¿Toda función en #P admite un FPRAS?

¿Toda función en #P admite un FPRAS?

Proposición

Si una función f admite un FPRAS, entonces $L_f \in BPP$

¿Toda función en #P admite un FPRAS?

Proposición

Si una función f admite un FPRAS, entonces $L_f \in BPP$

Demostración: Suponga que $f : \Sigma^* \to \mathbb{N}$, y sea $\mathcal{A} : \Sigma^* \times \mathbb{R}^+ \to \mathbb{N}$ un FPRAS para f

Dado $w \in \Sigma^*$, considere el siguiente algoritmo aleatorizado $\mathcal{B}$ para verificar si $w \in L_f$:

1. Sea k el resultado de ejecutar $\mathcal{A}(w, \frac{1}{2})$
2. Si $k > 0$ entonces retorne **sí**, en caso contrario retorne **no**

Una demostración de la proposición

Necesitamos calcular la probabilidad de error del algoritmo $\mathcal{B}$

Suponga primero que $w \in L_f$, vale decir, $f(w) > 0$

Dado que $\mathcal{A}$ es un FPRAS para f, tenemos que:

$$\mathbf{Pr}(|\mathcal{A}(w, \frac{1}{2}) - f(w)| \leq \frac{1}{2} \cdot f(w)) \quad \geq \quad \frac{3}{4}$$

Lo cual es equivalente a:

$$\mathbf{Pr}(\frac{1}{2} \cdot f(w) \leq \mathcal{A}(w, \frac{1}{2}) \leq \frac{3}{2} \cdot f(w)) \quad \geq \quad \frac{3}{4}$$

Una demostración de la proposición

Concluimos entonces que:

$$\mathbf{Pr}(\frac{1}{2} \cdot f(w) \leq \mathcal{A}(w, \frac{1}{2})) \quad \geq \quad \frac{3}{4}$$

Dado que $f(w) > 0$, tenemos que $\frac{1}{2} \cdot f(w) > 0$, por lo que tenemos lo siguiente:

$$\mathbf{Pr}(0 < \mathcal{A}(w, \frac{1}{2})) \quad \geq \quad \frac{3}{4}$$

Vale decir, $\mathbf{Pr}(\mathcal{B}$ retorne sí$) \geq \frac{3}{4}$

Una demostración de la proposición

Suponga ahora que $w \notin L_f$, vale decir, $f(w) = 0$

Sabemos que:

$$\mathbf{Pr}(\frac{1}{2} \cdot f(w) \leq \mathcal{A}(w, \frac{1}{2}) \leq \frac{3}{2} \cdot f(w)) \quad \geq \quad \frac{3}{4}$$

Dado que $f(w) = 0$, concluimos entonces que $\mathbf{Pr}(\mathcal{A}(w, \frac{1}{2}) = 0) \geq \frac{3}{4}$

Vale decir, $\mathbf{Pr}(\mathcal{B}$ retorne no$) \geq \frac{3}{4}$

Una demostración de la proposición

Suponga ahora que $w \notin L_f$, vale decir, $f(w) = 0$

Sabemos que:

$$\mathbf{Pr}(\frac{1}{2} \cdot f(w) \leq \mathcal{A}(w, \frac{1}{2}) \leq \frac{3}{2} \cdot f(w)) \quad \geq \quad \frac{3}{4}$$

Dado que $f(w) = 0$, concluimos entonces que $\mathbf{Pr}(\mathcal{A}(w, \frac{1}{2}) = 0) \geq \frac{3}{4}$

Vale decir, $\mathbf{Pr}(\mathcal{B}$ retorne no$) \geq \frac{3}{4}$

De los dos casos concluimos que $\mathbf{Pr}(\mathcal{B}$ retorne un resultado incorrecto$) \leq \frac{1}{4}$ $\square$

¿Qué funciones en #P tienen FPRAS?

Por la proposición anterior no esperamos tener FPRAS para #SAT y #CNF-SAT

- A menos que NP $\subseteq$ BPP

¿Qué funciones en #P tienen FPRAS?

Por la proposición anterior no esperamos tener FPRAS para #SAT y #CNF-SAT

- A menos que NP $\subseteq$ BPP

¿Cada función $f \in$ #P tal que $L_f \in$ BPP admite un FPRAS?

¿Qué funciones en #P tienen FPRAS?

Por la proposición anterior no esperamos tener FPRAS para #SAT y #CNF-SAT

- A menos que NP $\subseteq$ BPP

¿Cada función $f \in$ #P tal que $L_f \in$ BPP admite un FPRAS?

- ¿Al menos esto se cumple para cada función $g \in$ #P tal que $L_g \in$ PTIME?

¿Qué funciones en #P tienen FPRAS?

Por la proposición anterior no esperamos tener FPRAS para #SAT y #CNF-SAT

- A menos que NP $\subseteq$ BPP

¿Cada función $f \in$ #P tal que $L_f \in$ BPP admite un FPRAS?

- ¿Al menos esto se cumple para cada función $g \in$ #P tal que $L_g \in$ PTIME?

Para responder esta pregunta necesitamos considerar otros problemas #P-completos

Conjuntos independientes de un grafo

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

- Vale decir, los nodos mencionados en S no están conectados entre sí en el grafo

Sea $\text{GIS} = \{(G, k) \mid G$ tiene un conjunto independiente S con $|S| \geq k\}$

Ejercicio

Demuestre que GIS es NP-completo

Contando el número de conjuntos independientes

Sea $\#\text{GIS}$ una función que, dado un grafo G y un número natural k, retorna el número de conjuntos independientes S de G tales que $|S| \geq k$

Contando el número de conjuntos independientes

Sea #GIS una función que, dado un grafo G y un número natural k, retorna el número de conjuntos independientes S de G tales que $|S| \geq k$

Proposición

#GIS es #P-completo bajo reducciones parsimoniosas.

Una función de conteo más simple

Sea #LIS una función que, dado un grafo G y un número natural k, retorna el número de conjuntos independientes S de G tales que $|S| \leq k$

Una función de conteo más simple

Sea #LIS una función que, dado un grafo G y un número natural k, retorna el número de conjuntos independientes S de G tales que $|S| \leq k$

Proposición

#LIS es #P-completo.

Una función de conteo más simple

Sea #LIS una función que, dado un grafo G y un número natural k, retorna el número de conjuntos independientes S de G tales que $|S| \leq k$

Proposición

#LIS es #P-completo.

Ejercicio

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

- Es posible entonces que #LIS tenga un FPRAS

#LIS es #P-completo: demostración

Dado un grafo G con n nodos, y un número natural $k \geq 1$, se tiene que:

$$\#\text{GIS}(G, k) \quad = \quad \#\text{LIS}(G, n) - \#\text{LIS}(G, k - 1)$$

#LIS es #P-completo: demostración

Dado un grafo G con n nodos, y un número natural $k \geq 1$, se tiene que:

$$\#\text{GIS}(G, k) \quad = \quad \#\text{LIS}(G, n) - \#\text{LIS}(G, k - 1)$$

Concluimos entonces que $\#\text{GIS} \in \text{FP}^{\#\text{LIS}}$, vale decir, $\#\text{GIS} \leq_T^p \#\text{LIS}$

- Tenemos que #LIS es #P-hard
- Dado que $\#\text{LIS} \in \#\text{P}$, tenemos entonces que #LIS es #P-completo

$\square$

¿Existe un FPRAS para #LIS?

¿Existe un FPRAS para #LIS?

Vamos a dar una respuesta negativa a esta pregunta basada en una suposición de complejidad.

- Este resultado nos va a ayudar a identificar otros problemas que no admiten FPRAS

¿Existe un FPRAS para #LIS?

Vamos a dar una respuesta negativa a esta pregunta basada en una suposición de complejidad.

- Este resultado nos va a ayudar a identificar otros problemas que no admiten FPRAS

Teorema

Si existe un FPRAS para #LIS, entonces $NP \subseteq BPP$

#LIS no admite un FPRAS: demostración

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

- Dado que GIS es NP-completo, se concluye que NP $\subseteq$ BPP

#LIS no admite un FPRAS: demostración

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

- Dado que GIS es NP-completo, se concluye que NP $\subseteq$ BPP

Sea $G = (N, A)$ un grafo con $|N| = n$, y sea $k \geq 1$ un número natural.

- (G, k) es una entrada de GIS

#LIS no admite un FPRAS: demostración

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

- Dado que GIS es NP-completo, se concluye que NP $\subseteq$ BPP

Sea $G = (N, A)$ un grafo con $|N| = n$, y sea $k \geq 1$ un número natural.

- (G, k) es una entrada de GIS

Consideramos un número natural r, cuyo valor definiremos más tarde, y para cada $v \in N$ definimos un nuevo conjunto de r nodos:

$$C_v \quad = \quad \{v_1, v_2, \ldots, v_r\}$$

Técnica de demostración: construyendo un grafo aumentado

Usamos r para definir un nuevo grafo $G^r = (N^r, A^r)$:

$$
\begin{aligned}
N^r &= \bigcup_{v \in N} C_v \\
A^r &= \bigcup_{(u,v) \in A} \{(a, b) \mid a \in C_u \text{ y } b \in C_v\}
\end{aligned}
$$

G^r es un grafo con $r \cdot n$ nodos, donde dos nodos son adyacentes si y sólo si sus nodos de origen en G son adyacentes.

La noción de testigo

Definición

Dados conjuntos independientes S de G y T de G^r, decimos que T es un testigo para S si:

$$S \quad = \quad \{v \in N \mid C_v \cap T \neq \emptyset\}$$

La noción de testigo

Definición

Dados conjuntos independientes S de G y T de G^r, decimos que T es un testigo para S si:

$$S \quad = \quad \{v \in N \mid C_v \cap T \neq \emptyset\}$$

Un conjunto independiente de G^r es testigo de un único conjunto independiente de G

- Un conjunto independiente de G puede tener muchos testigos en G^r

La noción de testigo

Definición

Dados conjuntos independientes S de G y T de G^r, decimos que T es un testigo para S si:

$$S = \{v \in N \mid C_v \cap T \neq \emptyset\}$$

Un conjunto independiente de G^r es testigo de un único conjunto independiente de G

- Un conjunto independiente de G puede tener muchos testigos en G^r

Dado un conjunto independiente T de G^r, denotamos como S_T al único independiente de G que tiene como testigo a T

El número de testigos

Definimos los conjuntos:

$$
\begin{array}{rcl}
I^r(G,k) & = & \{T \mid T \text{ es un conjunto independiente de } G^r \text{ tal que } |S_T| = k\} \\
M^r(G,k) & = & \{T \mid T \text{ es un conjunto independiente de } G^r \text{ tal que } |S_T| < k\}
\end{array}
$$

El número de testigos

Definimos los conjuntos:

$$
\begin{array}{rcl}
I^r(G,k) & = & \{T \mid T \text{ es un conjunto independiente de } G^r \text{ tal que } |S_T| = k\} \\
M^r(G,k) & = & \{T \mid T \text{ es un conjunto independiente de } G^r \text{ tal que } |S_T| < k\}
\end{array}
$$

La conexión fundamental:

$$(G,k) \in \text{GIS si y sólo si } I^r(G,k) \neq \emptyset$$

El número de testigos

Definimos los conjuntos:

$$\begin{aligned} I^r(G,k) &= \{T \mid T \text{ es un conjunto independiente de } G^r \text{ tal que } |S_T| = k\} \\ M^r(G,k) &= \{T \mid T \text{ es un conjunto independiente de } G^r \text{ tal que } |S_T| < k\} \end{aligned}$$

La conexión fundamental:

$$(G,k) \in \text{GIS si y sólo si } I^r(G,k) \neq \emptyset$$

Suponiendo que existe un FPRAS para #LIS, vamos a desarrollar un algoritmo aleatorizado de tiempo polinomial para verificar si $I^r(G,k) \neq \emptyset$

- Obtenemos un algoritmo aleatorizado de tiempo polinomial para GIS

El número de testigos

Para $\ell \in \{0, 1, \ldots, k\}$ definimos t_ℓ como la cantidad de testigos en G^r para un conjunto independiente de G con ℓ nodos.

El número de testigos

Para $\ell \in \{0, 1, \ldots, k\}$ definimos t_ℓ como la cantidad de testigos en G^r para un conjunto independiente de G con ℓ nodos.

Por la definición de G^r tenemos que $t_\ell = (2^r - 1)^\ell$

- ¿Por qué?

El número de testigos

Para $\ell \in \{0, 1, \ldots, k\}$ definimos t_ℓ como la cantidad de testigos en G^r para un conjunto independiente de G con ℓ nodos.

Por la definición de G^r tenemos que $t_\ell = (2^r - 1)^\ell$

- ¿Por qué?

Además, G tiene a lo más $\binom{n}{\ell}$ conjuntos independientes con ℓ nodos

El número de testigos

A partir de lo anterior concluimos que:

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

Técnica de demostración: aumentando las diferencias

Por otra parte, si $(G, k) \in \text{GIS}$ tenemos que:

$$|I^r(G, k)| \ \geq \ t_k \ = \ (2^r - 1)^k$$

Técnica de demostración: aumentando las diferencias

Por otra parte, si $(G, k) \in$ GIS tenemos que:

$$|I^r(G,k)| \ \geq \ t_k \ = \ (2^r - 1)^k$$

Concluimos que:

- Si $(G, k) \not\in$ GIS: $\#\text{LIS}(G^r, k \cdot r) \ = \ |M^r(G,k)| \ \leq \ 2^n(2^r - 1)^{k-1}$
- Si $(G, k) \in$ GIS: $\#\text{LIS}(G^r, k \cdot r) \ \geq \ |I^r(G,k)| \ \geq \ (2^r - 1)^k$

Técnica de demostración: aumentando las diferencias

Por otra parte, si $(G, k) \in \text{GIS}$ tenemos que:

$$|I^r(G, k)| \ \geq \ t_k \ = \ (2^r - 1)^k$$

Concluimos que:

- Si $(G, k) \notin \text{GIS}$: $\#\text{LIS}(G^r, k \cdot r) \ = \ |M^r(G, k)| \ \leq \ 2^n(2^r - 1)^{k-1}$
- Si $(G, k) \in \text{GIS}$: $\#\text{LIS}(G^r, k \cdot r) \ \geq \ |I^r(G, k)| \ \geq \ (2^r - 1)^k$

En particular, eligiendo $r = n + 3$:

- Si $(G, k) \notin \text{GIS}$: $\#\text{LIS}(G^{n+3}, k \cdot (n + 3)) \ \leq \ 2^n(2^{n+3} - 1)^{k-1}$
- Si $(G, k) \in \text{GIS}$: $\#\text{LIS}(G^{n+3}, k \cdot (n + 3)) \ \geq \ (2^{n+3} - 1)^k$

Un algoritmo aleatorizado para GIS

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

Un algoritmo aleatorizado para GIS

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

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

1. Dada la entrada (G, k), donde G es un grafo con n nodos, si $k = 0$ retorne **sí**, en otro caso vaya al paso 2
2. Genere el grafo G^{n+3}
3. Sea s el resultado de ejecutar $\mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2})$
4. Si $s > (1 + \frac{1}{2}) \cdot 2^n (2^{n+3} - 1)^{k-1}$, entonces retorne **sí**, en caso contrario retorne **no**

Un algoritmo aleatorizado para GIS

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

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

1. Dada la entrada (G, k), donde G es un grafo con n nodos, si $k = 0$ retorne **sí**, en otro caso vaya al paso 2
2. Genere el grafo G^{n+3}
3. Sea s el resultado de ejecutar $\mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2})$
4. Si $s > (1 + \frac{1}{2}) \cdot 2^n(2^{n+3} - 1)^{k-1}$, entonces retorne **sí**, en caso contrario retorne **no**

Nótese que $\mathcal{B}$ funciona en tiempo polinomial en el tamaño de la entrada (G, k)

- Dado que el tamaño de G^{n+3} es polinomial en el tamaño de G (G^{n+3} tienes $n \cdot (n+3)$ nodos), $\mathcal{A}$ es un FPRAS y $\varepsilon = \frac{1}{2}$ cuando se ejecuta $\mathcal{A}$

La probabilidad de error del algoritmo

Si $(G, k) \notin \text{GIS}$ obtenemos que $\mathbf{Pr}(\mathcal{B}(G, k)$ retorne **sí**$)$ es igual a:

La probabilidad de error del algoritmo

Si $(G, k) \notin$ GIS obtenemos que $\mathbf{Pr}(\mathcal{B}(G, k)$ retorne **sí**$)$ es igual a:

$$\mathbf{Pr}(\mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2}) > (1 + \frac{1}{2}) \cdot 2^n (2^{n+3} - 1)^{k-1})$$

$$\leq \mathbf{Pr}(\mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2}) > (1 + \frac{1}{2}) \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)))$$

$$\leq \mathbf{Pr}(\mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2}) > (1 + \frac{1}{2}) \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)) \vee \mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2}) < (1 - \frac{1}{2}) \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)))$$

$$= 1 - \mathbf{Pr}(\mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2}) \leq (1 + \frac{1}{2}) \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)) \wedge \mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2}) \geq (1 - \frac{1}{2}) \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)))$$

$$= 1 - \mathbf{Pr}(|\mathcal{A}(G^{n+3}, k \cdot (n+3), \frac{1}{2}) - \#\text{LIS}(G^{n+3}, k \cdot (n+3))| \leq \frac{1}{2} \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)))$$

$$\leq 1 - \frac{3}{4} = \frac{1}{4}$$

La probabilidad de error del algoritmo

Consideramos ahora el caso en que $(G, k) \in$ GIS

La probabilidad de error del algoritmo

Consideramos ahora el caso en que $(G, k) \in \text{GIS}$

Primero debemos demostrar que:

$$(1 - \frac{1}{2}) \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)) \quad > \quad (1 + \frac{1}{2}) \cdot 2^n(2^{n+3} - 1)^{k-1}$$

Dado que $\#\text{LIS}(G^{n+3}, k \cdot (n+3)) \geq (2^{n+3} - 1)^k$, basta demostrar que:

$$(1 - \frac{1}{2}) \cdot (2^{n+3} - 1)^k \quad > \quad (1 + \frac{1}{2}) \cdot 2^n(2^{n+3} - 1)^{k-1}$$

La probabilidad de error del algoritmo

Tenemos que:

$$\begin{aligned}
& (1 - \frac{1}{2}) \cdot (2^{n+3} - 1)^k \;>\; (1 + \frac{1}{2}) \cdot 2^n (2^{n+3} - 1)^{k-1} \\
\Leftrightarrow \quad & \frac{1}{2} \cdot (2^{n+3} - 1)^k \;>\; \frac{3}{2} \cdot 2^n (2^{n+3} - 1)^{k-1} \\
\Leftrightarrow \quad & \frac{1}{2} \cdot (2^{n+3} - 1) \;>\; \frac{3}{2} \cdot 2^n \\
\Leftrightarrow \quad & 2^{n+3} - 1 \;>\; 3 \cdot 2^n \\
\Leftrightarrow \quad & 2^n (2^3 - 3) - 1 \;>\; 0 \\
\Leftrightarrow \quad & 5 \cdot 2^n - 1 \;>\; 0
\end{aligned}$$

Esta última condición es cierta para todo $n \geq 0$, de lo cual concluimos que $(1 - \frac{1}{2}) \cdot \#\text{LIS}(G^{n+3}, k \cdot (n+3)) > (1 + \frac{1}{2}) \cdot 2^n (2^{n+3} - 1)^{k-1}$

La probabilidad de error del algoritmo

Si $(G, k) \in$ GIS obtenemos que $\mathbf{Pr}(\mathcal{B}(G, k)$ retorne **no**) es igual a:

La probabilidad de error del algoritmo

Si $(G, k) \in \text{GIS}$ obtenemos que $\mathbf{Pr}(\mathcal{B}(G,k)$ retorne **no**) es igual a:

$$\mathbf{Pr}(\mathcal{A}(G^{n+3}, k\cdot(n+3), \frac{1}{2}) \leq (1+\frac{1}{2})\cdot 2^n(2^{n+3}-1)^{k-1})$$

$$\leq \mathbf{Pr}(\mathcal{A}(G^{n+3}, k\cdot(n+3), \frac{1}{2}) < (1-\frac{1}{2})\cdot \#\text{LIS}(G^{n+3}, k\cdot(n+3)))$$

$$\leq \mathbf{Pr}(\mathcal{A}(G^{n+3}, k\cdot(n+3), \frac{1}{2}) < (1-\frac{1}{2})\cdot \#\text{LIS}(G^{n+3}, k\cdot(n+3)) \vee$$
$$\mathcal{A}(G^{n+3}, k\cdot(n+3), \frac{1}{2}) > (1+\frac{1}{2})\cdot \#\text{LIS}(G^{n+3}, k\cdot(n+3)))$$

$$= 1 - \mathbf{Pr}(\mathcal{A}(G^{n+3}, k\cdot(n+3), \frac{1}{2}) \geq (1-\frac{1}{2})\cdot \#\text{LIS}(G^{n+3}, k\cdot(n+3)) \wedge$$
$$\mathcal{A}(G^{n+3}, k\cdot(n+3), \frac{1}{2}) \leq (1+\frac{1}{2})\cdot \#\text{LIS}(G^{n+3}, k\cdot(n+3)))$$

$$= 1 - \mathbf{Pr}(|\mathcal{A}(G^{n+3}, k\cdot(n+3), \frac{1}{2}) - \#\text{LIS}(G^{n+3}, k\cdot(n+3))| \leq$$
$$\frac{1}{2}\cdot \#\text{LIS}(G^{n+3}, k\cdot(n+3)))$$

$$\leq 1 - \frac{3}{4} = \frac{1}{4}$$

#LIS no admite un FPRAS: conclusión

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

Concluimos entonces que GIS $\in$ BPP

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

Utilizando la técnica anterior en otros problemas

Sea $\#$IS una función que, dado un grafo G, retorna el número de conjuntos independientes de G

Utilizando la técnica anterior en otros problemas

Sea $\#$IS una función que, dado un grafo G, retorna el número de conjuntos independientes de G

Teorema

Si existe un FPRAS para $\#IS$, entonces $NP \subseteq BPP$

Utilizando la técnica anterior en otros problemas

Sea $\#$IS una función que, dado un grafo G, retorna el número de conjuntos independientes de G

Teorema

Si existe un FPRAS para $\#$IS, entonces NP $\subseteq$ BPP

Ejercicio

Demuestre el teorema utilizando la técnica usada para el caso de $\#$LIS

Una solución del ejercicio

Al igual que para el caso de #LIS, demostramos que si existe un FPRAS para #IS, entonces GIS $\in$ BPP

- Dado que GIS es NP-completo, se concluye que NP $\subseteq$ BPP

Una solución del ejercicio

Al igual que para el caso de #LIS, demostramos que si existe un FPRAS para #IS, entonces GIS $\in$ BPP

- Dado que GIS es NP-completo, se concluye que NP $\subseteq$ BPP

Dada una entrada (G, k) de GIS, definimos G^r, $I^r(G, k)$ y $M^r(G, k)$ como para el caso de #LIS

Una solución del ejercicio

Al igual que para el caso de #LIS, demostramos que si existe un FPRAS para #IS, entonces GIS $\in$ BPP

- Dado que GIS es NP-completo, se concluye que NP $\subseteq$ BPP

Dada una entrada (G, k) de GIS, definimos G^r, $I^r(G, k)$ y $M^r(G, k)$ como para el caso de #LIS

Tenemos entonces que:

- Si $(G, k) \not\in$ GIS: $\#\text{IS}(G^r) \ = \ |M^r(G, k)| \ \leq \ 2^n(2^r - 1)^{k-1}$
- Si $(G, k) \in$ GIS: $\#\text{IS}(G^r) \ \geq \ |I^r(G, k)| \ \geq \ (2^r - 1)^k$

Una solución del ejercicio

Concluimos entonces que para $r = n + 3$:

- Si $(G, k) \notin$ GIS: $\#\text{IS}(G^{n+3}) \ \leq \ 2^n(2^{n+3} - 1)^{k-1}$
- Si $(G, k) \in$ GIS: $\#\text{IS}(G^{n+3}) \ \geq \ (2^{n+3} - 1)^k$

Una solución del ejercicio

Concluimos entonces que para $r = n + 3$:

- Si $(G, k) \notin \text{GIS}$: $\#\text{IS}(G^{n+3}) \ \leq \ 2^n(2^{n+3} - 1)^{k-1}$
- Si $(G, k) \in \text{GIS}$: $\#\text{IS}(G^{n+3}) \ \geq \ (2^{n+3} - 1)^k$

Al igual que para el caso de #LIS, suponemos que existe un FPRAS para #IS, y mostramos como esto puede ser utilizado para obtener que GIS $\in$ BPP

- El algoritmo aleatorizado de tiempo polinomial para GIS es definido de la misma forma que para el caso de #LIS
- Las cotas mencionadas arriba son utilizadas para demostrar que el error de este algoritmo está acotado superiormente por $\frac{1}{4}$ $\square$