en los casos anteriores, concluimos que NP $=$ RP a partir de que NP $\subseteq$ BPP

#CicloSimple no admite un FPRAS: demostración

Sea HAM $= \{G \mid G$ tiene un ciclo hamiltoniano$\}$

Vamos a demostrar que si existe un FPRAS para #CicloSimple, entonces HAM $\in$ BPP

- Dado que HAM es NP-completo, se concluye que NP $\subseteq$ BPP
- Al igual que en los casos anteriores, concluimos que NP $=$ RP a partir de que NP $\subseteq$ BPP

Sea $G = (N, A)$ un grafo con $|N| = n$ y $n \geq 2$

- G es una entrada de HAM

¿Cómo construimos el grafo aumentado?

A partir de un ciclo simple con ℓ nodos esperamos generar $2^{\ell \cdot r}$ ciclos simples en el grafo aumentado

¿Cómo construimos el grafo aumentado?

A partir de un ciclo simple con ℓ nodos esperamos generar $2^{\ell \cdot r}$ ciclos simples en el grafo aumentado

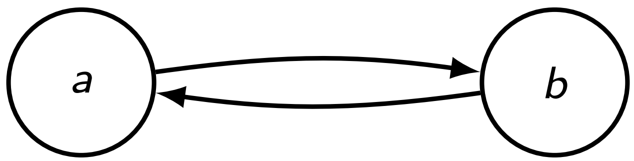
- Primer intento: construir G^r como para el caso de #LIS

¿Cómo construimos el grafo aumentado?

A partir de un ciclo simple con ℓ nodos esperamos generar $2^{\ell \cdot r}$ ciclos simples en el grafo aumentado

- Primer intento: construir G^r como para el caso de #LIS

Considere el siguiente grafo H:

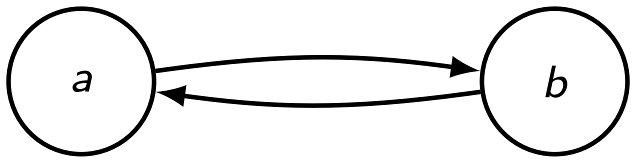


¿Cómo construimos el grafo aumentado?

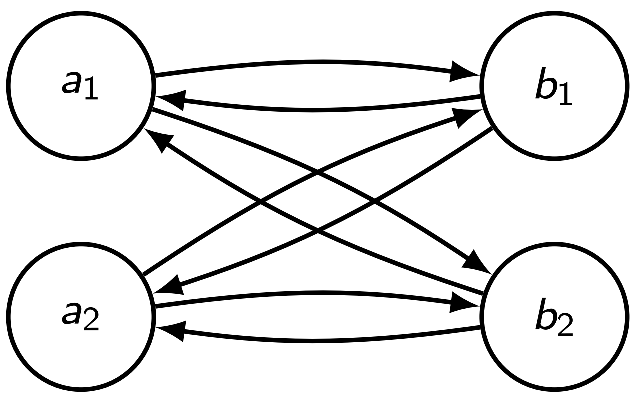
A partir de un ciclo simple con ℓ nodos esperamos generar $2^{\ell \cdot r}$ ciclos simples en el grafo aumentado

- Primer intento: construir G^r como para el caso de #LIS

Considere el siguiente grafo H:



En este caso H^2 es el siguiente grafo:



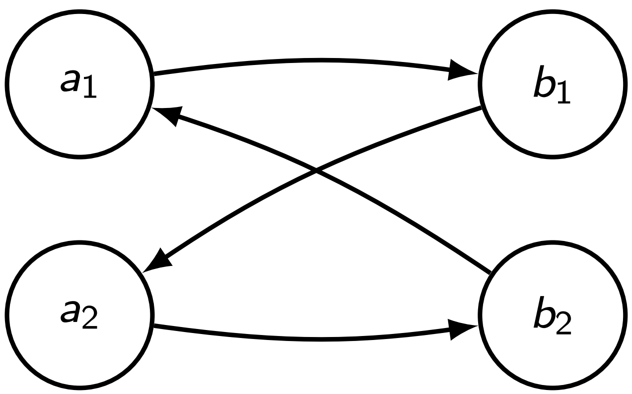
¿Cómo construimos el grafo aumentado?

En H tenemos un ciclo simple y esperamos tener 2^4 ciclos simples en H^2

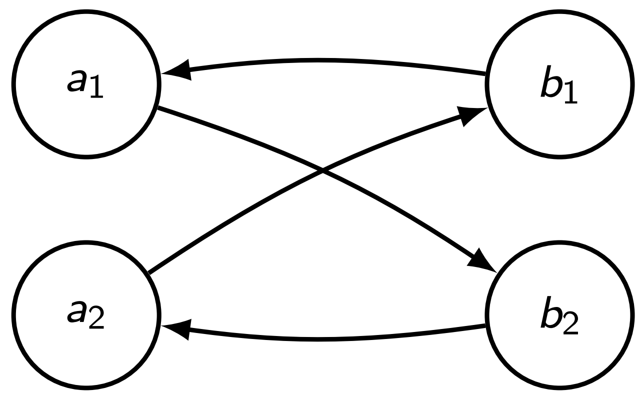
¿Cómo construimos el grafo aumentado?

En H tenemos un ciclo simple y esperamos tener 2^4 ciclos simples en H^2

El problema es que H^2 sólo tiene 6 ciclos simples: (a_1, b_1), (a_1, b_2), (a_2, b_1), (a_2, b_2) y:



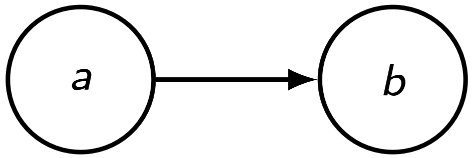
(a_1, b_1, a_2, b_2)



(a_1, b_2, a_2, b_1)

¿Cómo construimos el grafo aumentado?

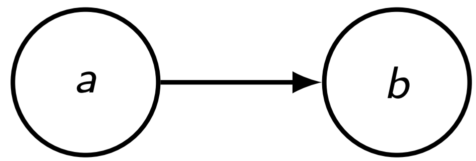
Segundo intento: reemplazamos cada arco



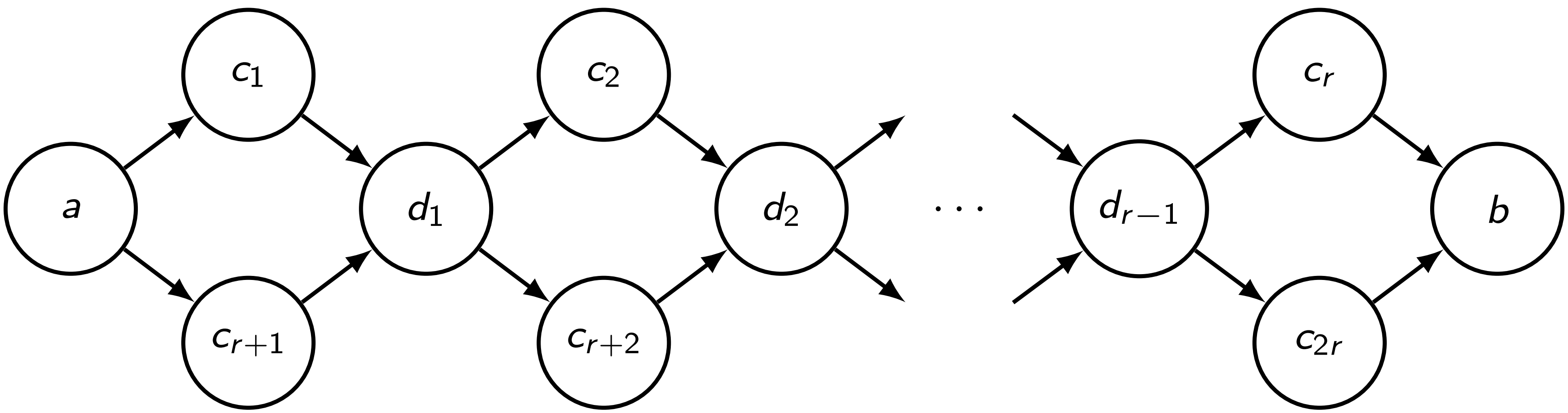
por

¿Cómo construimos el grafo aumentado?

Segundo intento: reemplazamos cada arco



por



donde $c_1, \ldots, c_r, c_{r+1}, \ldots, c_{2r}, d_1, \ldots, d_{r-1}$ son nodos nuevos creados para reemplazar un arco.

El grafo aumentado

Sea G^r el grafo definido reemplazando cada arco como fue mostrado en la transparencia anterior.

Ejercicio

Defina formalmente el grafo G^r

El grafo aumentado

Sea G^r el grafo definido reemplazando cada arco como fue mostrado en la transparencia anterior.

Ejercicio

Defina formalmente el grafo G^r

Dado que G no tiene loops puede tener a lo más $n \cdot (n-1)$ arcos

- Por lo tanto, G^r tiene a lo más $n + n \cdot (n-1) \cdot (3 \cdot r - 1)$ nodos, y es de tamaño polinomial en el tamaño de G

La noción de testigo

Cada ciclo simple de G de largo $\ell \geq 2$ corresponde con $2^{\ell \cdot r}$ ciclos de G^r de largo $2 \cdot \ell \cdot r$

- Cada uno de estos ciclos de largo $2 \cdot \ell \cdot r$ en G^r es considerado como un testigo del ciclo de largo ℓ en G

Cada ciclo simple de G^r es testigo de un único ciclo simple de G

La noción de testigo

Cada ciclo simple de G de largo $\ell \geq 2$ corresponde con $2^{\ell \cdot r}$ ciclos de G^r de largo $2 \cdot \ell \cdot r$

- Cada uno de estos ciclos de largo $2 \cdot \ell \cdot r$ en G^r es considerado como un testigo del ciclo de largo ℓ en G

Cada ciclo simple de G^r es testigo de un único ciclo simple de G

Ejercicio

Formalice la noción de testigo.

La noción de testigo

La conexión fundamental entre los grafos:

$G \in \text{HAM}$ si y sólo si G^r tiene un ciclo simple de largo $2 \cdot r \cdot n$

La noción de testigo

La conexión fundamental entre los grafos:

$$G \in \text{HAM} \;\text{ si y sólo si }\; G^r \text{ tiene un ciclo simple de largo } 2 \cdot r \cdot n$$

Utilizando esta conexión y eligiendo un valor adecuado para r vamos a construir un algoritmo aleatorizado de tiempo polinomial para HAM

- Teniendo como hipótesis la existencia de un FPRAS para #CicloSimple

El número de testigos

Definimos los conjuntos:

$$
\begin{aligned}
I^r(G) &= \{C \mid C \text{ es un ciclo simple en } G^r \text{ de largo } 2 \cdot r \cdot n\} \\
M^r(G) &= \{C \mid C \text{ es un ciclo simple en } G^r \text{ de largo menor que } 2 \cdot r \cdot n\}
\end{aligned}
$$

Tenemos que $G \in$ HAM si y sólo si $I^r(G) \neq \emptyset$

- Suponiendo la existencia de un FPRAS para #CicloSimple, vamos a desarrollar un algoritmo aleatorizado de tiempo polinomial para verificar si $I^r(G) \neq \emptyset$

El número de testigos

Para $\ell \in \{2, 3, \ldots, n\}$ definimos t_ℓ como la cantidad de testigos en G^r para un ciclo simple en G de largo ℓ

El número de testigos

Para $\ell \in \{2, 3, \ldots, n\}$ definimos t_ℓ como la cantidad de testigos en G^r para un ciclo simple en G de largo ℓ

Por la definición de G^r tenemos que $t_\ell = 2^{\ell \cdot r}$

El número de testigos

Para $\ell \in \{2, 3, \ldots, n\}$ definimos t_ℓ como la cantidad de testigos en G^r para un ciclo simple en G de largo ℓ

Por la definición de G^r tenemos que $t_\ell = 2^{\ell \cdot r}$

Además, el número de ciclos simples en G de largo ℓ está acotado por:

$$\binom{n \cdot (n-1)}{\ell}$$

El número de testigos

A partir de lo anterior concluimos que:

$$
\begin{aligned}
|M^r(G)| &\leq \sum_{\ell=2}^{n-1} \binom{n \cdot (n-1)}{\ell} t_\ell \\
&\leq \sum_{\ell=2}^{n-1} \binom{n \cdot (n-1)}{\ell} t_{n-1} \\
&\leq t_{n-1} \sum_{\ell=0}^{n \cdot (n-1)} \binom{n \cdot (n-1)}{\ell} \\
&= 2^{n \cdot (n-1)} \cdot t_{n-1} \\
&= 2^{n \cdot (n-1)} \cdot 2^{(n-1) \cdot r} \\
&= 2^{(n-1) \cdot (n+r)}
\end{aligned}
$$

Idea fundamental: aumentando las diferencias

Por otra parte, si $G \in$ HAM tenemos que:

$$|I^r(G)| \;\geq\; t_n \;=\; 2^{n \cdot r}$$

Idea fundamental: aumentando las diferencias

Por otra parte, si $G \in$ HAM tenemos que:

$$|I^r(G)| \ \geq \ t_n \ = \ 2^{n \cdot r}$$

Concluimos que:

- Si $G \notin$ HAM: $\#\text{CicloSimple}(G^r) \ = \ |M^r(G)| \ \leq \ 2^{(n-1)\cdot(n+r)}$
- Si $G \in$ HAM: $\#\text{CicloSimple}(G^r) \ \geq \ |I^r(G)| \ \geq \ 2^{n \cdot r}$

Idea fundamental: aumentando las diferencias

Por otra parte, si $G \in$ HAM tenemos que:

$$|I^r(G)| \ \geq \ t_n \ = \ 2^{n \cdot r}$$

Concluimos que:

- Si $G \notin$ HAM: $\#\text{CicloSimple}(G^r) \ = \ |M^r(G)| \ \leq \ 2^{(n-1)\cdot(n+r)}$
- Si $G \in$ HAM: $\#\text{CicloSimple}(G^r) \ \geq \ |I^r(G)| \ \geq \ 2^{n \cdot r}$

En las siguientes transparencias vamos a obtener un valor de r que nos permita tener un algoritmo aleatorizado de tiempo polinomial para HAM

Un algoritmo aleatorizado para HAM

Suponemos que existe un FPRAS $\mathcal{A}$ para #CicloSimple, y mostramos como esto puede ser utilizado para obtener que HAM $\in$ BPP

Un algoritmo aleatorizado para HAM

Suponemos que existe un FPRAS $\mathcal{A}$ para #CicloSimple, y mostramos como esto puede ser utilizado para obtener que HAM $\in$ BPP

Definimos el siguiente algoritmo aleatorizado $\mathcal{B}$ para HAM:

1. Dada la entrada G, donde G es un grafo con n nodos, si $n \leq 1$ retorne **no**, en otro caso vaya al paso 2
2. Genere el grafo G^r
3. Sea s el resultado de ejecutar $\mathcal{A}(G^r, \frac{1}{2})$
4. Si $s > (1 + \frac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)}$, entonces retorne **sí**, en caso contrario retorne **no**

Un algoritmo aleatorizado para HAM

Suponemos que existe un FPRAS $\mathcal{A}$ para #CicloSimple, y mostramos como esto puede ser utilizado para obtener que HAM $\in$ BPP

Definimos el siguiente algoritmo aleatorizado $\mathcal{B}$ para HAM:

1. Dada la entrada G, donde G es un grafo con n nodos, si $n \leq 1$ retorne **no**, en otro caso vaya al paso 2
2. Genere el grafo G^r
3. Sea s el resultado de ejecutar $\mathcal{A}(G^r, \frac{1}{2})$
4. Si $s > (1 + \frac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)}$, entonces retorne **sí**, en caso contrario retorne **no**

Nótese que $\mathcal{B}$ funciona en tiempo polinomial en el tamaño de la entrada G si el valor de r es polinomial en n

- Recuerde que tenemos que determinar el valor de r

La probabilidad de error del algoritmo

Si $G \not\in$ HAM obtenemos que $\mathbf{Pr}(\mathcal{B}(G)$ retorne **sí**) es igual a:

La probabilidad de error del algoritmo

Si $G \not\in$ HAM obtenemos que $\mathbf{Pr}(\mathcal{B}(G)$ retorne **sí**) es igual a:

$$
\begin{aligned}
&\mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) > (1 + \tfrac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)}) \\
&\quad \leq \mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) > (1 + \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r)) \\
&\quad \leq \mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) > (1 + \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r)) \vee \\
&\qquad\qquad\qquad \mathcal{A}(G^r, \tfrac{1}{2}) < (1 - \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r)) \\
&\quad = 1 - \mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) \leq (1 + \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r) \wedge \\
&\qquad\qquad\qquad \mathcal{A}(G^r, \tfrac{1}{2}) \geq (1 - \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r)) \\
&\quad = 1 - \mathbf{Pr}(|\mathcal{A}(G^r, \tfrac{1}{2}) - \#\text{CicloSimple}(G^r)| \leq \tfrac{1}{2} \cdot \#\text{CicloSimple}(G^r)) \\
&\quad \leq 1 - \frac{3}{4} = \frac{1}{4}
\end{aligned}
$$

La probabilidad de error del algoritmo

Consideramos ahora el caso en que $G \in \text{HAM}$

La probabilidad de error del algoritmo

Consideramos ahora el caso en que $G \in \text{HAM}$

Primero debemos encontrar un valor de r tal que:

$$(1 - \frac{1}{2}) \cdot \#\text{CicloSimple}(G^r) \quad > \quad (1 + \frac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)}$$

Dado que $\#\text{CicloSimple}(G^r) \geq 2^{n \cdot r}$, basta entonces encontrar un valor de r tal que:

$$(1 - \frac{1}{2}) \cdot 2^{n \cdot r} \quad > \quad (1 + \frac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)}$$

La probabilidad de error del algoritmo

Tenemos que:

$$
\begin{aligned}
& (1 - \frac{1}{2}) \cdot 2^{n \cdot r} \ > \ (1 + \frac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)} \\
\Leftrightarrow \quad & \frac{1}{2} \cdot 2^{n \cdot r} \ > \ \frac{3}{2} \cdot 2^{(n-1)\cdot(n+r)} \\
\Leftrightarrow \quad & 2^{n \cdot r} \ > \ 3 \cdot 2^{(n-1)\cdot(n+r)} \\
\Leftrightarrow \quad & 2^{n \cdot r} \ > \ 2^{(n-1)\cdot(n+r)+\log_2(3)} \\
\Leftrightarrow \quad & n \cdot r \ > \ n \cdot (n-1) + (n-1) \cdot r + \log_2(3) \\
\Leftrightarrow \quad & r \ > \ n \cdot (n-1) + \log_2(3)
\end{aligned}
$$

La probabilidad de error del algoritmo

Tenemos que:

$$
\begin{aligned}
& (1-\frac{1}{2}) \cdot 2^{n \cdot r} \;>\; (1+\frac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)} \\
\Leftrightarrow \quad & \frac{1}{2} \cdot 2^{n \cdot r} \;>\; \frac{3}{2} \cdot 2^{(n-1)\cdot(n+r)} \\
\Leftrightarrow \quad & 2^{n \cdot r} \;>\; 3 \cdot 2^{(n-1)\cdot(n+r)} \\
\Leftrightarrow \quad & 2^{n \cdot r} \;>\; 2^{(n-1)\cdot(n+r)+\log_2(3)} \\
\Leftrightarrow \quad & n \cdot r \;>\; n \cdot (n-1) + (n-1) \cdot r + \log_2(3) \\
\Leftrightarrow \quad & r \;>\; n \cdot (n-1) + \log_2(3)
\end{aligned}
$$

Considerando $r = n \cdot (n-1) + 2$ se cumple la condición.

- Concluimos que $(1 - \frac{1}{2}) \cdot \#\text{CicloSimple}(G^r) > (1 + \frac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)}$

La probabilidad de error del algoritmo

Si $G \in$ HAM y $r = n \cdot (n-1) + 2$ tenemos que $\mathbf{Pr}(\mathcal{B}(G)$ retorne **no**) es igual a:

La probabilidad de error del algoritmo

Si $G \in \text{HAM}$ y $r = n \cdot (n-1) + 2$ tenemos que $\mathbf{Pr}(\mathcal{B}(G)$ retorne **no**$)$ es igual a:

$$
\begin{aligned}
\mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) &\leq (1 + \tfrac{1}{2}) \cdot 2^{(n-1)\cdot(n+r)}) \\
&\leq \mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) < (1 - \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r)) \\
&\leq \mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) < (1 - \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r) \ \vee \\
&\qquad\qquad\qquad \mathcal{A}(G^r, \tfrac{1}{2}) > (1 + \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r)) \\
&= 1 - \mathbf{Pr}(\mathcal{A}(G^r, \tfrac{1}{2}) \geq (1 - \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r) \ \wedge \\
&\qquad\qquad\qquad \mathcal{A}(G^r, \tfrac{1}{2}) \leq (1 + \tfrac{1}{2}) \cdot \#\text{CicloSimple}(G^r)) \\
&= 1 - \mathbf{Pr}(|\mathcal{A}(G^r, \tfrac{1}{2}) - \#\text{CicloSimple}(G^r)| \leq \tfrac{1}{2} \cdot \#\text{CicloSimple}(G^r)) \\
&\leq 1 - \frac{3}{4} \ = \ \frac{1}{4}
\end{aligned}
$$

#CicloSimple no admite un FPRAS: conclusión

Tenemos que en ambos casos el error del algoritmo $\mathcal{B}$ está acotado superiormente por $\frac{1}{4}$

- Considerando $r = n \cdot (n-1) + 2$ en ambos casos

Concluimos entonces que HAM $\in$ BPP

- Por lo tanto NP $\subseteq$ BPP $\square$

La existencia de FPRAS y las reducciones parsimoniosas

Sean $f, g : \Sigma^* \to \mathbb{N}$ dos funciones en #P

Teorema

Si $f \leq^p_{par} g$ y existe un FPRAS para g, entonces existe un FPRAS para f

La existencia de FPRAS y las reducciones parsimoniosas

Sean $f, g : \Sigma^* \to \mathbb{N}$ dos funciones en $\#$P

Teorema

Si $f \leq^p_{par} g$ y existe un FPRAS para g, entonces existe un FPRAS para f

Demostración: Suponga que $h : \Sigma^* \to \Sigma^*$ es una función computable en tiempo polinomial tal que $f(w) = g(h(w))$ para todo $w \in \Sigma^*$

La existencia de FPRAS y las reducciones parsimoniosas

Además, suponga que $\mathcal{A}$ es un FPRAS para g, y defina un algoritmo aleatorizado $\mathcal{B} : \Sigma^* \times (0,1) \to \mathbb{N}$ de la siguiente forma:

Para cada $w \in \Sigma^*$ y $\varepsilon \in (0,1)$: $\mathcal{B}(w, \varepsilon) = \mathcal{A}(h(w), \varepsilon)$

La existencia de FPRAS y las reducciones parsimoniosas

Además, suponga que $\mathcal{A}$ es un FPRAS para g, y defina un algoritmo aleatorizado $\mathcal{B} : \Sigma^* \times (0,1) \to \mathbb{N}$ de la siguiente forma:

Para cada $w \in \Sigma^*$ y $\varepsilon \in (0,1)$: $\mathcal{B}(w,\varepsilon) = \mathcal{A}(h(w),\varepsilon)$

Para cada $w \in \Sigma^*$ y $\varepsilon \in (0,1)$:

$$
\begin{aligned}
\mathbf{Pr}\Big(|\mathcal{B}(w,\varepsilon) - f(w)| \leq \varepsilon \cdot f(w)\Big) \\
&= \mathbf{Pr}\Big(|\mathcal{A}(h(w),\varepsilon) - g(h(w))| \leq \varepsilon \cdot g(h(w))\Big) \\
&\geq \frac{3}{4}
\end{aligned}
$$

La existencia de FPRAS y las reducciones parsimoniosas

Además, suponga que $\mathcal{A}$ es un FPRAS para g, y defina un algoritmo aleatorizado $\mathcal{B} : \Sigma^* \times (0,1) \to \mathbb{N}$ de la siguiente forma:

Para cada $w \in \Sigma^*$ y $\varepsilon \in (0,1)$: $\mathcal{B}(w, \varepsilon) = \mathcal{A}(h(w), \varepsilon)$

Para cada $w \in \Sigma^*$ y $\varepsilon \in (0,1)$:

$$
\begin{aligned}
\mathbf{Pr}\Big(|\mathcal{B}(w,\varepsilon) - f(w)| \leq \varepsilon \cdot f(w)\Big) \\
&= \mathbf{Pr}\Big(|\mathcal{A}(h(w),\varepsilon) - g(h(w))| \leq \varepsilon \cdot g(h(w))\Big) \\
&\geq \frac{3}{4}
\end{aligned}
$$

Por lo tanto $\mathcal{B}$ es un FPRAS para f $\square$

Utilizando las reducciones parsimoniosas

Suponiendo que $f \leq^{p}_{par} g$, el resultado anterior puede utilizarse de dos formas:

- Si tenemos un FPRAS para g, entonces podemos construir un FPRAS para f
- Si tenemos una demostración que no existe un FPRAS para f, entonces concluimos que no existe un FPRAS para g

Utilizando las reducciones parsimoniosas

Suponiendo que $f \leq^{p}_{par} g$, el resultado anterior puede utilizarse de dos formas:

- Si tenemos un FPRAS para g, entonces podemos construir un FPRAS para f
- Si tenemos una demostración que no existe un FPRAS para f, entonces concluimos que no existe un FPRAS para g

Vamos a encontrar otras funciones que no admiten FPRAS utilizando el resultado anterior.

Contando el número de cliques

Dado un grafo $G = (N, A)$, decimos que $S \subseteq N$ es un clique si para cada $a, b \in S$ se tiene que $(a, b) \in A$

Sea #CLIQUE una función que, dado un grafo G, retorna el número de cliques de G

Contando el número de cliques

Dado un grafo $G = (N, A)$, decimos que $S \subseteq N$ es un clique si para cada $a, b \in S$ se tiene que $(a, b) \in A$

Sea #CLIQUE una función que, dado un grafo G, retorna el número de cliques de G

Teorema

Si existe un FPRAS para #CLIQUE, entonces $NP = RP$

No existe un FPRAS para #CLIQUE: demostración

Dado un grafo $G = (N, A)$, defina $\overline{G} = (N, \overline{A})$ con $\overline{A} = (N \times N) \smallsetminus A$

No existe un FPRAS para #CLIQUE: demostración

Dado un grafo $G = (N, A)$, defina $\overline{G} = (N, \overline{A})$ con $\overline{A} = (N \times N) \smallsetminus A$

Tenemos que $\#\text{IS}(G) = \#\text{CLIQUE}(\overline{G})$

No existe un FPRAS para #CLIQUE: demostración

Dado un grafo $G = (N, A)$, defina $\overline{G} = (N, \overline{A})$ con $\overline{A} = (N \times N) \setminus A$

Tenemos que $\#\text{IS}(G) = \#\text{CLIQUE}(\overline{G})$

Concluimos que $\#\text{IS} \leq^{p}_{par} \#\text{CLIQUE}$

- Por lo tanto, si existe un FPRAS para #CLIQUE, entonces existe un FPRAS para #IS y NP $=$ RP

Otro ejemplo: las cláusulas de Horn

Recuerde que una cláusula se dice de Horn si contiene a lo más un literal positivo.

Ejemplo

$(p \vee \neg q \vee \neg r)$, p, $(\neg q \vee \neg r)$ y $\neg q$ son cláusulas de Horn, mientras que $(p \vee q)$ y $(p \vee \neg q \vee r \vee \neg s)$ no lo son.

Además, decimos que una fórmula proposicional φ es de Horn si φ es una conjunción de cláusulas de Horn.

La función #HORN-SAT

Sea #HORN-SAT una función que, dada una fórmula proposicional φ de Horn, retorna el número de valuaciones que satisfacen φ

La función #HORN-SAT

Sea #HORN-SAT una función que, dada una fórmula proposicional φ de Horn, retorna el número de valuaciones que satisfacen φ

Teorema

#HORN-SAT es #P-completo.

La función #HORN-SAT

Sea #HORN-SAT una función que, dada una fórmula proposicional φ de Horn, retorna el número de valuaciones que satisfacen φ

Teorema

#HORN-SAT es #P-completo.

Ejercicio

Demuestre que $L_{\#\text{HORN-SAT}} \in \text{PTIME}$

No existe un FPRAS para #HORN-SAT

Teorema

Si existe un FPRAS para #HORN-SAT, entonces NP = RP

No existe un FPRAS para #HORN-SAT

Teorema

Si existe un FPRAS para #HORN-SAT, entonces NP = RP

Demostración: Sea $G = (N, A)$ un grafo tal que $N \neq \emptyset$

No existe un FPRAS para #HORN-SAT

Teorema

Si existe un FPRAS para #HORN-SAT, entonces NP = RP

Demostración: Sea $G = (N, A)$ un grafo tal que $N \neq \emptyset$

Por cada $v \in N$, considere una variable proposicional x_v, y defina φ_G como la siguiente fórmula proposicional:

$$\left(\bigwedge_{a \in N} (x_a \vee \neg x_a) \right) \wedge \left(\bigwedge_{(b,c) \in A} (\neg x_b \vee \neg x_c) \right)$$

No existe un FPRAS para #HORN-SAT: demostración

Tenemos que φ_G es una fórmula proposicional de Horn y $\#\text{IS}(G) = \#\text{HORN-SAT}(\varphi_G)$

- ¿Cómo se puede extraer un conjunto independiente de G desde una valuación que satisface φ_G?

No existe un FPRAS para #HORN-SAT: demostración

Tenemos que φ_G es una fórmula proposicional de Horn y $\#\text{IS}(G) = \#\text{HORN-SAT}(\varphi_G)$

- ¿Cómo se puede extraer un conjunto independiente de G desde una valuación que satisface φ_G?

Concluimos que $\#\text{IS} \leq^p_{par} \#\text{HORN-SAT}$

- ¿Cómo se maneja el caso en que $N = \emptyset$ en la reducción?

No existe un FPRAS para #HORN-SAT: demostración

Por lo tanto, si existe un FPRAS para #HORN-SAT, entonces existe un FPRAS para #IS y NP = RP $\square$

No existe un FPRAS para #HORN-SAT: demostración

Por lo tanto, si existe un FPRAS para #HORN-SAT, entonces existe un FPRAS para #IS y NP $=$ RP $\square$

Comentario final

Dado que #IS $\leq^{p}_{par}$ #HORN-SAT, se tiene que #HORN-SAT es #P-hard

- Como #HORN-SAT $\in$ #P, concluimos que #HORN-SAT es #P-completo