

Un protocolo interactivo para conteo

Defina el siguiente lenguaje:

$$\text{COUNT-CNF-SAT}^{\leq} = \{(\varphi, k) \mid \varphi \text{ es una fórmula en CNF y} \\ \text{el número de valuaciones que satisface a } \varphi \text{ es menor o igual a } k\}$$

Un protocolo interactivo para conteo

Defina el siguiente lenguaje:

$$\text{COUNT-CNF-SAT}^{\leq} = \{(\varphi, k) \mid \varphi \text{ es una fórmula en CNF y} \\ \text{el número de valuaciones que satisface a } \varphi \text{ es menor o igual a } k\}$$

Y además considere la siguiente función que recibe como entrada a una fórmula φ en CNF:

$$\# \text{CNF-SAT}(\varphi) = |\{\sigma \mid \sigma(\varphi) = 1\}|$$

Un protocolo interactivo para conteo

Defina el siguiente lenguaje:

$$\text{COUNT-CNF-SAT}^{\leq} = \{(\varphi, k) \mid \varphi \text{ es una fórmula en CNF y} \\ \text{el número de valuaciones que satisface a } \varphi \text{ es menor o igual a } k\}$$

Y además considere la siguiente función que recibe como entrada a una fórmula φ en CNF:

$$\# \text{CNF-SAT}(\varphi) = |\{\sigma \mid \sigma(\varphi) = 1\}|$$

$\text{COUNT-CNF-SAT}^{\leq}$ y $\# \text{CNF-SAT}$ son polinomialmente equivalentes

- ▶ Si uno de los problemas se puede solucionar en tiempo polinomial, entonces el otro problema también

Un protocolo interactivo para conteo

Teorema

$COUNT-CNF-SAT^{\leq} \in IP[2n]$

Un protocolo interactivo para conteo

Teorema

$$COUNT-CNF-SAT^{\leq} \in IP[2n]$$

Ejercicio

Demuestre el teorema

La probabilidad de que el verificador sea engañado

En los protocolos aleatorizados anteriores, la probabilidad de que **V** sea engañado puede ser reducida a

$$\left(\frac{1}{4}\right)^\ell$$

para una constante ℓ arbitraria

La probabilidad de que el verificador sea engañado

En los protocolos aleatorizados anteriores, la probabilidad de que **V** sea engañado puede ser reducida a

$$\left(\frac{1}{4}\right)^\ell$$

para una constante ℓ arbitraria

Vamos a mostrar que esto se puede generalizar a cualquier lenguaje en IP

Un lema de amplificación para IP

Lema

Suponga que $\ell > 0$ y $L \in IP$. Entonces existe un verificador \mathbf{V} que funciona en tiempo polinomial (MT aleatorizada de tiempo polinomial) tal que para cada $w \in \Sigma^$:*

- ▶ *Si $w \in L$, entonces existe demostrador \mathbf{D} tal que*

$$\Pr((\mathbf{V}, \mathbf{D}) \text{ acepte } w) \geq 1 - \left(\frac{1}{4}\right)^\ell$$

- ▶ *Si $w \notin L$, entonces para todo demostrador \mathbf{D}' se tiene que*

$$\Pr((\mathbf{V}, \mathbf{D}') \text{ acepte } w) \leq \left(\frac{1}{4}\right)^\ell$$

Un lema de amplificación para IP

Lema

Suponga que $\ell > 0$ y $L \in IP$. Entonces existe un verificador \mathbf{V} que funciona en tiempo polinomial (MT aleatorizada de tiempo polinomial) tal que para cada $w \in \Sigma^$:*

- ▶ *Si $w \in L$, entonces existe demostrador \mathbf{D} tal que*

$$\Pr((\mathbf{V}, \mathbf{D}) \text{ acepte } w) \geq 1 - \left(\frac{1}{4}\right)^\ell$$

- ▶ *Si $w \notin L$, entonces para todo demostrador \mathbf{D}' se tiene que*

$$\Pr((\mathbf{V}, \mathbf{D}') \text{ acepte } w) \leq \left(\frac{1}{4}\right)^\ell$$

Ejercicio

Demuestre el lema

¿Cuál es el poder de IP?

Ya sabemos que $NP \subseteq IP$ y $co-NP \subseteq IP$

▶ ¿Por que se tiene que $NP \subseteq IP$?

¿Cuál es el poder de IP?

Ya sabemos que $NP \subseteq IP$ y $co-NP \subseteq IP$

▶ ¿Por que se tiene que $NP \subseteq IP$?

Además tenemos que $BPP \subseteq IP$

¿Cuál es el poder de IP?

Ya sabemos que $NP \subseteq IP$ y $co-NP \subseteq IP$

▶ ¿Por que se tiene que $NP \subseteq IP$?

Además tenemos que $BPP \subseteq IP$

▶ ¿Cómo se demuestra esto?

¿Cuál es el poder de IP?

¿Hay problemas en cada nivel de la jerarquía polinomial en IP? ¿Es cierto que $PSPACE \subseteq IP$? ¿En qué clase está contenido IP?

¿Cuál es el poder de IP?

¿Hay problemas en cada nivel de la jerarquía polinomial en IP? ¿Es cierto que $PSPACE \subseteq IP$? ¿En qué clase está contenido IP?

En las siguientes láminas vamos a caracterizar de manera precisa el poder de los protocolos interactivos.

Una caracterización de IP

Teorema (Shamir)

$$IP = PSPACE$$

Una caracterización de IP

Teorema (Shamir)

$$IP = PSPACE$$

Ejercicio

Demuestre que $IP \subseteq PSPACE$

- ▶ Para hacer esto, piense primero como demuestra directamente que $BPP \subseteq PSPACE$, sin utilizar el teorema de Gács-Sipser-Lautemann

Una caracterización de IP

Teorema (Shamir)

$$IP = PSPACE$$

Ejercicio

Demuestre que $IP \subseteq PSPACE$

- ▶ Para hacer esto, piense primero como demuestra directamente que $BPP \subseteq PSPACE$, sin utilizar el teorema de Gács-Sipser-Lautemann

A continuación vamos a demostrar que $PSPACE \subseteq IP$

Dos problemas PSPACE-completos

Recuerde que una formula proposicional cuantificada es de la forma:

$$Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n),$$

donde cada $Q_i \in \{\exists, \forall\}$ y $\psi(x_1, \dots, x_n)$ es una fórmula proposicional cuyas variables son x_1, \dots, x_n

Dos problemas PSPACE-completos

Recuerde que una formula proposicional cuantificada es de la forma:

$$Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n),$$

donde cada $Q_i \in \{\exists, \forall\}$ y $\psi(x_1, \dots, x_n)$ es una fórmula proposicional cuyas variables son x_1, \dots, x_n

Por ejemplo, las siguientes son fórmulas proposicionales cuantificadas:

$$\forall x \exists y x \wedge y$$

$$\forall x \exists y x \vee y$$

Dos problemas PSPACE-completos

El problema QBF recibe como entrada una fórmula proposicional cuantificada, y verifica si esta fórmula es cierta

Dos problemas PSPACE-completos

El problema QBF recibe como entrada una fórmula proposicional cuantificada, y verifica si esta fórmula es cierta

▶ ¿Es $\forall x \exists y x \wedge y$ cierta?

Dos problemas PSPACE-completos

El problema QBF recibe como entrada una fórmula proposicional cuantificada, y verifica si esta fórmula es cierta

▶ ¿Es $\forall x \exists y x \wedge y$ cierta? No

Dos problemas PSPACE-completos

El problema QBF recibe como entrada una fórmula proposicional cuantificada, y verifica si esta fórmula es cierta

- ▶ ¿Es $\forall x \exists y x \wedge y$ cierta? No
- ▶ ¿Es $\forall x \exists y x \vee y$ cierta?

Dos problemas PSPACE-completos

El problema QBF recibe como entrada una fórmula proposicional cuantificada, y verifica si esta fórmula es cierta

- ▶ ¿Es $\forall x \exists y x \wedge y$ cierta? No
- ▶ ¿Es $\forall x \exists y x \vee y$ cierta? Sí

Dos problemas PSPACE-completos

El problema QBF recibe como entrada una fórmula proposicional cuantificada, y verifica si esta fórmula es cierta

- ▶ ¿Es $\forall x \exists y x \wedge y$ cierta? No
- ▶ ¿Es $\forall x \exists y x \vee y$ cierta? Sí

QBF restringido al cuantificador \exists corresponde a SAT

Dos problemas PSPACE-completos

El problema QBF recibe como entrada una fórmula proposicional cuantificada, y verifica si esta fórmula es cierta

- ▶ ¿Es $\forall x \exists y x \wedge y$ cierta? No
- ▶ ¿Es $\forall x \exists y x \vee y$ cierta? Sí

QBF restringido al cuantificador \exists corresponde a SAT

- ▶ Y además vimos que para cada nivel Σ_k^P ($k \geq 1$) de la jerarquía polinomial, hay una restricción de QBF que es Σ_k^P -completo

Dos problemas PSPACE-completos

Teorema

QBF es PSPACE-completo

Dos problemas PSPACE-completos

Teorema

QBF es PSPACE-completo

Definimos CNF-QBF como el problema QBF restringido a las fórmulas

$$Q_1x_1 \cdots Q_nx_n \psi(x_1, \dots, x_n)$$

donde $\psi(x_1, \dots, x_n)$ está en CNF

Dos problemas PSPACE-completos

Teorema

QBF es PSPACE-completo

Definimos CNF-QBF como el problema QBF restringido a las fórmulas

$$Q_1x_1 \cdots Q_nx_n \psi(x_1, \dots, x_n)$$

donde $\psi(x_1, \dots, x_n)$ está en CNF

Teorema

CNF-QBF es PSPACE-completo

$\text{PSPACE} \subseteq \text{IP}$

Teorema

CNF-QBF está en $\text{IP}[n^2 + n]$

$PSPACE \subseteq IP$

Teorema

$CNF\text{-}QBF$ está en $IP[n^2 + n]$

Corolario

$PSPACE \subseteq IP$

CNF-QBF está en $IP[n^2 + n]$

Sea φ la siguiente fórmula en CNF-QBF:

$$Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n),$$

donde $\psi(x_1, \dots, x_n) = C_1 \wedge \cdots \wedge C_m$ es una fórmula en CNF cuyas variables son x_1, \dots, x_n

CNF-QBF está en $IP[n^2 + n]$

Sea φ la siguiente fórmula en CNF-QBF:

$$Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n),$$

donde $\psi(x_1, \dots, x_n) = C_1 \wedge \cdots \wedge C_m$ es una fórmula en CNF cuyas variables son x_1, \dots, x_n

Suponemos que:

- ▶ Cada cláusula en $\psi(x_1, \dots, x_n)$ no tiene literales complementarios ni repetidos
- ▶ $m \geq 2$

CNF-QBF está en $IP[n^2 + n]$

Sea φ la siguiente fórmula en CNF-QBF:

$$Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n),$$

donde $\psi(x_1, \dots, x_n) = C_1 \wedge \cdots \wedge C_m$ es una fórmula en CNF cuyas variables son x_1, \dots, x_n

Suponemos que:

- ▶ Cada cláusula en $\psi(x_1, \dots, x_n)$ no tiene literales complementarios ni repetidos
- ▶ $m \geq 2$
 - ▶ Si $m = 1$ simplemente repetimos la cláusula para obtener $m = 2$

CNF-QBF está en $IP[n^2 + n]$

Al igual que para la demostración de que COUNT-CNF-SAT está en $IP[2n]$:

CNF-QBF está en $IP[n^2 + n]$

Al igual que para la demostración de que COUNT-CNF-SAT está en $IP[2n]$:

► Para cada literal ℓ , defina

$$\tau_\ell = \begin{cases} (1 - x_i) & \ell = x_i \\ x_i & \ell = \neg x_i \end{cases}$$

CNF-QBF está en $IP[n^2 + n]$

Al igual que para la demostración de que COUNT-CNF-SAT está en $IP[2n]$:

- ▶ Para cada literal ℓ , defina

$$\tau_{\ell} = \begin{cases} (1 - x_i) & \ell = x_i \\ x_i & \ell = \neg x_i \end{cases}$$

- ▶ Para cada cláusula $C = (\ell_1 \vee \cdots \vee \ell_k)$, defina

$$\tau_C = 1 - \prod_{i=1}^k \tau_{\ell_i}$$

CNF-QBF está en $IP[n^2 + n]$

Al igual que para la demostración de que COUNT-CNF-SAT está en $IP[2n]$:

- ▶ Para cada literal ℓ , defina

$$\tau_{\ell} = \begin{cases} (1 - x_i) & \ell = x_i \\ x_i & \ell = \neg x_i \end{cases}$$

- ▶ Para cada cláusula $C = (\ell_1 \vee \cdots \vee \ell_k)$, defina

$$\tau_C = 1 - \prod_{i=1}^k \tau_{\ell_i}$$

- ▶ Y defina

$$g(x_1, \dots, x_n) = \prod_{i=1}^m \tau_{C_i}$$

CNF-QBF está en $IP[n^2 + n]$

Recuerde que para cada valuación $\sigma : \{x_1, \dots, x_n\} \rightarrow \{0, 1\}$, tenemos que:

- ▶ Si $\sigma(\varphi) = 1$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 1$
- ▶ Si $\sigma(\varphi) = 0$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 0$

CNF-QBF está en $IP[n^2 + n]$

Recuerde que para cada valuación $\sigma : \{x_1, \dots, x_n\} \rightarrow \{0, 1\}$, tenemos que:

- ▶ Si $\sigma(\varphi) = 1$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 1$
- ▶ Si $\sigma(\varphi) = 0$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 0$

En el caso de la demostración de que COUNT-CNF-SAT está en $IP[2n]$ usamos la siguiente condición:

$$\sum_{(a_1, \dots, a_n) \in \{0, 1\}^n} g(a_1, \dots, a_n) = k$$

CNF-QBF está en $IP[n^2 + n]$

Recuerde que para cada valuación $\sigma : \{x_1, \dots, x_n\} \rightarrow \{0, 1\}$, tenemos que:

- ▶ Si $\sigma(\varphi) = 1$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 1$
- ▶ Si $\sigma(\varphi) = 0$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 0$

En el caso de la demostración de que COUNT-CNF-SAT está en $IP[2n]$ usamos la siguiente condición:

$$\sum_{(a_1, \dots, a_n) \in \{0, 1\}^n} g(a_1, \dots, a_n) = k$$

Para CNF-QBF nos gustaría usar una condición similar donde $\exists x_i$ corresponde a una suma y $\forall x_i$ corresponde a una multiplicación

CNF-QBF está en $IP[n^2 + n]$

Recuerde que para cada valuación $\sigma : \{x_1, \dots, x_n\} \rightarrow \{0, 1\}$, tenemos que:

- ▶ Si $\sigma(\varphi) = 1$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 1$
- ▶ Si $\sigma(\varphi) = 0$, entonces $g(\sigma(x_1), \dots, \sigma(x_n)) = 0$

En el caso de la demostración de que COUNT-CNF-SAT está en $IP[2n]$ usamos la siguiente condición:

$$\sum_{(a_1, \dots, a_n) \in \{0, 1\}^n} g(a_1, \dots, a_n) = k$$

Para CNF-QBF nos gustaría usar una condición similar donde $\exists x_i$ corresponde a una suma y $\forall x_i$ corresponde a una multiplicación

- ▶ ¿Pero cómo se interpreta la salida de la expresión?

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \wedge y$. En este caso tenemos que:

$$g(x, y) = x \cdot y$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \wedge y$. En este caso tenemos que:

$$g(x, y) = x \cdot y$$

Tenemos entonces que:

$$\begin{aligned} \prod_{a \in \{0,1\}} \sum_{b \in \{0,1\}} g(a, b) &= (g(0, 0) + g(0, 1)) \cdot (g(1, 0) + g(1, 1)) \\ &= (0 + 0) \cdot (0 + 1) \\ &= 0 \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \wedge y$. En este caso tenemos que:

$$g(x, y) = x \cdot y$$

Tenemos entonces que:

$$\begin{aligned} \prod_{a \in \{0,1\}} \sum_{b \in \{0,1\}} g(a, b) &= (g(0, 0) + g(0, 1)) \cdot (g(1, 0) + g(1, 1)) \\ &= (0 + 0) \cdot (0 + 1) \\ &= 0 \end{aligned}$$

Obtenemos el valor 0 que representa que la fórmula no es cierta

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \vee y$. En este caso tenemos que:

$$g(x, y) = 1 - (1 - x) \cdot (1 - y)$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \vee y$. En este caso tenemos que:

$$g(x, y) = 1 - (1 - x) \cdot (1 - y)$$

Tenemos entonces que:

$$\begin{aligned} \prod_{a \in \{0,1\}} \sum_{b \in \{0,1\}} g(a, b) &= (g(0, 0) + g(0, 1)) \cdot (g(1, 0) + g(1, 1)) \\ &= (0 + 1) \cdot (1 + 1) \\ &= 2 \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \vee y$. En este caso tenemos que:

$$g(x, y) = 1 - (1 - x) \cdot (1 - y)$$

Tenemos entonces que:

$$\begin{aligned} \prod_{a \in \{0,1\}} \sum_{b \in \{0,1\}} g(a, b) &= (g(0, 0) + g(0, 1)) \cdot (g(1, 0) + g(1, 1)) \\ &= (0 + 1) \cdot (1 + 1) \\ &= 2 \end{aligned}$$

Obtenemos el valor 2 que representa que la fórmula es cierta

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \vee y$. En este caso tenemos que:

$$g(x, y) = 1 - (1 - x) \cdot (1 - y)$$

Tenemos entonces que:

$$\begin{aligned} \prod_{a \in \{0,1\}} \sum_{b \in \{0,1\}} g(a, b) &= (g(0,0) + g(0,1)) \cdot (g(1,0) + g(1,1)) \\ &= (0 + 1) \cdot (1 + 1) \\ &= 2 \end{aligned}$$

Obtenemos el valor 2 que representa que la fórmula es cierta

► El valor es mayor que 0. ¿Pero que representa?

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x_1 \cdots \forall x_n \exists x_{n+1} x_1 \vee \cdots \vee x_n \vee x_{n+1}$. En este caso tenemos que:

$$g(x_1, \dots, x_n, x_{n+1}) = 1 - (1 - x_1) \cdot \dots \cdot (1 - x_n) \cdot (1 - x_{n+1})$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x_1 \cdots \forall x_n \exists x_{n+1} x_1 \vee \cdots \vee x_n \vee x_{n+1}$. En este caso tenemos que:

$$g(x_1, \dots, x_n, x_{n+1}) = 1 - (1 - x_1) \cdot \dots \cdot (1 - x_n) \cdot (1 - x_{n+1})$$

Es posible demostrar que:

$$\prod_{a_1 \in \{0,1\}} \cdots \prod_{a_n \in \{0,1\}} \sum_{a_{n+1} \in \{0,1\}} g(a_1, \dots, a_n, a_{n+1}) = 2^{2^n - 1}$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x_1 \cdots \forall x_n \exists x_{n+1} x_1 \vee \cdots \vee x_n \vee x_{n+1}$. En este caso tenemos que:

$$g(x_1, \dots, x_n, x_{n+1}) = 1 - (1 - x_1) \cdot \dots \cdot (1 - x_n) \cdot (1 - x_{n+1})$$

Es posible demostrar que:

$$\prod_{a_1 \in \{0,1\}} \cdots \prod_{a_n \in \{0,1\}} \sum_{a_{n+1} \in \{0,1\}} g(a_1, \dots, a_n, a_{n+1}) = 2^{2^n - 1}$$

Obtenemos un valor mayor que 0 que representa que la fórmula es cierta

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x_1 \cdots \forall x_n \exists x_{n+1} x_1 \vee \cdots \vee x_n \vee x_{n+1}$. En este caso tenemos que:

$$g(x_1, \dots, x_n, x_{n+1}) = 1 - (1 - x_1) \cdot \dots \cdot (1 - x_n) \cdot (1 - x_{n+1})$$

Es posible demostrar que:

$$\prod_{a_1 \in \{0,1\}} \cdots \prod_{a_n \in \{0,1\}} \sum_{a_{n+1} \in \{0,1\}} g(a_1, \dots, a_n, a_{n+1}) = 2^{2^n - 1}$$

Obtenemos un valor mayor que 0 que representa que la fórmula es cierta

- ▶ Pero este número tiene 2^n dígitos en binario, por lo que el demostrador no se lo puede enviar al verificador

CNF-QBF está en $IP[n^2 + n]$: operadores $\exists x_i$ y $\forall x_i$

La solución al primer problema:

CNF-QBF está en $IP[n^2 + n]$: operadores $\exists x_i$ y $\forall x_i$

La solución al primer problema:

- ▶ $\exists x_i$ es considerado un operador que elimina la variable x_i a través del siguiente cálculo:

$$\begin{aligned} \exists x_i g(x_1, \dots, x_n) = & \\ & g(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) + \\ & g(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) - \\ & g(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \cdot g(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$: operadores $\exists x_i$ y $\forall x_i$

La solución al primer problema:

- ▶ $\exists x_i$ es considerado un operador que elimina la variable x_i a través del siguiente cálculo:

$$\begin{aligned}\exists x_i g(x_1, \dots, x_n) = & \\ & g(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) + \\ & g(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) - \\ & g(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \cdot g(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)\end{aligned}$$

- ▶ $\forall x_i$ es considerado un operador que elimina la variable x_i a través del siguiente cálculo:

$$\begin{aligned}\forall x_i g(x_1, \dots, x_n) = & \\ & g(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \cdot g(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)\end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$

Tenemos entonces que la expresión

$$Q_1 x_1 \cdots Q_n x_n g(x_1, \dots, x_n)$$

es igual a 0 o 1

CNF-QBF está en $IP[n^2 + n]$

Tenemos entonces que la expresión

$$Q_1 x_1 \cdots Q_n x_n g(x_1, \dots, x_n)$$

es igual a 0 o 1

El valor 0 significa que la fórmula $Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n)$ no es cierta, y el valor 1 que $Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n)$ es cierta

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \wedge y$. En este caso tenemos que:

$$g(x, y) = x \cdot y$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \wedge y$. En este caso tenemos que:

$$g(x, y) = x \cdot y$$

Tenemos entonces que:

$$\begin{aligned}\forall x \exists y g(x, y) &= (g(0, 0) + g(0, 1) - g(0, 0) \cdot g(0, 1)) \cdot \\ &\quad (g(1, 0) + g(1, 1) - g(1, 0) \cdot g(1, 1)) \\ &= (0 + 0 - 0) \cdot (0 + 1 - 0) \\ &= 0\end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \wedge y$. En este caso tenemos que:

$$g(x, y) = x \cdot y$$

Tenemos entonces que:

$$\begin{aligned}\forall x \exists y g(x, y) &= (g(0, 0) + g(0, 1) - g(0, 0) \cdot g(0, 1)) \cdot \\ &\quad (g(1, 0) + g(1, 1) - g(1, 0) \cdot g(1, 1)) \\ &= (0 + 0 - 0) \cdot (0 + 1 - 0) \\ &= 0\end{aligned}$$

Obtenemos el valor 0 que representa que la fórmula no es cierta

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \vee y$. En este caso tenemos que:

$$g(x, y) = 1 - (1 - x) \cdot (1 - y)$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \vee y$. En este caso tenemos que:

$$g(x, y) = 1 - (1 - x) \cdot (1 - y)$$

Tenemos entonces que:

$$\begin{aligned}\forall x \exists y g(x, y) &= (g(0, 0) + g(0, 1) - g(0, 0) \cdot g(0, 1)) \cdot \\ &\quad (g(1, 0) + g(1, 1) - g(1, 0) \cdot g(1, 1)) \\ &= (0 + 1 - 0) \cdot (1 + 1 - 1) \\ &= 1\end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$

Ejemplo

Considere la fórmula $\forall x \exists y x \vee y$. En este caso tenemos que:

$$g(x, y) = 1 - (1 - x) \cdot (1 - y)$$

Tenemos entonces que:

$$\begin{aligned}\forall x \exists y g(x, y) &= (g(0, 0) + g(0, 1) - g(0, 0) \cdot g(0, 1)) \cdot \\ &\quad (g(1, 0) + g(1, 1) - g(1, 0) \cdot g(1, 1)) \\ &= (0 + 1 - 0) \cdot (1 + 1 - 1) \\ &= 1\end{aligned}$$

Obtenemos el valor 1 que representa que la fórmula es cierta

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

Ya sabemos cuál es la condición de la cual el demostrador debe convencer al verificador

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

Ya sabemos cuál es la condición de la cual el demostrador debe convencer al verificador

Pero nos queda un problema por solucionar en la definición del protocolo

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

Considere la fórmula $\exists x_1 \cdots \exists x_n \psi(x_1, \dots, x_n)$, y recuerde que $\psi(x_1, \dots, x_n)$ es una fórmula en CNF con m cláusulas

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

Considere la fórmula $\exists x_1 \cdots \exists x_n \psi(x_1, \dots, x_n)$, y recuerde que $\psi(x_1, \dots, x_n)$ es una fórmula en CNF con m cláusulas

- ▶ Además, considere que el polinomio construido desde $\psi(x_1, \dots, x_n)$ es $g(x_1, \dots, x_n)$

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

Considere la fórmula $\exists x_1 \cdots \exists x_n \psi(x_1, \dots, x_n)$, y recuerde que $\psi(x_1, \dots, x_n)$ es una fórmula en CNF con m cláusulas

- ▶ Además, considere que el polinomio construido desde $\psi(x_1, \dots, x_n)$ es $g(x_1, \dots, x_n)$

En el protocolo para esta fórmula vamos a usar el siguiente polinomio:

$$h_1(x_1) = \exists x_2 \cdots \exists x_n g(x_1, x_2, \dots, x_n)$$

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

Considere la fórmula $\exists x_1 \cdots \exists x_n \psi(x_1, \dots, x_n)$, y recuerde que $\psi(x_1, \dots, x_n)$ es una fórmula en CNF con m cláusulas

- ▶ Además, considere que el polinomio construido desde $\psi(x_1, \dots, x_n)$ es $g(x_1, \dots, x_n)$

En el protocolo para esta fórmula vamos a usar el siguiente polinomio:

$$h_1(x_1) = \exists x_2 \cdots \exists x_n g(x_1, x_2, \dots, x_n)$$

¿Puede dar una cota para el grado del polinomio $h_1(x_1)$?

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

Considere la fórmula $\exists x_1 \cdots \exists x_n \psi(x_1, \dots, x_n)$, y recuerde que $\psi(x_1, \dots, x_n)$ es una fórmula en CNF con m cláusulas

- ▶ Además, considere que el polinomio construido desde $\psi(x_1, \dots, x_n)$ es $g(x_1, \dots, x_n)$

En el protocolo para esta fórmula vamos a usar el siguiente polinomio:

$$h_1(x_1) = \exists x_2 \cdots \exists x_n g(x_1, x_2, \dots, x_n)$$

¿Puede dar una cota para el grado del polinomio $h_1(x_1)$?

- ▶ El grado de $h_1(x_1)$ está acotado por $m \cdot 2^n$

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

El demostrador no puede enviar un polinomio de grado $m \cdot 2^n$

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

El demostrador no puede enviar un polinomio de grado $m \cdot 2^n$

- ▶ Un polinomio de grado exponencial puede tener un número exponencial de coeficientes

CNF-QBF está en $IP[n^2 + n]$: un segundo problema

El demostrador no puede enviar un polinomio de grado $m \cdot 2^n$

- ▶ Un polinomio de grado exponencial puede tener un número exponencial de coeficientes

¿Cómo podemos reducir el grado de los polinomios que vamos a construir?

CNF-QBF está en $IP[n^2 + n]$: linearización

Para solucionar el segundo problema introducimos un operador de linearización

CNF-QBF está en $IP[n^2 + n]$: linearización

Para solucionar el segundo problema introducimos un operador de linearización

- ▶ Este operador no elimina una variable

CNF-QBF está en $IP[n^2 + n]$: linearización

Para solucionar el segundo problema introducimos un operador de linearización

- ▶ Este operador no elimina una variable
- ▶ El resultado de aplicar el operador sobre una variable x_i es un polinomio lineal en x_i

CNF-QBF está en $IP[n^2 + n]$: linearización

Para solucionar el segundo problema introducimos un operador de linearización

- ▶ Este operador no elimina una variable
- ▶ El resultado de aplicar el operador sobre una variable x_i es un polinomio lineal en x_i

El operador L_{x_i} se define de la siguiente forma:

$$\begin{aligned} L_{x_i} g(x_1, \dots, x_n) = & \\ & (1 - x_i) \cdot g(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) + \\ & x_i \cdot g(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$: linearización

Sea $h(x_1, \dots, x_n) = Lx_i g(x_1, \dots, x_n)$

► Note que $h(x_1, \dots, x_n)$ tiene las mismas variables que $g(x_1, \dots, x_n)$

CNF-QBF está en $IP[n^2 + n]$: linearización

Sea $h(x_1, \dots, x_n) = Lx_i g(x_1, \dots, x_n)$

► Note que $h(x_1, \dots, x_n)$ tiene las mismas variables que $g(x_1, \dots, x_n)$

Tenemos que:

$$h(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) = g(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n)$$

$$h(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) = g(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$$

CNF-QBF está en $IP[n^2 + n]$: la condición a verificar

De la propiedad anterior concluimos que:

$Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \psi(x_1, \dots, x_n)$ es cierta

CNF-QBF está en $IP[n^2 + n]$: la condición a verificar

De la propiedad anterior concluimos que:

$$\begin{aligned} Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \psi(x_1, \dots, x_n) \text{ es cierta} \\ \Leftrightarrow \\ Q_1 x_1 Q_2 x_2 \cdots Q_n x_n g(x_1, \dots, x_n) = 1 \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$: la condición a verificar

De la propiedad anterior concluimos que:

$$\begin{aligned} Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \psi(x_1, \dots, x_n) \text{ es cierta} \\ \Leftrightarrow \\ Q_1 x_1 Q_2 x_2 \cdots Q_n x_n g(x_1, \dots, x_n) = 1 \\ \Leftrightarrow \\ Q_1 x_1 L x_1 Q_2 x_2 L x_1 L x_2 \cdots Q_{n-1} x_{n-1} L x_1 \cdots L x_{n-1} Q_n x_n g(x_1, \dots, x_n) = 1 \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$: la condición a verificar

De la propiedad anterior concluimos que:

$$\begin{aligned} Q_1 x_1 Q_2 x_2 \cdots Q_n x_n \psi(x_1, \dots, x_n) \text{ es cierta} \\ \Leftrightarrow \\ Q_1 x_1 Q_2 x_2 \cdots Q_n x_n g(x_1, \dots, x_n) = 1 \\ \Leftrightarrow \\ Q_1 x_1 Lx_1 Q_2 x_2 Lx_1 Lx_2 \cdots Q_{n-1} x_{n-1} Lx_1 \cdots Lx_{n-1} Q_n x_n g(x_1, \dots, x_n) = 1 \end{aligned}$$

Por lo tanto, el demostrador debe convencer al verificador que la siguiente propiedad es cierta:

$$Q_1 x_1 Lx_1 Q_2 x_2 Lx_1 Lx_2 \cdots Q_n x_n g(x_1, \dots, x_n) = 1$$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

La entrada del protocolo es $\varphi = Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n)$, el cual es transformado en la siguiente expresión:

$$Q_1 x_1 L x_1 Q_2 x_2 L x_1 L x_2 \cdots Q_n x_n g(x_1, \dots, x_n)$$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

La entrada del protocolo es $\varphi = Q_1 x_1 \cdots Q_n x_n \psi(x_1, \dots, x_n)$, el cual es transformado en la siguiente expresión:

$$Q_1 x_1 L x_1 Q_2 x_2 L x_1 L x_2 \cdots Q_n x_n g(x_1, \dots, x_n)$$

Tenemos $n + \frac{n(n-1)}{2}$ operadores en la expresión, a la cual denotamos como

$$O_1 O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n),$$

donde cada O_i representa a $Q_j x_j$ o $L x_k$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado
2. **D** le devuelve a **V** un polinomio $h_1(x_1)$ tal que

$$h_1(x_1) = O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n)$$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado

2. **D** le devuelve a **V** un polinomio $h_1(x_1)$ tal que

$$h_1(x_1) = O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n)$$

3. Si el grado de $h_1(x_1)$ es mayor que $2m$ entonces **V** rechaza

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado

2. **D** le devuelve a **V** un polinomio $h_1(x_1)$ tal que

$$h_1(x_1) = O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n)$$

3. Si el grado de $h_1(x_1)$ es mayor que $2m$ entonces **V** rechaza

4. **V** verifica si uno de los siguientes casos se cumple, y si no es así entonces rechaza

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado

2. **D** le devuelve a **V** un polinomio $h_1(x_1)$ tal que

$$h_1(x_1) = O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n)$$

3. Si el grado de $h_1(x_1)$ es mayor que $2m$ entonces **V** rechaza

4. **V** verifica si uno de los siguientes casos se cumple, y si no es así entonces rechaza

4.1 $O_1 = \exists x_1$ y $h_1(0) + h_1(1) = 1$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado

2. **D** le devuelve a **V** un polinomio $h_1(x_1)$ tal que

$$h_1(x_1) = O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n)$$

3. Si el grado de $h_1(x_1)$ es mayor que $2m$ entonces **V** rechaza

4. **V** verifica si uno de los siguientes casos se cumple, y si no es así entonces rechaza

4.1 $O_1 = \exists x_1$ y $h_1(0) + h_1(1) = 1$

4.2 $O_1 = \forall x_1$ y $h_1(0) \cdot h_1(1) = 1$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado

2. **D** le devuelve a **V** un polinomio $h_1(x_1)$ tal que

$$h_1(x_1) = O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n)$$

3. Si el grado de $h_1(x_1)$ es mayor que $2m$ entonces **V** rechaza

4. **V** verifica si uno de los siguientes casos se cumple, y si no es así entonces rechaza

4.1 $O_1 = \exists x_1$ y $h_1(0) + h_1(1) = 1$

4.2 $O_1 = \forall x_1$ y $h_1(0) \cdot h_1(1) = 1$

5. **V** define el contador $i = 1$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

El protocolo funciona de la siguiente forma:

1. **V** le indica a **D** que el protocolo ha comenzado

2. **D** le devuelve a **V** un polinomio $h_1(x_1)$ tal que

$$h_1(x_1) = O_2 \cdots O_{n + \frac{n(n-1)}{2}} g(x_1, \dots, x_n)$$

3. Si el grado de $h_1(x_1)$ es mayor que $2m$ entonces **V** rechaza

4. **V** verifica si uno de los siguientes casos se cumple, y si no es así entonces rechaza

4.1 $O_1 = \exists x_1$ y $h_1(0) + h_1(1) = 1$

4.2 $O_1 = \forall x_1$ y $h_1(0) \cdot h_1(1) = 1$

5. **V** define el contador $i = 1$

6. **V** genera al azar con distribución uniforme un número entero $s_1 \in \{0, \dots, 2^{(n^2+n)m} - 1\}$, y se lo envía a **D**

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7. Los siguientes pasos se repiten para $j = 2, \dots, n + \frac{n(n-1)}{2}$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7. Los siguientes pasos se repiten para $j = 2, \dots, n + \frac{n(n-1)}{2}$

7.1 **D** le devuelve a **V** un polinomio $h_j(x_i)$ tal que

$$h_j(x_i) = O_{j+1} \cdots O_{n + \frac{n(n-1)}{2}} g(r_1, \dots, r_{i-1}, x_i, \dots, x_n)$$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7. Los siguientes pasos se repiten para $j = 2, \dots, n + \frac{n(n-1)}{2}$

7.1 **D** le devuelve a **V** un polinomio $h_j(x_i)$ tal que

$$h_j(x_i) = O_{j+1} \cdots O_{n + \frac{n(n-1)}{2}} g(r_1, \dots, r_{i-1}, x_i, \dots, x_n)$$

7.2 Si el grado de $h_j(x_i)$ es mayor que $2m$ entonces **V** rechaza

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7. Los siguientes pasos se repiten para $j = 2, \dots, n + \frac{n(n-1)}{2}$

7.1 **D** le devuelve a **V** un polinomio $h_j(x_i)$ tal que

$$h_j(x_i) = O_{j+1} \cdots O_{n + \frac{n(n-1)}{2}} g(r_1, \dots, r_{i-1}, x_i, \dots, x_n)$$

7.2 Si el grado de $h_j(x_i)$ es mayor que $2m$ entonces **V** rechaza

7.3 **V** verifica que alguna de las siguientes condiciones es cierta, y si no es así entonces rechaza

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7. Los siguientes pasos se repiten para $j = 2, \dots, n + \frac{n(n-1)}{2}$

7.1 **D** le devuelve a **V** un polinomio $h_j(x_i)$ tal que

$$h_j(x_i) = O_{j+1} \cdots O_{n + \frac{n(n-1)}{2}} g(r_1, \dots, r_{i-1}, x_i, \dots, x_n)$$

7.2 Si el grado de $h_j(x_i)$ es mayor que $2m$ entonces **V** rechaza

7.3 **V** verifica que alguna de las siguientes condiciones es cierta, y si no es así entonces rechaza

7.3.1 $O_j = \exists x_i$ y $h_{j-1}(s_{j-1}) = h_j(0) + h_j(1)$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7. Los siguientes pasos se repiten para $j = 2, \dots, n + \frac{n(n-1)}{2}$

7.1 **D** le devuelve a **V** un polinomio $h_j(x_i)$ tal que

$$h_j(x_i) = O_{j+1} \cdots O_{n + \frac{n(n-1)}{2}} g(r_1, \dots, r_{i-1}, x_i, \dots, x_n)$$

7.2 Si el grado de $h_j(x_i)$ es mayor que $2m$ entonces **V** rechaza

7.3 **V** verifica que alguna de las siguientes condiciones es cierta, y si no es así entonces rechaza

7.3.1 $O_j = \exists x_i$ y $h_{j-1}(s_{j-1}) = h_j(0) + h_j(1)$

7.3.2 $O_j = \forall x_i$ y $h_{j-1}(s_{j-1}) = h_j(0) \cdot h_j(1)$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7. Los siguientes pasos se repiten para $j = 2, \dots, n + \frac{n(n-1)}{2}$

7.1 **D** le devuelve a **V** un polinomio $h_j(x_i)$ tal que

$$h_j(x_i) = O_{j+1} \cdots O_{n + \frac{n(n-1)}{2}} g(r_1, \dots, r_{i-1}, x_i, \dots, x_n)$$

7.2 Si el grado de $h_j(x_i)$ es mayor que $2m$ entonces **V** rechaza

7.3 **V** verifica que alguna de las siguientes condiciones es cierta, y si no es así entonces rechaza

7.3.1 $O_j = \exists x_i$ y $h_{j-1}(s_{j-1}) = h_j(0) + h_j(1)$

7.3.2 $O_j = \forall x_i$ y $h_{j-1}(s_{j-1}) = h_j(0) \cdot h_j(1)$

7.3.3 $O_j = Lx_k$ y $h_{j-1}(s_{j-1}) = (1 - s_{j-1}) \cdot h_j(0) + s_{j-1} \cdot h_j(1)$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7.4 **V** genera al azar con distribución uniforme un número entero
 $s_j \in \{0, \dots, 2^{(n^2+n)m} - 1\}$

CNF-QBF está en $IP[n^2 + n]$: el protocolo

- 7.4 **V** genera al azar con distribución uniforme un número entero $s_j \in \{0, \dots, 2^{(n^2+n)m} - 1\}$
- 7.5 Si $O_{j+1} = \exists x_{i+1}$ u $O_{j+1} = \forall x_{i+1}$, entonces **V** define $r_i = s_j$ y se incrementa el contador i en 1

CNF-QBF está en $IP[n^2 + n]$: el protocolo

- 7.4 **V** genera al azar con distribución uniforme un número entero $s_j \in \{0, \dots, 2^{(n^2+n)m} - 1\}$
- 7.5 Si $O_{j+1} = \exists x_{i+1}$ u $O_{j+1} = \forall x_{i+1}$, entonces **V** define $r_i = s_j$ y se incrementa el contador i en 1
- 7.6 Si $j < n + \frac{n(n-1)}{2}$, entonces le envía s_j a **D**

CNF-QBF está en $IP[n^2 + n]$: el protocolo

7.4 **V** genera al azar con distribución uniforme un número entero $s_j \in \{0, \dots, 2^{(n^2+n)m} - 1\}$

7.5 Si $O_{j+1} = \exists x_{i+1}$ u $O_{j+1} = \forall x_{i+1}$, entonces **V** define $r_i = s_j$ y se incrementa el contador i en 1

7.6 Si $j < n + \frac{n(n-1)}{2}$, entonces le envía s_j a **D**

8. **V** verifica si $h_{n + \frac{n(n-1)}{2}}(r_n) = g(r_1, \dots, r_n)$. Si es así entonces acepta, y en caso contrario rechaza

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

El protocolo tiene $n^2 + n$ rondas

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

El protocolo tiene $n^2 + n$ rondas

Si φ es cierta, entonces considerando un demostrador \mathbf{D} que utiliza el polinomio $g(x_1, \dots, x_n)$ obtenemos que:

$$\Pr((\mathbf{V}, \mathbf{D}) \text{ acepte } \varphi) = 1$$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

El protocolo tiene $n^2 + n$ rondas

Si φ es cierta, entonces considerando un demostrador **D** que utiliza el polinomio $g(x_1, \dots, x_n)$ obtenemos que:

$$\Pr((\mathbf{V}, \mathbf{D}) \text{ acepte } \varphi) = 1$$

Suponga que φ no es cierta.

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

El protocolo tiene $n^2 + n$ rondas

Si φ es cierta, entonces considerando un demostrador \mathbf{D} que utiliza el polinomio $g(x_1, \dots, x_n)$ obtenemos que:

$$\Pr((\mathbf{V}, \mathbf{D}) \text{ acepte } \varphi) = 1$$

Suponga que φ no es cierta. Nos falta demostrar que para cualquier demostrador \mathbf{D}' :

$$\Pr((\mathbf{V}, \mathbf{D}') \text{ acepte } \varphi) \leq \frac{1}{4}$$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Suponga que \mathbf{D}' está tratando de engañar a \mathbf{V}

- ▶ \mathbf{D}' está tratando de que \mathbf{V} acepte φ , aunque φ no es cierta

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Suponga que \mathbf{D}' está tratando de engañar a \mathbf{V}

- ▶ \mathbf{D}' está tratando de que \mathbf{V} acepte φ , aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por \mathbf{D}'

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Suponga que \mathbf{D}' está tratando de engañar a \mathbf{V}

▶ \mathbf{D}' está tratando de que \mathbf{V} acepte φ , aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por \mathbf{D}'

Tenemos que $h'_1(x_1) \neq h_1(x_1)$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Suponga que \mathbf{D}' está tratando de engañar a \mathbf{V}

- ▶ \mathbf{D}' está tratando de que \mathbf{V} acepte φ , aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por \mathbf{D}'

Tenemos que $h'_1(x_1) \neq h_1(x_1)$

- ▶ Si $O_1 = \exists x_1$, entonces $h'_1(x_1) \neq h_1(x_1)$ puesto que $h_1(0) + h_1(1) = 0$ y \mathbf{D}' está tratando de demostrar a \mathbf{V} que $h'_1(0) + h'_1(1) = 1$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Suponga que \mathbf{D}' está tratando de engañar a \mathbf{V}

- ▶ \mathbf{D}' está tratando de que \mathbf{V} acepte φ , aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por \mathbf{D}'

Tenemos que $h'_1(x_1) \neq h_1(x_1)$

- ▶ Si $O_1 = \exists x_1$, entonces $h'_1(x_1) \neq h_1(x_1)$ puesto que $h_1(0) + h_1(1) = 0$ y \mathbf{D}' está tratando de demostrar a \mathbf{V} que $h'_1(0) + h'_1(1) = 1$
- ▶ Si $O_1 = \forall x_1$, entonces $h'_1(x_1) \neq h_1(x_1)$ puesto que $h_1(0) \cdot h_1(1) = 0$ y \mathbf{D}' está tratando de demostrar a \mathbf{V} que $h'_1(0) \cdot h'_1(1) = 1$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Si $h'_1(s_1) = h_1(s_1)$, entonces \mathbf{D}' puede definir $h'_2(x_1) = h_2(x_1)$, y desde ahí puede engañar a \mathbf{V}

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Si $h'_1(s_1) = h_1(s_1)$, entonces \mathbf{D}' puede definir $h'_2(x_1) = h_2(x_1)$, y desde ahí puede engañar a \mathbf{V}

► Puesto que

$$(1 - s_1) \cdot h'_2(0) + s_1 \cdot h'_2(1) = (1 - s_1) \cdot h_2(0) + s_1 \cdot h_2(1) = h_1(s_1) = h'_1(s_1)$$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Si $h'_1(s_1) = h_1(s_1)$, entonces \mathbf{D}' puede definir $h'_2(x_1) = h_2(x_1)$, y desde ahí puede engañar a \mathbf{V}

► Puesto que

$$(1 - s_1) \cdot h'_2(0) + s_1 \cdot h'_2(1) = (1 - s_1) \cdot h_2(0) + s_1 \cdot h_2(1) = h_1(s_1) = h'_1(s_1)$$

Pero si $h'_1(s_1) \neq h_1(s_1)$, entonces se debe tener que $h'_2(x_2) \neq h_2(x_2)$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Si $h'_1(s_1) = h_1(s_1)$, entonces \mathbf{D}' puede definir $h'_2(x_1) = h_2(x_1)$, y desde ahí puede engañar a \mathbf{V}

► Puesto que

$$(1 - s_1) \cdot h'_2(0) + s_1 \cdot h'_2(1) = (1 - s_1) \cdot h_2(0) + s_1 \cdot h_2(1) = h_1(s_1) = h'_1(s_1)$$

Pero si $h'_1(s_1) \neq h_1(s_1)$, entonces se debe tener que $h'_2(x_2) \neq h_2(x_2)$

► Puesto que $(1 - s_1) \cdot h_2(0) + s_1 \cdot h_2(1) = h_1(s_1)$ y \mathbf{D}' está tratando de demostrar que $(1 - s_1) \cdot h'_2(0) + s_1 \cdot h'_2(1) = h'_1(s_1)$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Si continuamos con este razonamiento vemos que **D'** logra engañar a **V** si la siguiente condición es cierta:

$$\bigvee_{i=1}^{n + \frac{n(n-1)}{2}} h'_i(s_i) = h_i(s_i)$$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Si continuamos con este razonamiento vemos que \mathbf{D}' logra engañar a \mathbf{V} si la siguiente condición es cierta:

$$\bigvee_{i=1}^{n + \frac{n(n-1)}{2}} h'_i(s_i) = h_i(s_i)$$

En particular, la condición $h'_{n + \frac{n(n-1)}{2}}(r_n) = h_{n + \frac{n(n-1)}{2}}(r_n)$ es equivalente a pedir que $h'_{n + \frac{n(n-1)}{2}}(r_n) = g(r_1, \dots, r_n)$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Si continuamos con este razonamiento vemos que \mathbf{D}' logra engañar a \mathbf{V} si la siguiente condición es cierta:

$$\bigvee_{i=1}^{n + \frac{n(n-1)}{2}} h'_i(s_i) = h_i(s_i)$$

En particular, la condición $h'_{n + \frac{n(n-1)}{2}}(r_n) = h_{n + \frac{n(n-1)}{2}}(r_n)$ es equivalente a pedir que $h'_{n + \frac{n(n-1)}{2}}(r_n) = g(r_1, \dots, r_n)$

► Esta es la última condición que se necesita para que \mathbf{V} acepte

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Por definición del protocolo y dado que ninguna cláusula de φ tiene literales repetidos o complementarios, el grado de cada $h_i(x_i)$ y $h'_i(x'_i)$ es a lo más $2m$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

Por definición del protocolo y dado que ninguna cláusula de φ tiene literales repetidos o complementarios, el grado de cada $h_i(x_i)$ y $h'_i(x'_i)$ es a lo más $2m$

Por lo tanto tenemos que:

$\mathbf{Pr}((\mathbf{V}, \mathbf{D}') \text{ acepta } \varphi)$

$$\begin{aligned} &= \mathbf{Pr}\left(\bigvee_{i=1}^{n + \frac{n(n-1)}{2}} h'_i(r_i) = h_i(r_i)\right) \\ &= \mathbf{Pr}\left(\bigvee_{i=1}^{n + \frac{n(n-1)}{2}} \left[h'_i(r_i) = h_i(r_i) \wedge \bigwedge_{j=1}^{i-1} h'_j(r_j) \neq h_j(r_j) \right]\right) \\ &= \sum_{i=1}^{n + \frac{n(n-1)}{2}} \mathbf{Pr}\left(h'_i(r_i) = h_i(r_i) \wedge \bigwedge_{j=1}^{i-1} h'_j(r_j) \neq h_j(r_j)\right) \\ &\leq \sum_{i=1}^{n + \frac{n(n-1)}{2}} \mathbf{Pr}\left(h'_i(r_i) = h_i(r_i) \mid \bigwedge_{j=1}^{i-1} h'_j(r_j) \neq h_j(r_j)\right) \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

$\Pr((\mathbf{V}, \mathbf{D}') \text{ acepta } \varphi)$

$$\begin{aligned}
 &\leq \sum_{i=1}^{n + \frac{n(n-1)}{2}} \Pr\left(h'_i(r_i) = h_i(r_i) \mid \bigwedge_{j=1}^{i-1} h'_j(r_j) \neq h_j(r_j)\right) \\
 &\leq \sum_{i=1}^{n + \frac{n(n-1)}{2}} \frac{2m}{2^{(n^2+n)m}} \\
 &= \frac{(n + \frac{n(n-1)}{2})2m}{2^{(n^2+n)m}} \\
 &= \frac{(n^2 + n)m}{2^{(n^2+n)m}} \\
 &\leq \frac{1}{4}
 \end{aligned}$$

CNF-QBF está en $IP[n^2 + n]$: la probabilidad de error

$\Pr((\mathbf{V}, \mathbf{D}') \text{ acepta } \varphi)$

$$\begin{aligned}
 &\leq \sum_{i=1}^{n + \frac{n(n-1)}{2}} \Pr\left(h'_i(r_i) = h_i(r_i) \mid \bigwedge_{j=1}^{i-1} h'_j(r_j) \neq h_j(r_j)\right) \\
 &\leq \sum_{i=1}^{n + \frac{n(n-1)}{2}} \frac{2m}{2^{(n^2+n)m}} \\
 &= \frac{(n + \frac{n(n-1)}{2})2m}{2^{(n^2+n)m}} \\
 &= \frac{(n^2 + n)m}{2^{(n^2+n)m}} \\
 &\leq \frac{1}{4}
 \end{aligned}$$

□

Un corolario fundamental

Corolario

$$IP = co-IP$$

Un corolario fundamental

Corolario

$$IP = co-IP$$

Ejercicio

Demuestre el corolario

Un corolario fundamental

Corolario

$$IP = co-IP$$

Ejercicio

Demuestre el corolario

- ▶ Dado un protocolo interactivo para un lenguaje L , ¿cómo construye un protocolo interactivo para \bar{L} ?

¿Es necesario que la aleatoriedad sea privada?

Definimos la clase $AM[k]$ como $IP[k]$ pero con una restricción adicional:

Cada vez que **V** envía una pregunta a **D** tiene que enviarle adicionalmente los bits aleatorios usados

¿Es necesario que la aleatoriedad sea privada?

Definimos la clase $AM[k]$ como $IP[k]$ pero con una restricción adicional:

Cada vez que **V** envía una pregunta a **D** tiene que enviarle adicionalmente los bits aleatorios usados

Hablamos entonces de protocolos interactivos con bits aleatorios públicos

¿Es necesario que la aleatoriedad sea privada?

Definimos la clase $AM[k]$ como $IP[k]$ pero con una restricción adicional:

Cada vez que **V** envía una pregunta a **D** tiene que enviarle adicionalmente los bits aleatorios usados

Hablamos entonces de protocolos interactivos con bits aleatorios públicos

- ▶ Note que **D** conoce los bits aleatorios usados por **V**, no conoce los que **V** podría usar en el futuro

¿Es necesario que la aleatoriedad sea privada?

Definimos la clase $AM[k]$ como $IP[k]$ pero con una restricción adicional:

Cada vez que **V** envía una pregunta a **D** tiene que enviarle adicionalmente los bits aleatorios usados

Hablamos entonces de protocolos interactivos con bits aleatorios públicos

- ▶ Note que **D** conoce los bits aleatorios usados por **V**, no conoce los que **V** podría usar en el futuro

¿Los protocolos interactivos con bit aleatorios públicos son menos poderosos?

¿Es necesario que la aleatoriedad sea privada?

Definimos la clase $AM[k]$ como $IP[k]$ pero con una restricción adicional:

Cada vez que **V** envía una pregunta a **D** tiene que enviarle adicionalmente los bits aleatorios usados

Hablamos entonces de protocolos interactivos con bits aleatorios públicos

- ▶ Note que **D** conoce los bits aleatorios usados por **V**, no conoce los que **V** podría usar en el futuro

¿Los protocolos interactivos con bit aleatorios públicos son menos poderosos?

- ▶ ¿Funciona el protocolo interactivo estudiado para $\overline{\text{GRAPH-ISO}}$ con bit aleatorios públicos?

¿Es necesario que la aleatoriedad sea privada?

Teorema

$$IP = \bigcup_{k \in \mathbb{N}} AM[n^k]$$

¿Es necesario que la aleatoriedad sea privada?

Teorema

$$IP = \bigcup_{k \in \mathbb{N}} AM[n^k]$$

Ejercicio

Demuestre el teorema utilizando la demostración de que $PSPACE \subseteq IP$

¿Es necesario que la aleatoriedad sea privada?

Teorema

$$IP = \bigcup_{k \in \mathbb{N}} AM[n^k]$$

Ejercicio

Demuestre el teorema utilizando la demostración de que $PSPACE \subseteq IP$

Tenemos entonces un protocolo aleatorizado para $\overline{\text{GRAPH-ISOM}}^*$ con bit aleatorios públicos

¿Es necesario que la aleatoriedad sea privada?

Teorema

$$IP = \bigcup_{k \in \mathbb{N}} AM[n^k]$$

Ejercicio

Demuestre el teorema utilizando la demostración de que $PSPACE \subseteq IP$

Tenemos entonces un protocolo aleatorizado para $\overline{\text{GRAPH-ISO}}$ con bit aleatorios públicos

- ▶ ¿Cómo construye este protocolo?

¿Es necesario que la aleatoriedad sea privada?

Teorema

$$IP = \bigcup_{k \in \mathbb{N}} AM[n^k]$$

Ejercicio

Demuestre el teorema utilizando la demostración de que $PSPACE \subseteq IP$

Tenemos entonces un protocolo aleatorizado para $\overline{\text{GRAPH-ISOM}}^*$ con bit aleatorios públicos

- ▶ ¿Cómo construye este protocolo? ¿Tiene un número constante de rondas?

La clase de complejidad AM (Arthur-Merlin)

Para entender la complejidad de GRAPH-ISO necesitamos saber el número exacto de rondas de un protocolo aleatorizado para $\overline{\text{GRAPH-ISO}}$ con bit aleatorios públicos

La clase de complejidad AM (Arthur-Merlin)

Para entender la complejidad de GRAPH-ISO necesitamos saber el número exacto de rondas de un protocolo aleatorizado para $\overline{\text{GRAPH-ISO}}$ con bit aleatorios públicos

La siguiente clase de complejidad juega un papel fundamental en este estudio:

Definición

$$AM = AM[2]$$

La clase de complejidad AM (Arthur-Merlin)

Para entender la complejidad de GRAPH-ISO necesitamos saber el número exacto de rondas de un protocolo aleatorizado para $\overline{\text{GRAPH-ISO}}$ con bit aleatorios públicos

La siguiente clase de complejidad juega un papel fundamental en este estudio:

Definición

$$AM = AM[2]$$

Note que AM no es definida de manera análoga a IP

¿Cuál es el poder de AM?

No sabemos si $AM = IP$

¿Cuál es el poder de AM?

No sabemos si $AM = IP$

- ▶ Se sabe que $AM \subseteq \Pi_2^P$, así que se cree que no es cierto que $AM = IP$

¿Cuál es el poder de AM?

No sabemos si $AM = IP$

- ▶ Se sabe que $AM \subseteq \Pi_2^P$, así que se cree que no es cierto que $AM = IP$

De hecho es un problema abierto si $AM = co-AM$

¿Cuál es el poder de AM?

No sabemos si $AM = IP$

- ▶ Se sabe que $AM \subseteq \Pi_2^P$, así que se cree que no es cierto que $AM = IP$

De hecho es un problema abierto si $AM = co-AM$

- ▶ Pero sabemos que $BPP \subseteq AM \cap co-AM \subseteq \Sigma_2^P \cap \Pi_2^P$

Un resultado fundamental: $\overline{\text{GRAPH-ISO}}$ está en AM

Teorema

$\overline{\text{GRAPH-ISO}} \in \text{AM}$

Un resultado fundamental: $\overline{\text{GRAPH-ISO}}$ está en AM

Teorema

$$\overline{\text{GRAPH-ISO}} \in \text{AM}$$

Este resultado es fundamental para entender la complejidad de GRAPH-ISO

Un resultado fundamental: $\overline{\text{GRAPH-ISO}}$ está en AM

Teorema

$\overline{\text{GRAPH-ISO}} \in \text{AM}$

Este resultado es fundamental para entender la complejidad de GRAPH-ISO

- ▶ Note que de este resultado concluimos que $\text{GRAPH-ISO} \in \text{AM} \cap \text{co-AM}$

Un resultado fundamental: $\overline{\text{GRAPH-ISO}}$ está en AM

Teorema

$\overline{\text{GRAPH-ISO}} \in \text{AM}$

Este resultado es fundamental para entender la complejidad de GRAPH-ISO

- ▶ Note que de este resultado concluimos que $\text{GRAPH-ISO} \in \text{AM} \cap \text{co-AM}$

Para hacer la demostración primero mostramos que AM es la extensión de BPP cuando consideramos no determinismo

Extendiendo la definición de BPP

Recuerde que un lenguaje L sobre un alfabeto Σ está en BPP si existe una MT probabilística M tal que $t_M(n)$ es $O(n^k)$ y para cada $w \in \Sigma^*$:

- ▶ Si $w \in L$, entonces $\Pr_s(M(w, s) \text{ acepta}) \geq \frac{3}{4}$
- ▶ Si $w \notin L$, entonces $\Pr_s(M(w, s) \text{ acepta}) \leq \frac{1}{4}$

Extendiendo la definición de BPP

Recuerde que un lenguaje L sobre un alfabeto Σ está en BPP si existe una MT probabilística M tal que $t_M(n)$ es $O(n^k)$ y para cada $w \in \Sigma^*$:

- ▶ Si $w \in L$, entonces $\Pr_s(M(w, s) \text{ acepta}) \geq \frac{3}{4}$
- ▶ Si $w \notin L$, entonces $\Pr_s(M(w, s) \text{ acepta}) \leq \frac{1}{4}$

Podemos extender la definición permitiendo a M ser no determinista

- ▶ $M(w, s)$ acepta si y sólo si existe una ejecución de M con entrada (w, s) que se detiene en un estado final

La clase de complejidad ND-BPP

Definición

Sea L un lenguaje sobre un alfabeto Σ . Entonces L está en ND-BPP si existe una MT probabilística *no determinista* M tal que $t_M(n)$ es $O(n^k)$ y para cada $w \in \Sigma^*$:

- ▶ Si $w \in L$, entonces $\Pr_s(M(w, s) \text{ acepta}) \geq \frac{3}{4}$
- ▶ Si $w \notin L$, entonces $\Pr_s(M(w, s) \text{ acepta}) \leq \frac{1}{4}$

AM y ND-BPP son la misma clase

Teorema

$$AM = ND-BPP$$

AM y ND-BPP son la misma clase

Teorema

$$AM = ND-BPP$$

Ejercicio

Demuestre el teorema

AM y ND-BPP son la misma clase

Teorema

$$AM = ND-BPP$$

Ejercicio

Demuestre el teorema

Vamos a utilizar la caracterización de AM dada por ND-BPP en la demostración de que $\overline{\text{GRAPH-ISO}} \in AM$

Un poco de notación para grafos

Sin pérdida de generalidad, suponemos desde ahora en adelante que si un grafo $G = (N, A)$ tiene n nodos, entonces $N = \{1, \dots, n\}$

- ▶ Tenemos entonces 2^{n^2} grafos con n nodos

Un poco de notación para grafos

Sin pérdida de generalidad, suponemos desde ahora en adelante que si un grafo $G = (N, A)$ tiene n nodos, entonces $N = \{1, \dots, n\}$

► Tenemos entonces 2^{n^2} grafos con n nodos

Notación

Dado un grafo $G = (N, A)$ y una biyección $f : N \rightarrow N$, definimos $f(G)$ como un grafo (N, A') tal que para cada $(a, b) \in N \times N$:

$$(a, b) \in A \text{ si y sólo si } (f(a), f(b)) \in A'$$

Un poco de notación para grafos

Sin pérdida de generalidad, suponemos desde ahora en adelante que si un grafo $G = (N, A)$ tiene n nodos, entonces $N = \{1, \dots, n\}$

- ▶ Tenemos entonces 2^{n^2} grafos con n nodos

Notación

Dado un grafo $G = (N, A)$ y una biyección $f : N \rightarrow N$, definimos $f(G)$ como un grafo (N, A') tal que para cada $(a, b) \in N \times N$:

$$(a, b) \in A \text{ si y sólo si } (f(a), f(b)) \in A'$$

Note que G y $f(G)$ son grafos isomorfos en la definición anterior.

- ▶ De hecho f es un isomorfismo de G en $f(G)$

Los automorfismos de un grafo

Definición

Dado un grafo $G = (N, A)$ y una biyección $f : N \rightarrow N$, decimos que f es un automorfismo para G si $f(G) = G$

El conjunto de los automorfismos de un grafo G es denotado como $\text{Aut}(G)$

- ▶ Note que si G tiene n nodos, entonces $|\text{Aut}(G)| \leq n!$

Los automorfismos de un grafo

Definición

Dado un grafo $G = (N, A)$ y una biyección $f : N \rightarrow N$, decimos que f es un automorfismo para G si $f(G) = G$

El conjunto de los automorfismos de un grafo G es denotado como $\text{Aut}(G)$

► Note que si G tiene n nodos, entonces $|\text{Aut}(G)| \leq n!$

Ejercicio

Sea n un número natural arbitrario.

1. Construya un grafo G_1 con n nodos tal que $|\text{Aut}(G_1)| = n!$
2. Construya un grafo G_2 con n nodos tal que $|\text{Aut}(G_2)| = 1$

Contando el número de grafos isomorfos a un grafo

Considere el siguiente grafo $G = (N, A)$:



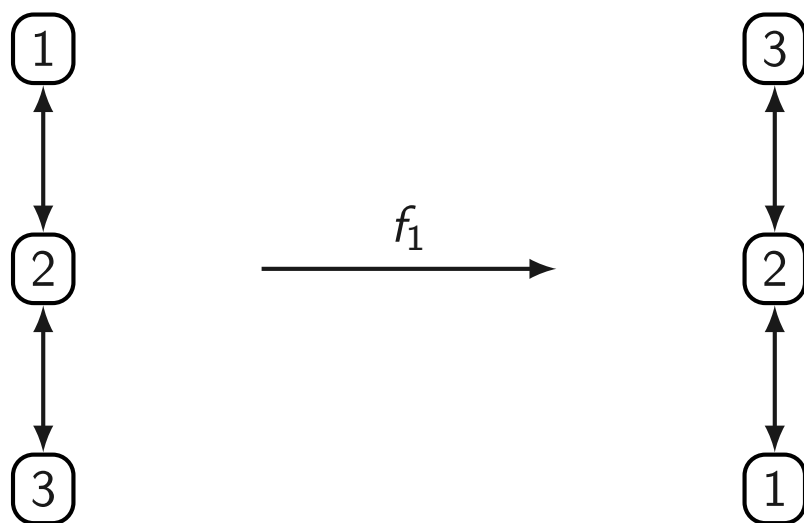
En este caso tenemos que $N = \{1, 2, 3\}$ y $A = \{(1, 2), (2, 1), (2, 3), (3, 2)\}$

Contando el número de grafos isomorfos a un grafo

Considere la biyección $f_1(1) = 3$, $f_1(2) = 2$ y $f_1(3) = 1$:

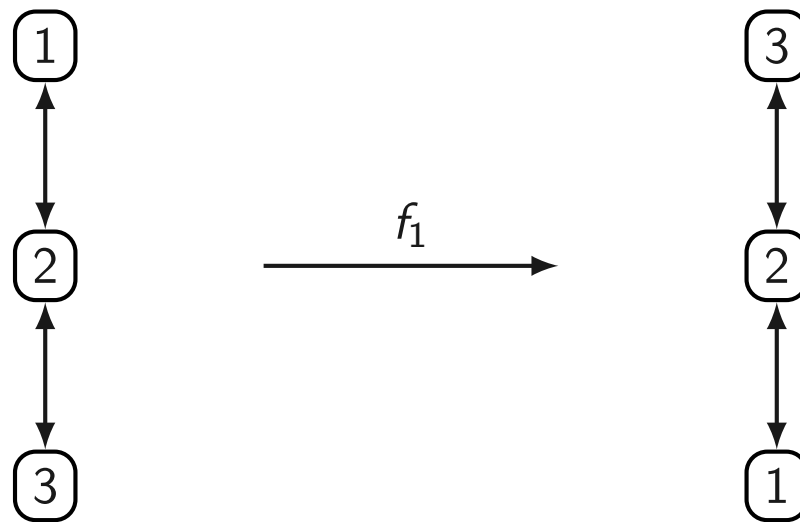
Contando el número de grafos isomorfos a un grafo

Considere la biyección $f_1(1) = 3$, $f_1(2) = 2$ y $f_1(3) = 1$:



Contando el número de grafos isomorfos a un grafo

Considere la biyección $f_1(1) = 3$, $f_1(2) = 2$ y $f_1(3) = 1$:



f_1 es un automorfismo para G ya que $f_1(G) = G$

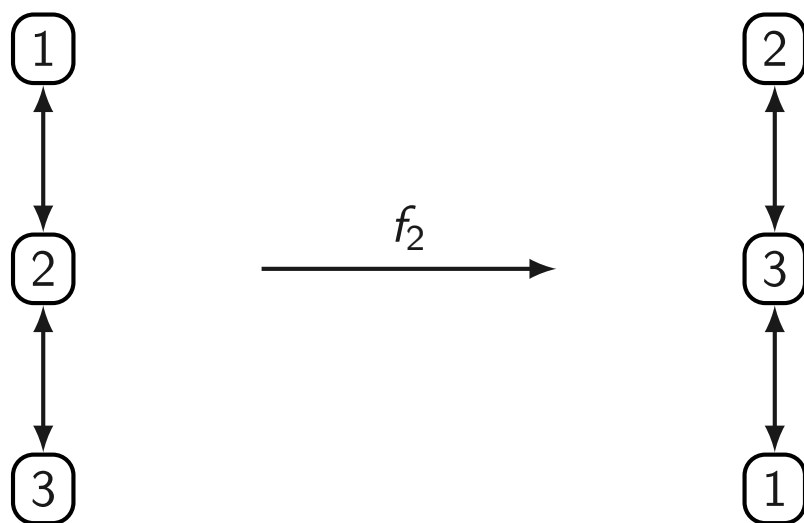
► En particular, si $f_1(G) = (N, A')$ entonces $A = A'$

Contando el número de grafos isomorfos a un grafo

Considere ahora la biyección $f_2(1) = 2$, $f_2(2) = 3$ y $f_2(3) = 1$:

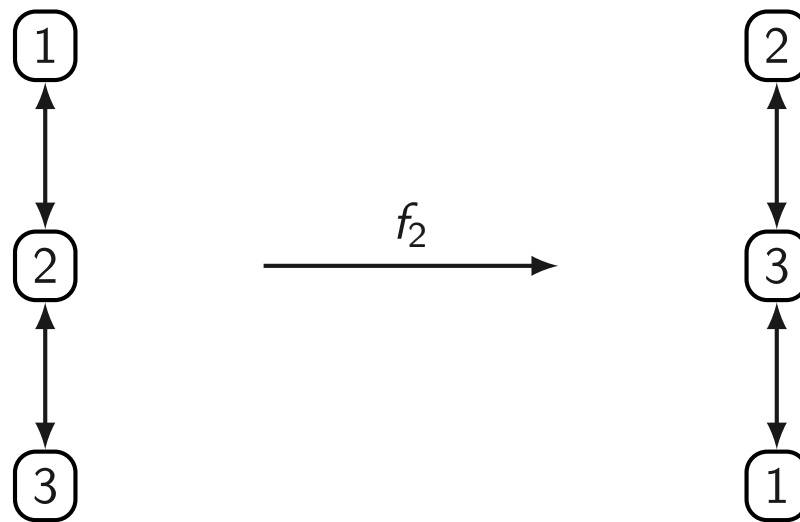
Contando el número de grafos isomorfos a un grafo

Considere ahora la biyección $f_2(1) = 2$, $f_2(2) = 3$ y $f_2(3) = 1$:



Contando el número de grafos isomorfos a un grafo

Considere ahora la biyección $f_2(1) = 2$, $f_2(2) = 3$ y $f_2(3) = 1$:

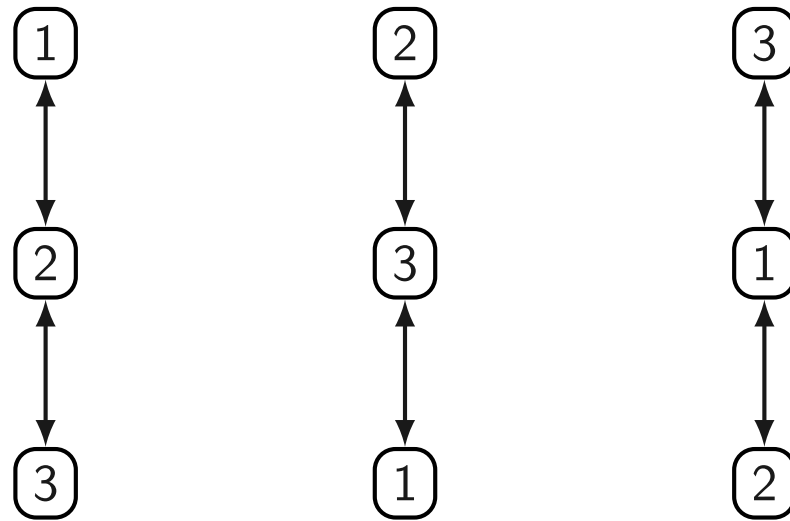


f_2 no es un automorfismo para G ya que $f_2(G) \neq G$

- ▶ En particular, el arco $(1, 2)$ está G pero no en $f_2(G)$

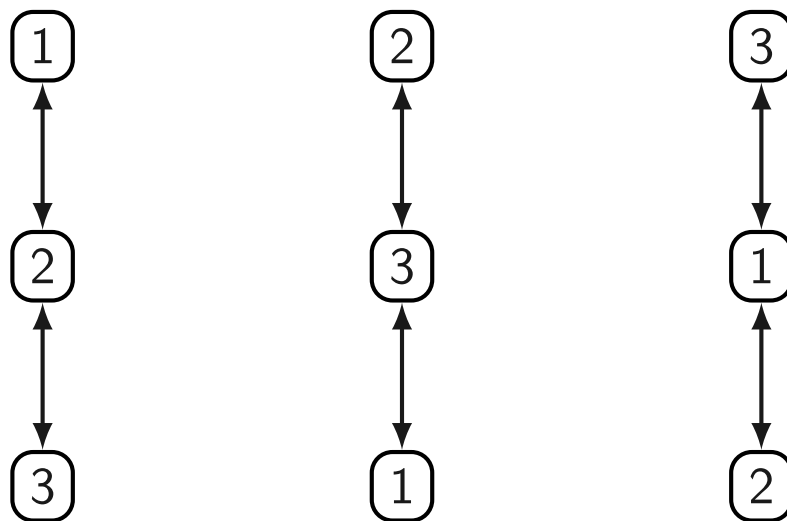
Contando el número de grafos isomorfos a un grafo

Para el caso de G tenemos seis biyecciones posibles que generan tres grafos distintos:



Contando el número de grafos isomorfos a un grafo

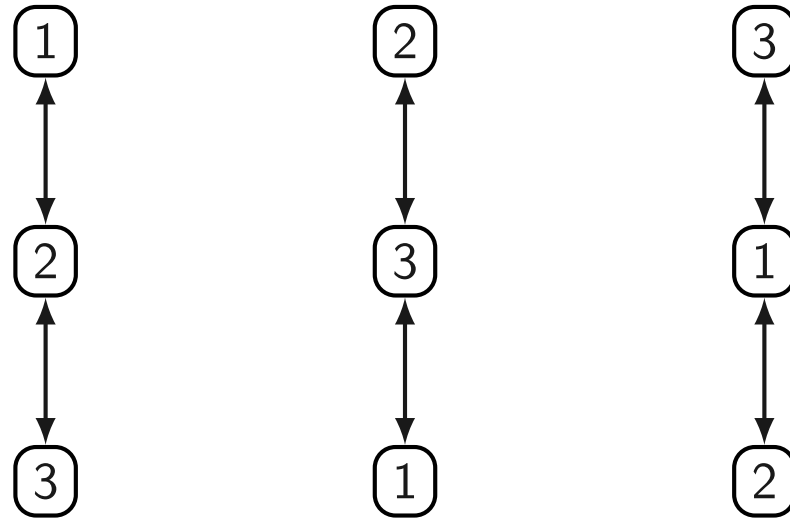
Para el caso de G tenemos seis biyecciones posibles que generan tres grafos distintos:



Tenemos entonces tres grafos distintos que son isomorfos a G

Contando el número de grafos isomorfos a un grafo

Para el caso de G tenemos seis biyecciones posibles que generan tres grafos distintos:



Tenemos entonces tres grafos distintos que son isomorfos a G

- ▶ Esto corresponde al número de biyecciones de tres elementos dividido por el número de automorfismo de G . ¿Tiene sentido esta interpretación? ¿Puede ser generalizada?

El número de grafos isomorfos a un grafo

Recuerde que estamos suponiendo que si un grafo tiene n nodos, entonces sus nodos son $1, \dots, n$

El número de grafos isomorfos a un grafo

Recuerde que estamos suponiendo que si un grafo tiene n nodos, entonces sus nodos son $1, \dots, n$

Lema

Sea G es un grafo con n nodos. El número de grafos isomorfos a G es:

$$\frac{n!}{|Aut(G)|}$$

El número de grafos isomorfos a un grafo

Recuerde que estamos suponiendo que si un grafo tiene n nodos, entonces sus nodos son $1, \dots, n$

Lema

Sea G es un grafo con n nodos. El número de grafos isomorfos a G es:

$$\frac{n!}{|Aut(G)|}$$

Demostración: Sea $B = \{f : \{1, \dots, n\} \rightarrow \{1, \dots, n\} \mid f \text{ es una biyección}\}$

Defina \sim como la siguiente relación sobre B . Para cada $f_1, f_2 \in B$:

$$f_1 \sim f_2 \quad \text{si y sólo si} \quad f_1(G) = f_2(G)$$

Demostración del lema

\sim es una relación de equivalencia sobre B

▶ ¿Por qué?

Demostración del lema

\sim es una relación de equivalencia sobre B

▶ ¿Por qué?

Sea $[f]_{\sim}$ la clase de equivalencia de $f \in B$

Demostración del lema

\sim es una relación de equivalencia sobre B

▶ ¿Por qué?

Sea $[f]_{\sim}$ la clase de equivalencia de $f \in B$

El número de clases de equivalencia de \sim corresponde al número de grafos isomorfos a G

▶ ¿Por qué?

Demostración del lema

Vamos a demostrar las siguientes propiedades:

1. Si id es la función identidad sobre $\{1, \dots, n\}$: $[\text{id}]_{\sim} = \text{Aut}(G)$
2. Para cada $f_1, f_2 \in B$: $|[f_1]_{\sim}| = |[f_2]_{\sim}|$

Demostración del lema

Vamos a demostrar las siguientes propiedades:

1. Si id es la función identidad sobre $\{1, \dots, n\}$: $[\text{id}]_{\sim} = \text{Aut}(G)$
2. Para cada $f_1, f_2 \in B$: $|[f_1]_{\sim}| = |[f_2]_{\sim}|$

De esto concluimos que el número de clases de equivalencia de \sim es $\frac{n!}{|\text{Aut}(G)|}$, que es lo que teníamos que demostrar.

▶ ¿Por qué?

Demostración del lema

En primer lugar tenemos que:

$$\begin{aligned} [\text{id}]_{\sim} &= \{f \in B \mid \text{id} \sim f\} \\ &= \{f \in B \mid \text{id}(G) = f(G)\} \\ &= \{f \in B \mid G = f(G)\} \\ &= \text{Aut}(G) \end{aligned}$$

Demostración del lema

Sean $f_1, f_2 \in B$

Demostración del lema

Sean $f_1, f_2 \in B$

En segundo lugar tenemos que demostrar que $|[f_1]_{\sim}| = |[f_2]_{\sim}|$

Demostración del lema

Sean $f_1, f_2 \in B$

En segundo lugar tenemos que demostrar que $|[f_1]_{\sim}| = |[f_2]_{\sim}|$

Para hacer esto vamos a construir una biyección $\mathcal{T} : [f_1]_{\sim} \rightarrow [f_2]_{\sim}$

Demostración del lema

Sean $f_1, f_2 \in B$

En segundo lugar tenemos que demostrar que $|[f_1]_{\sim}| = |[f_2]_{\sim}|$

Para hacer esto vamos a construir una biyección $\mathcal{T} : [f_1]_{\sim} \rightarrow [f_2]_{\sim}$

Para cada $f \in [f_1]_{\sim}$, se define $\mathcal{T}(f)$ de la siguiente forma:

$$\mathcal{T}(f) = (f_2 \circ f_1^{-1} \circ f)$$

Demostración del lema

Primero tenemos que demostrar que \mathcal{T} está bien definida.

- ▶ Vale decir, si $f \in [f_1]_{\sim}$, entonces $\mathcal{T}(f) \in [f_2]_{\sim}$

Demostración del lema

Primero tenemos que demostrar que \mathcal{T} está bien definida.

► Vale decir, si $f \in [f_1]_{\sim}$, entonces $\mathcal{T}(f) \in [f_2]_{\sim}$

Si $f \in [f_1]_{\sim}$ tenemos que $f(G) = f_1(G)$. De esto concluimos que:

$$\begin{aligned} f_2(f_1^{-1}(f(G))) &= f_2(f_1^{-1}(f_1(G))) \\ &= f_2(G) \end{aligned}$$

Demostración del lema

Primero tenemos que demostrar que \mathcal{T} está bien definida.

► Vale decir, si $f \in [f_1]_{\sim}$, entonces $\mathcal{T}(f) \in [f_2]_{\sim}$

Si $f \in [f_1]_{\sim}$ tenemos que $f(G) = f_1(G)$. De esto concluimos que:

$$\begin{aligned} f_2(f_1^{-1}(f(G))) &= f_2(f_1^{-1}(f_1(G))) \\ &= f_2(G) \end{aligned}$$

Tenemos entonces que $\mathcal{T}(f)(G) = f_2(G)$

► Vale decir $f_2 \sim \mathcal{T}(f)$, de lo que concluimos que $\mathcal{T}(f) \in [f_2]_{\sim}$

Demostración del lema

Vamos a demostrar ahora que \mathcal{T} es una función 1-1

Demostración del lema

Vamos a demostrar ahora que \mathcal{T} es una función 1-1

Utilizando la asociatividad de la composición de funciones obtenemos:

$$\begin{aligned}\mathcal{T}(f) = \mathcal{T}(g) &\Rightarrow (f_2 \circ f_1^{-1} \circ f) = (f_2 \circ f_1^{-1} \circ g) \\ &\Rightarrow (f_1 \circ f_2^{-1}) \circ (f_2 \circ f_1^{-1} \circ f) = (f_1 \circ f_2^{-1}) \circ (f_2 \circ f_1^{-1} \circ g) \\ &\Rightarrow (f_1 \circ (f_2^{-1} \circ f_2) \circ f_1^{-1} \circ f) = (f_1 \circ (f_2^{-1} \circ f_2) \circ f_1^{-1} \circ g) \\ &\Rightarrow (f_1 \circ \text{id} \circ f_1^{-1} \circ f) = (f_1 \circ \text{id} \circ f_1^{-1} \circ g) \\ &\Rightarrow ((f_1 \circ f_1^{-1}) \circ f) = ((f_1 \circ f_1^{-1}) \circ g) \\ &\Rightarrow (\text{id} \circ f) = (\text{id} \circ g) \\ &\Rightarrow f = g\end{aligned}$$

Demostración del lema

Finalmente vamos a demostrar que \mathcal{T} es sobre.

Demostración del lema

Finalmente vamos a demostrar que \mathcal{T} es sobre.

Sea $g \in [f_2]_{\sim}$ y defina f como $(f_1 \circ f_2^{-1} \circ g)$

Demostración del lema

Finalmente vamos a demostrar que \mathcal{T} es sobre.

Sea $g \in [f_2]_{\sim}$ y defina f como $(f_1 \circ f_2^{-1} \circ g)$

Tenemos que $f \in [f_1]_{\sim}$ ya que:

$$\begin{aligned} f(G) &= (f_1 \circ f_2^{-1} \circ g)(G) \\ &= f_1(f_2^{-1}(g(G))) \\ &= f_1(f_2^{-1}(f_2(G))) \\ &= f_1(G) \end{aligned}$$

Demostración del lema

Además, tenemos que:

$$\begin{aligned}\mathcal{T}(f) &= (f_2 \circ f_1^{-1} \circ f) \\ &= (f_2 \circ f_1^{-1} \circ (f_1 \circ f_2^{-1} \circ g)) \\ &= (f_2 \circ (f_1^{-1} \circ f_1) \circ f_2^{-1} \circ g) \\ &= (f_2 \circ \text{id} \circ f_2^{-1} \circ g) \\ &= ((f_2 \circ f_2^{-1}) \circ g) \\ &= (\text{id} \circ g) \\ &= g\end{aligned}$$

Demostración del lema

Además, tenemos que:

$$\begin{aligned}\mathcal{T}(f) &= (f_2 \circ f_1^{-1} \circ f) \\ &= (f_2 \circ f_1^{-1} \circ (f_1 \circ f_2^{-1} \circ g)) \\ &= (f_2 \circ (f_1^{-1} \circ f_1) \circ f_2^{-1} \circ g) \\ &= (f_2 \circ \text{id} \circ f_2^{-1} \circ g) \\ &= ((f_2 \circ f_2^{-1}) \circ g) \\ &= (\text{id} \circ g) \\ &= g\end{aligned}$$

Concluimos entonces que $\mathcal{T}(f) = g$



Definiendo un testigo (probabilístico) para grafos no isomorfos

Dado un par de grafos (G_1, G_2) , queremos definir un conjunto $\text{num}(G_1, G_2)$ con las siguientes propiedades:

1. Cada elemento de $\text{num}(G_1, G_2)$ es de tamaño polinomial en el tamaño de (G_1, G_2)
2. Cada elemento de $\text{num}(G_1, G_2)$ tiene un testigo de tamaño polinomial de su pertenencia al conjunto
3. Para grafos con n nodos, la cantidad de elementos de $\text{num}(G_1, G_2)$ es necesariamente mayor si G_1 y G_2 no son isomorfos.

Definiendo un testigo (probabilístico) para grafos no isomorfos

Ejemplo

Podríamos intentar definir $\text{num}(G_1, G_2)$ de la siguiente forma:

$$\text{num}(G_1, G_2) = \{f \mid f \text{ es un isomorfismo de } G_1 \text{ a } G_2\}$$

Definiendo un testigo (probabilístico) para grafos no isomorfos

Ejemplo

Podríamos intentar definir $\text{num}(G_1, G_2)$ de la siguiente forma:

$$\text{num}(G_1, G_2) = \{f \mid f \text{ es un isomorfismo de } G_1 \text{ a } G_2\}$$

Esta función satisface 1 y 2, pero no 3

Definiendo un testigo (probabilístico) para grafos no isomorfos

Vamos a considerar la siguiente definición del conjunto $\text{num}(G_1, G_2)$:

$$\text{num}(G_1, G_2) = \{(H, i, f) \mid H \text{ es un grafo isomorfo a } G_1 \text{ o } G_2, \\ i \in \{1, 2\} \text{ y } f \in \text{Aut}(G_i)\}$$

Definiendo un testigo (probabilístico) para grafos no isomorfos

Vamos a considerar la siguiente definición del conjunto $\text{num}(G_1, G_2)$:

$$\text{num}(G_1, G_2) = \{(H, i, f) \mid H \text{ es un grafo isomorfo a } G_1 \text{ o } G_2, \\ i \in \{1, 2\} \text{ y } f \in \text{Aut}(G_i)\}$$

$\text{num}(G_1, G_2)$ satisface las condiciones 1 y 2

- ▶ ¿Cómo se demuestra que satisface la condición 2?

Definiendo un testigo (probabilístico) para grafos no isomorfos

Vamos a considerar la siguiente definición del conjunto $\text{num}(G_1, G_2)$:

$$\text{num}(G_1, G_2) = \{(H, i, f) \mid H \text{ es un grafo isomorfo a } G_1 \text{ o } G_2, \\ i \in \{1, 2\} \text{ y } f \in \text{Aut}(G_i)\}$$

$\text{num}(G_1, G_2)$ satisface las condiciones 1 y 2

▶ ¿Cómo se demuestra que satisface la condición 2?

Vamos a demostrar que $\text{num}(G_1, G_2)$ además satisface la condición 3

El conjunto $\text{num}(G_1, G_2)$ nos ayuda a distinguir

Lema

Sean G_1 y G_2 dos grafos con n nodos cada uno. Si G_1 es isomorfo a G_2 , entonces se tiene que $|\text{num}(G_1, G_2)| = 2 \cdot n!$, si no se tiene que $|\text{num}(G_1, G_2)| \geq 4 \cdot n!$

El conjunto $\text{num}(G_1, G_2)$ nos ayuda a distinguir

Lema

Sean G_1 y G_2 dos grafos con n nodos cada uno. Si G_1 es isomorfo a G_2 , entonces se tiene que $|\text{num}(G_1, G_2)| = 2 \cdot n!$, si no se tiene que $|\text{num}(G_1, G_2)| \geq 4 \cdot n!$

¿Por qué en el lema sólo consideramos grafos con el mismo número de nodos?

- ▶ ¿Cómo manejamos el caso en el que los grafos tienen distinto número de nodos?

Demostración del lema

Primero suponemos que G_1 y G_2 son grafos isomorfos

- ▶ Recuerde que el número de grafos isomorfos a un grafo G con n nodos es $\frac{n!}{|\text{Aut}(G)|}$

Demostración del lema

Primero suponemos que G_1 y G_2 son grafos isomorfos

- ▶ Recuerde que el número de grafos isomorfos a un grafo G con n nodos es $\frac{n!}{|\text{Aut}(G)|}$

Tenemos que:

$$\begin{aligned} |\text{num}(G_1, G_2)| &= |\{(H, i, f) \mid H \text{ es un grafo isomorfo a } G_1 \text{ o } G_2, \\ &\quad i \in \{1, 2\} \text{ y } f \in \text{Aut}(G_i)\}| \\ &= |\{H \mid H \text{ es un grafo isomorfo a } G_1 \text{ o } G_2\}| \cdot \\ &\quad (|\text{Aut}(G_1)| + |\text{Aut}(G_2)|) \\ &= |\{H \mid H \text{ es un grafo isomorfo a } G_1\}| \cdot 2|\text{Aut}(G_1)| \\ &= \frac{n!}{|\text{Aut}(G_1)|} \cdot 2|\text{Aut}(G_1)| \\ &= 2 \cdot n! \end{aligned}$$

Demostración del lema

Suponemos ahora que G_1 y G_2 no son grafos isomorfos

Demostración del lema

Suponemos ahora que G_1 y G_2 no son grafos isomorfos

Tenemos que:

$$\begin{aligned} |\text{num}(G_1, G_2)| &= |\{(H, i, f) \mid H \text{ es un grafo isomorfo a } G_1 \text{ o } G_2, \\ &\quad i \in \{1, 2\} \text{ y } f \in \text{Aut}(G_i)\}| \\ &= (|\{H_1 \mid H_1 \text{ es un grafo isomorfo a } G_1\}| + \\ &\quad |\{H_2 \mid H_2 \text{ es un grafo isomorfo a } G_2\}|) \cdot \\ &\quad (|\text{Aut}(G_1)| + |\text{Aut}(G_2)|) \\ &= \left(\frac{n!}{|\text{Aut}(G_1)|} + \frac{n!}{|\text{Aut}(G_2)|} \right) \cdot (|\text{Aut}(G_1)| + |\text{Aut}(G_2)|) \\ &= n! \frac{(|\text{Aut}(G_1)| + |\text{Aut}(G_2)|)^2}{|\text{Aut}(G_1)| \cdot |\text{Aut}(G_2)|} \end{aligned}$$

Demostración del lema

Para terminar la demostración usamos la siguiente observación:

Observación

Para cada $a, b \in \mathbb{R}$ se tiene que $(a + b)^2 \geq 4ab$, puesto que:

$$\begin{aligned}(a - b)^2 \geq 0 &\Rightarrow a^2 - 2ab + b^2 \geq 0 \\&\Rightarrow a^2 + b^2 \geq 2ab \\&\Rightarrow a^2 + 2ab + b^2 \geq 4ab \\&\Rightarrow (a + b)^2 \geq 4ab\end{aligned}$$

Demostración del lema

Para terminar la demostración usamos la siguiente observación:

Observación

Para cada $a, b \in \mathbb{R}$ se tiene que $(a + b)^2 \geq 4ab$, puesto que:

$$\begin{aligned}(a - b)^2 \geq 0 &\Rightarrow a^2 - 2ab + b^2 \geq 0 \\ &\Rightarrow a^2 + b^2 \geq 2ab \\ &\Rightarrow a^2 + 2ab + b^2 \geq 4ab \\ &\Rightarrow (a + b)^2 \geq 4ab\end{aligned}$$

Concluimos que $\frac{(|\text{Aut}(G_1)| + |\text{Aut}(G_2)|)^2}{|\text{Aut}(G_1)| \cdot |\text{Aut}(G_2)|} \geq 4$, de lo que obtenemos que $|\text{num}(G_1, G_2)| \geq 4 \cdot n!$

