Un protocolo interactivo para conteo

Defina el siguiente lenguaje:

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

Un protocolo interactivo para conteo

Defina el siguiente lenguaje:

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

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

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

Un protocolo interactivo para conteo

Defina el siguiente lenguaje:

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

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

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

COUNT-CNF-SAT$^{\leq}$ y #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

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

Un protocolo interactivo para conteo

Teorema

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

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* **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* **D** *tal que*

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

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

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

Un lema de amplificación para IP

Lema

Suponga que $\ell > 0$ *y* $L \in IP$*. Entonces existe un verificador* **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* **D** *tal que*

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

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

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

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)

$IP = PSPACE$

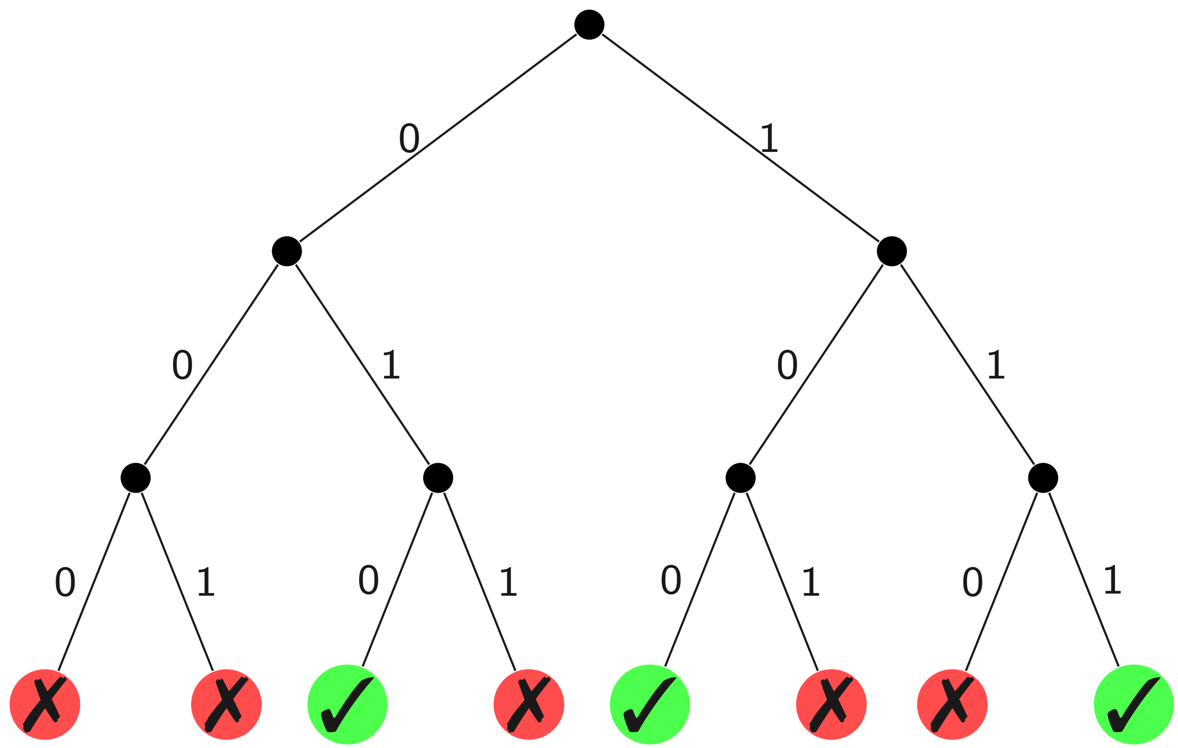
Ejercicio

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

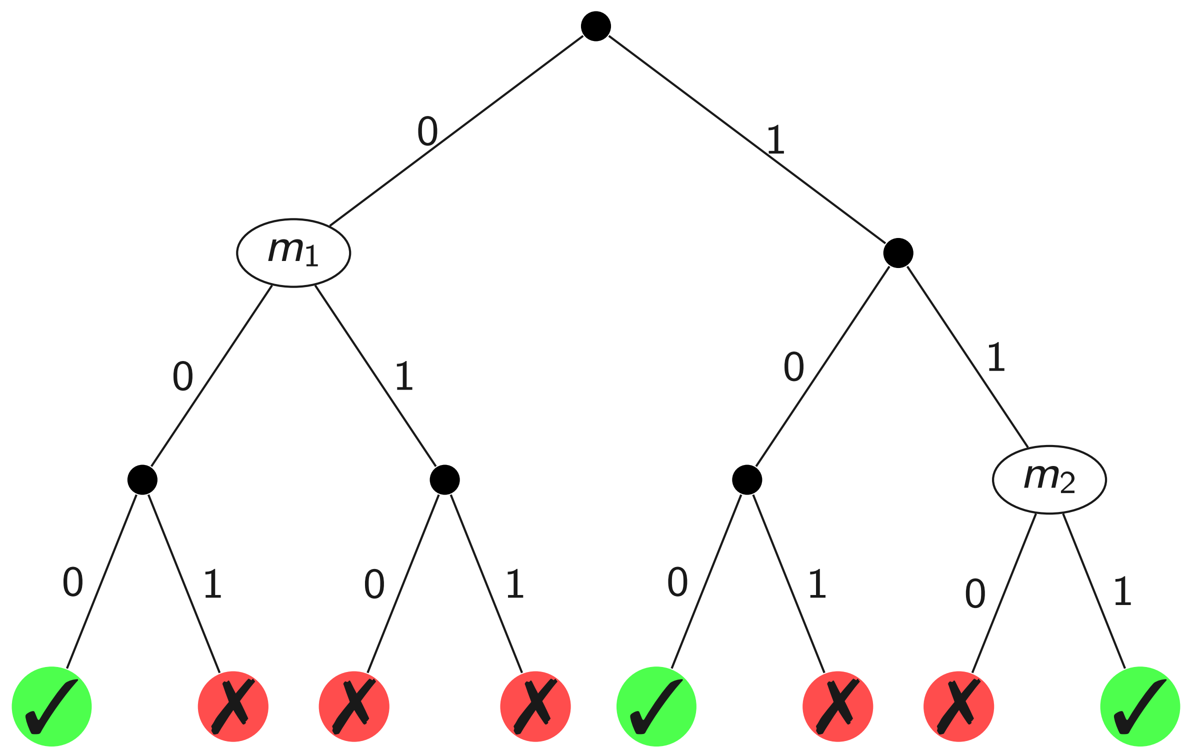
Una solución al ejercicio

Para demostrar que BPP $\subseteq$ PSPACE podemos usar la siguiente representación de la computación de una MT aleatorizada de tiempo polinomial:



Una solución al ejercicio

Para demostrar que IP $\subseteq$ PSPACE usamos una representación similar, pero distinguimos los nodos donde se usa un bit aleatorio de los nodos donde **D** entrega una respuesta:



Una solución al ejercicio

Suponemos que las respuestas de **D** consisten de un bit

Una solución al ejercicio

Suponemos que las respuestas de **D** consisten de un bit

- ¿Por qué podemos suponer esto?

Una solución al ejercicio

Suponemos que las respuestas de **D** consisten de un bit

- ¿Por qué podemos suponer esto?

El paso fundamental: queremos calcular la estrategia de **D** que maximiza la probabilidad de aceptar

Una solución al ejercicio

Suponemos que las respuestas de **D** consisten de un bit

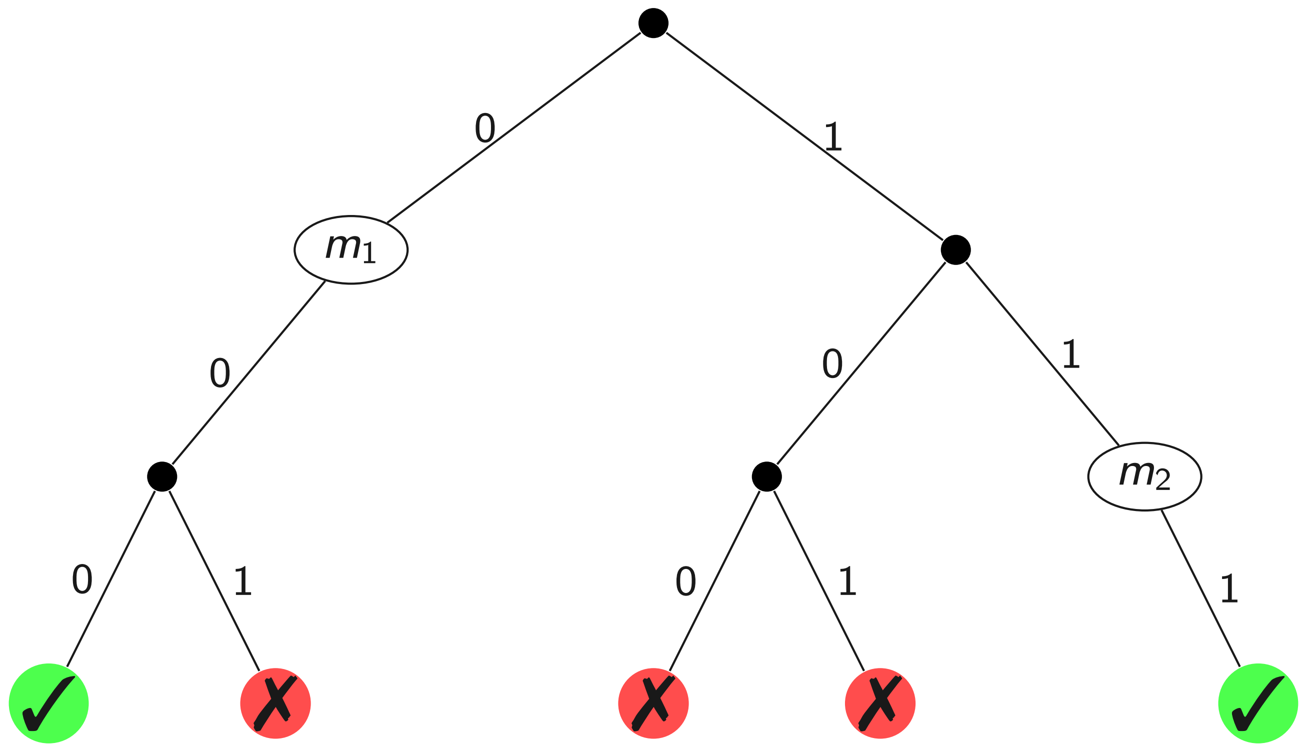
- ¿Por qué podemos suponer esto?

El paso fundamental: queremos calcular la estrategia de **D** que maximiza la probabilidad de aceptar

- ¿Cómo hacemos esto?

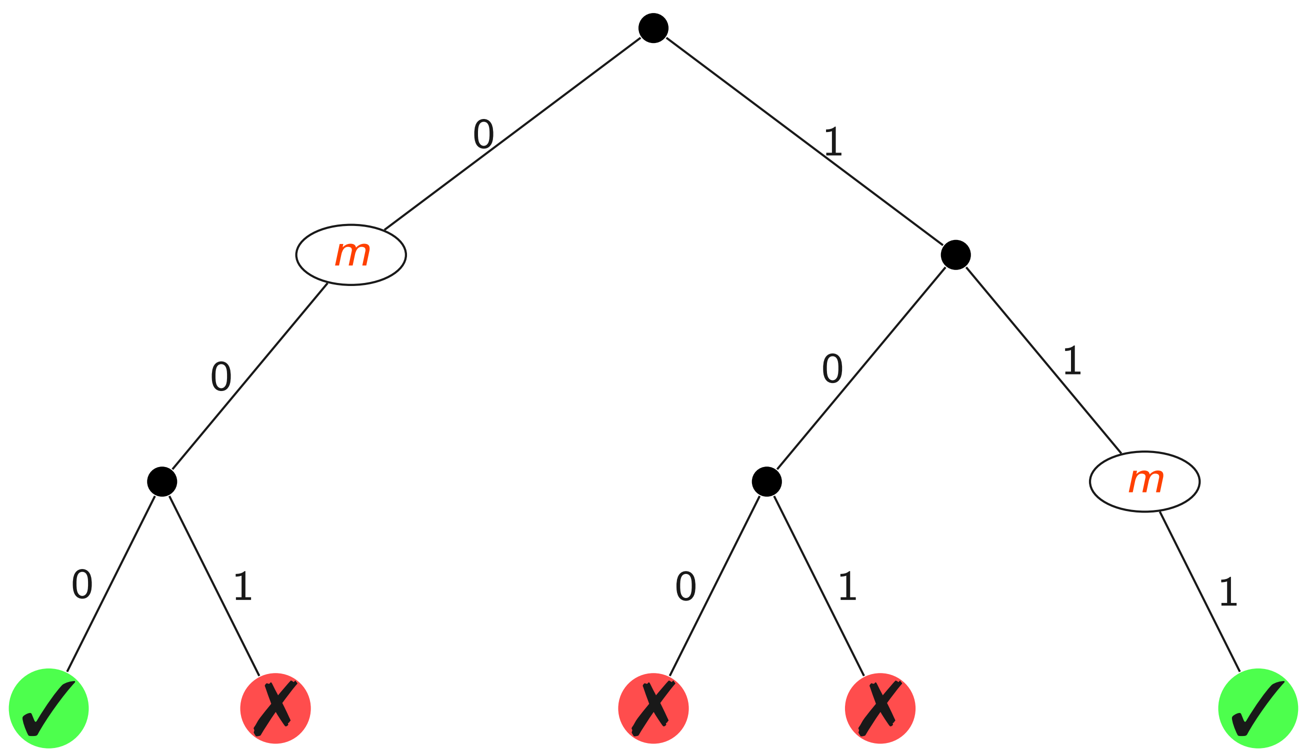
Una solución al ejercicio

La estrategia que maximiza la probabilidad de aceptar:



Una solución al ejercicio

Pero podemos tener un problema de consistencia:



Una solución al ejercicio

Condición de consistencia: **D** debe responder lo mismo si recibe la misma pregunta

Una solución al ejercicio

Condición de consistencia: **D** debe responder lo mismo si recibe la misma pregunta

Para terminar nos falta imponer la condición de consistencia en la ejecución de la MT de espacio polinomial que calcula la estrategia óptima de **D**

Una solución al ejercicio

Condición de consistencia: **D** debe responder lo mismo si recibe la misma pregunta

Para terminar nos falta imponer la condición de consistencia en la ejecución de la MT de espacio polinomial que calcula la estrategia óptima de **D**

Almacenar esta condición para verificar que la decisión sobre un nodo es consistente toma espacio exponencial

Una solución al ejercicio

Condición de consistencia: **D** debe responder lo mismo si recibe la misma pregunta

Para terminar nos falta imponer la condición de consistencia en la ejecución de la MT de espacio polinomial que calcula la estrategia óptima de **D**

Almacenar esta condición para verificar que la decisión sobre un nodo es consistente toma espacio exponencial

- Veamos en la pizarra cómo se soluciona este problema

PSPACE $\subseteq$ IP: Dos problemas PSPACE-completos

Recuerde que una formula proposicional cuantificada es de la forma:

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

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

PSPACE $\subseteq$ IP: Dos problemas PSPACE-completos

Recuerde que una formula proposicional cuantificada es de la forma:

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

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

Por ejemplo, las siguientes son fórmulas proposicionales cuantificadas:

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

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

PSPACE $\subseteq$ IP: Dos problemas PSPACE-completos

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

PSPACE $\subseteq$ IP: 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?

PSPACE $\subseteq$ IP: 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

PSPACE $\subseteq$ IP: 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?

PSPACE $\subseteq$ IP: 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

PSPACE $\subseteq$ IP: 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

PSPACE $\subseteq$ IP: 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

PSPACE $\subseteq$ IP: Dos problemas PSPACE-completos

Teorema

QBF es PSPACE-completo

PSPACE $\subseteq$ IP: Dos problemas PSPACE-completos

Teorema

QBF es PSPACE-completo

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

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

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

PSPACE $\subseteq$ IP: Dos problemas PSPACE-completos

Teorema

QBF es PSPACE-completo

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

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

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

Teorema

CNF-QBF es PSPACE-completo

PSPACE $\subseteq$ IP

Teorema

CNF-QBF está en $IP[n^2 + n]$

PSPACE $\subseteq$ IP

Teorema

CNF-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:

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

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

CNF-QBF está en IP$[n^2 + n]$

Sea φ la siguiente fórmula en CNF-QBF:

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

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

Suponemos que:

- Cada cláusula en $\psi(x_1, \ldots, x_n)$ 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:

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

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

Suponemos que:

- Cada cláusula en $\psi(x_1, \ldots, x_n)$ 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

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

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

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

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

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

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

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

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

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

- Y defina

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

CNF-QBF está en IP$[n^2 + n]$

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

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

CNF-QBF está en IP$[n^2 + n]$

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

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

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

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

CNF-QBF está en IP$[n^2 + n]$

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

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

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

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

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 $\sigma : \{x_1, \ldots, x_n\} \to \{0, 1\}$, tenemos que:

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

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

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

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:

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

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:

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

Es posible demostrar que:

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

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:

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

Es posible demostrar que:

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

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:

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

Es posible demostrar que:

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

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:

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

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:

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

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

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

CNF-QBF está en IP$[n^2 + n]$

Tenemos entonces que la expresión

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

es igual a 0 o 1

CNF-QBF está en IP$[n^2 + n]$

Tenemos entonces que la expresión

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

es igual a 0 o 1

El valor 0 significa que la fórmula $Q_1 x_1 \cdots Q_n x_n \, \psi(x_1, \ldots, x_n)$ no es cierta, y el valor 1 que $Q_1 x_1 \cdots Q_n x_n \, \psi(x_1, \ldots, x_n)$ 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 $\mathrm{IP}[n^2 + n]$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

- El grado de $h_1(x_1)$ está acotado por $m \cdot 2^{n-1}$

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

El demostrador no puede enviar un polinomio de grado $m \cdot 2^{n-1}$

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

El demostrador no puede enviar un polinomio de grado $m \cdot 2^{n-1}$

- 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 $m \cdot 2^{n-1}$

- 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 Lx_i se define de la siguiente forma:

$$
\begin{aligned}
Lx_i\, g(x_1, \ldots, x_n) \;=&\\
&(1 - x_i) \cdot g(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) + \\
&\qquad x_i \cdot g(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)
\end{aligned}
$$

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

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

- Note que $h(x_1, \ldots, x_n)$ tiene las mismas variables que $g(x_1, \ldots, x_n)$

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

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

- Note que $h(x_1, \ldots, x_n)$ tiene las mismas variables que $g(x_1, \ldots, x_n)$

Tenemos que:

$$\begin{aligned} h(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) &= g(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \\ h(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n) &= g(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n) \end{aligned}$$

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

De la propiedad anterior concluimos que:

$$Q_1 x_1 \, Q_2 x_2 \cdots Q_n x_n \, \psi(x_1, \ldots, x_n) \text{ es cierta}$$

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

De la propiedad anterior concluimos que:

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

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

De la propiedad anterior concluimos que:

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

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

De la propiedad anterior concluimos que:

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

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

$$Q_1 x_1 \, L x_1 \, Q_2 x_2 \, L x_1 \, L x_2 \cdots Q_n x_n \, g(x_1, \ldots, x_n) \; = \; 1$$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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 $s_1 \in \{0, \ldots, 2^{(n^2+n)m} - 1\}$, y se lo envía a **D**

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

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

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

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

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

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

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

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

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

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

   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 $j = 2, \ldots, n + \frac{n(n-1)}{2}$

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

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

   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

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

   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 $j = 2, \ldots, n + \frac{n(n-1)}{2}$

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

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

    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 $j = 2, \ldots, n + \frac{n(n-1)}{2}$

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

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

    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 $s_j \in \{0, \ldots, 2^{(n^2+n)m} - 1\}$

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

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

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 $s_j \in \{0, \ldots, 2^{(n^2+n)m} - 1\}$

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 $s_j \in \{0, \ldots, 2^{(n^2+n)m} - 1\}$

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** define $r_n = s_{n + \frac{n(n-1)}{2}}$

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

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

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** define $r_n = s_{n+\frac{n(n-1)}{2}}$

9. **V** verifica si $h_{n+\frac{n(n-1)}{2}}(r_n) = g(r_1, \ldots, r_n)$. 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 **D** que utiliza el polinomio $g(x_1, \ldots, x_n)$ obtenemos que:

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

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 $g(x_1, \ldots, x_n)$ obtenemos que:

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

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 **D** que utiliza el polinomio $g(x_1, \ldots, x_n)$ obtenemos que:

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

Suponga que φ no es cierta. Nos falta demostrar que para cualquier demostrador **D'**:

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

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

Suponga que **D'** está tratando de engañar a **V**

- **D'** está tratando de que **V** acepte φ, aunque φ no es cierta

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

Suponga que **D'** está tratando de engañar a **V**

- **D'** está tratando de que **V** acepte φ, aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por **D'**

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

Suponga que **D'** está tratando de engañar a **V**

- **D'** está tratando de que **V** acepte φ, aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por **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 **D'** está tratando de engañar a **V**

- **D'** está tratando de que **V** acepte φ, aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por **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 **D'** está tratando de demostrar a **V** que $h'_1(0) + h'_1(1) = 1$

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

Suponga que **D'** está tratando de engañar a **V**

- **D'** está tratando de que **V** acepte φ, aunque φ no es cierta

Sean $h'_j(x_i)$ los polinomios generados por **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 **D'** está tratando de demostrar a **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 **D'** está tratando de demostrar a **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 **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 **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 **D'** logra engañar a **V** si la siguiente condición es cierta:

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

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 $h'_{n+\frac{n(n-1)}{2}}(r_n) = g(r_1, \ldots, r_n)$

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:

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

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 $h'_{n+\frac{n(n-1)}{2}}(r_n) = g(r_1, \ldots, r_n)$

- Esta es la última condición que se necesita para que **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:

$$
\begin{aligned}
&\mathbf{Pr}\big((\mathbf{V}, \mathbf{D'}) \text{ acepte } \varphi\big)\\
&\quad = \mathbf{Pr}\left(\bigvee_{i=1}^{n+\frac{n(n-1)}{2}} h'_i(s_i) = h_i(s_i) \right)\\
&\quad = \mathbf{Pr}\left(\bigvee_{i=1}^{n+\frac{n(n-1)}{2}} \left[h'_i(s_i) = h_i(s_i) \wedge \bigwedge_{j=1}^{i-1} h'_j(s_j) \neq h_j(s_j) \right] \right)\\
&\quad = \sum_{i=1}^{n+\frac{n(n-1)}{2}} \mathbf{Pr}\left(h'_i(s_i) = h_i(s_i) \wedge \bigwedge_{j=1}^{i-1} h'_j(s_j) \neq h_j(s_j) \right)\\
&\quad \leq \sum_{i=1}^{n+\frac{n(n-1)}{2}} \mathbf{Pr}\left(h'_i(s_i) = h_i(s_i) \;\middle|\; \bigwedge_{j=1}^{i-1} h'_j(s_j) \neq h_j(s_j) \right)
\end{aligned}
$$

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

$$
\begin{aligned}
&\mathbf{Pr}\big((\mathbf{V},\mathbf{D'}) \text{ acepte } \varphi\big) \\
&\quad \leq \sum_{i=1}^{n+\frac{n(n-1)}{2}} \mathbf{Pr}\left(h'_i(s_i) = h_i(s_i) \;\middle|\; \bigwedge_{j=1}^{i-1} h'_j(s_j) \neq h_j(s_j) \right) \\
&\quad \leq \sum_{i=1}^{n+\frac{n(n-1)}{2}} \frac{2m}{2^{(n^2+n)m}} \\
&\quad = \frac{(n+\frac{n(n-1)}{2})2m}{2^{(n^2+n)m}} \\
&\quad = \frac{(n^2+n)m}{2^{(n^2+n)m}} \\
&\quad \leq \frac{1}{4}
\end{aligned}
$$

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

$$
\begin{aligned}
&\mathbf{Pr}\big((\mathbf{V}, \mathbf{D'}) \text{ acepte } \varphi\big) \\
&\qquad \leq \sum_{i=1}^{n+\frac{n(n-1)}{2}} \mathbf{Pr}\left(h'_i(s_i) = h_i(s_i) \;\middle|\; \bigwedge_{j=1}^{i-1} h'_j(s_j) \neq h_j(s_j)\right) \\
&\qquad \leq \sum_{i=1}^{n+\frac{n(n-1)}{2}} \frac{2m}{2^{(n^2+n)m}} \\
&\qquad = \frac{(n + \frac{n(n-1)}{2})2m}{2^{(n^2+n)m}} \\
&\qquad = \frac{(n^2 + n)m}{2^{(n^2+n)m}} \\
&\qquad \leq \frac{1}{4}
\end{aligned}
$$

□

Un corolario fundamental

Corolario

$IP = co\text{-}IP$

Un corolario fundamental

Corolario

$IP = co\text{-}IP$

Ejercicio

Demuestre el corolario

Un corolario fundamental

Corolario

$IP = co\text{-}IP$

Ejercicio

Demuestre el corolario

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

¿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

Suponga que **V** quiere enviar la pregunta u a **D**, y ha usado los bit aleatorios v en la cinta de bit aleatorios (desde el inicio del protocolo hasta el momento de enviar la pregunta)

¿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

Suponga que **V** quiere enviar la pregunta u a **D**, y ha usado los bit aleatorios v en la cinta de bit aleatorios (desde el inicio del protocolo hasta el momento de enviar la pregunta)

- Entonces **V** escribe $\vdash u\#v\mathtt{BB}\cdots$ en la cinta de comunicación

¿Es necesario que la aleatoriedad sea privada?

Hablamos entonces de protocolos interactivos con bits aleatorios públicos

¿Es necesario que la aleatoriedad sea privada?

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?

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?

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

Vamos a demostrar que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

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

Vamos a demostrar que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

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

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

Vamos a demostrar que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

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

- Note que de este resultado concluimos que GRAPH-ISO $\in$ AM $\cap$ co-AM

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

Vamos a demostrar que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

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

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

A continuación vamos a ver una herramienta fundamental para la demostración del teorema

Un ingrediente necesario: Funciones de hash

Sea A una matriz Booleana de $n \times m$

Un ingrediente necesario: Funciones de hash

Sea A una matriz Booleana de $n \times m$

La matriz A define una función de hash $h : \{0,1\}^m \rightarrow \{0,1\}^n$

Un ingrediente necesario: Funciones de hash

Sea A una matriz Booleana de $n \times m$

La matriz A define una función de hash $h : \{0,1\}^m \to \{0,1\}^n$

Para cada $u \in \{0,1\}^m$ se tiene que $h(u) = Au$

- Donde la suma y la multiplicación se realizan en el cuerpo $(\{0,1\}, \oplus, \wedge)$

Un ingrediente necesario: Funciones de hash

Sea A una matriz Booleana de $n \times m$

La matriz A define una función de hash $h : \{0,1\}^m \to \{0,1\}^n$

Para cada $u \in \{0,1\}^m$ se tiene que $h(u) = Au$

- Donde la suma y la multiplicación se realizan en el cuerpo $(\{0,1\}, \oplus, \wedge)$

Sea $\mathcal{H}(m,n)$ el conjunto de todas las funciones de hash $h : \{0,1\}^m \to \{0,1\}^n$ definidas a partir de una matriz Booleana de $n \times m$

Un ingrediente necesario: Funciones de hash

Necesitamos introducir notación para las funciones de hash en $\mathcal{H}(m, n)$

Un ingrediente necesario: Funciones de hash

Necesitamos introducir notación para las funciones de hash en $\mathcal{H}(m, n)$

Para un string w: $\pi_k(w)$ es el k-ésimo símbolo de w

Un ingrediente necesario: Funciones de hash

Necesitamos introducir notación para las funciones de hash en $\mathcal{H}(m, n)$

Para un string w: $\pi_k(w)$ es el k-ésimo símbolo de w

Si $h \in \mathcal{H}(m, n)$, entonces para cada $u \in \{0, 1\}^m$ se tiene que $h(u) = v$ con $v \in \{0, 1\}^n$, donde para cada $j \in \{1, \ldots, n\}$:

$$\pi_j(v) = (a_{j,1} \wedge \pi_1(u)) \oplus (a_{j,2} \wedge \pi_2(u)) \oplus \cdots \oplus (a_{j,m} \wedge \pi_m(u))$$

Las funciones de $\mathcal{H}(m, n)$ como funciones de hash

Las funciones de $\mathcal{H}(m, n)$ como funciones de hash

$$h : \{0,1\}^m \to \{0,1\}^n$$

Las funciones de $\mathcal{H}(m, n)$ como funciones de hash

$X \subseteq \{0,1\}^m$

$h : \{0,1\}^m \to \{0,1\}^n$

Las funciones de $\mathcal{H}(m, n)$ como funciones de hash

$$X \subseteq \{0,1\}^m \quad \begin{array}{|c|c|c|c|c|} \hline x_1 & x_2 & x_3 & x_4 & x_5 \\ \hline \end{array}$$

$$h : \{0,1\}^m \to \{0,1\}^n$$

Las funciones de $\mathcal{H}(m, n)$ como funciones de hash

$$X \subseteq \{0,1\}^m \quad \begin{array}{|c|c|c|c|c|} \hline x_1 & x_2 & x_3 & x_4 & x_5 \\ \hline \end{array}$$

$$h : \{0,1\}^m \rightarrow \{0,1\}^n$$

Buckets $\{0,1\}^n$

Las funciones de $\mathcal{H}(m, n)$ como funciones de hash

$X \subseteq \{0,1\}^m$

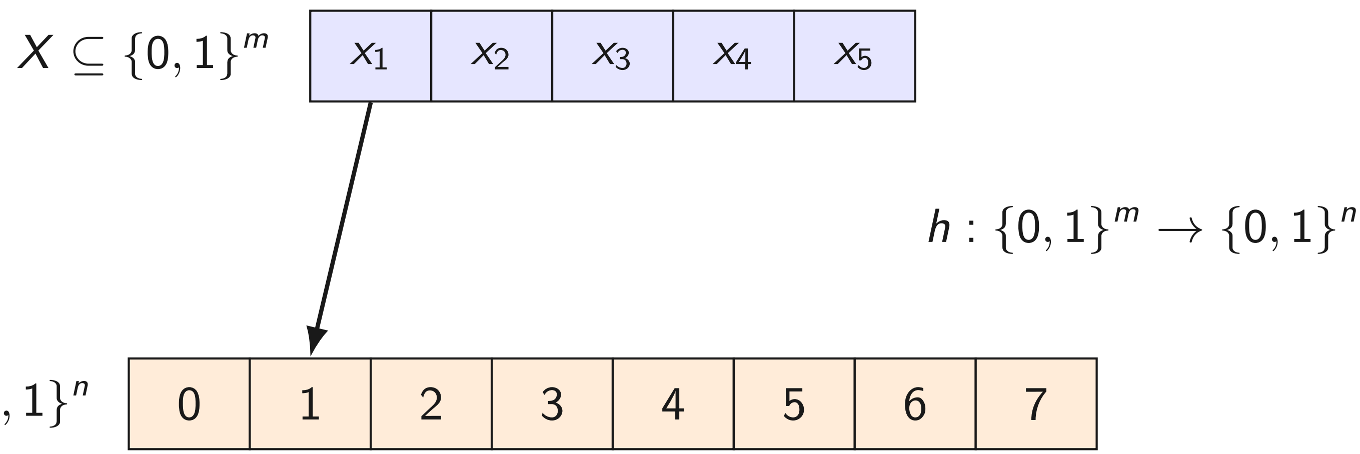
x_1	x_2	x_3	x_4	x_5

$h : \{0,1\}^m \to \{0,1\}^n$

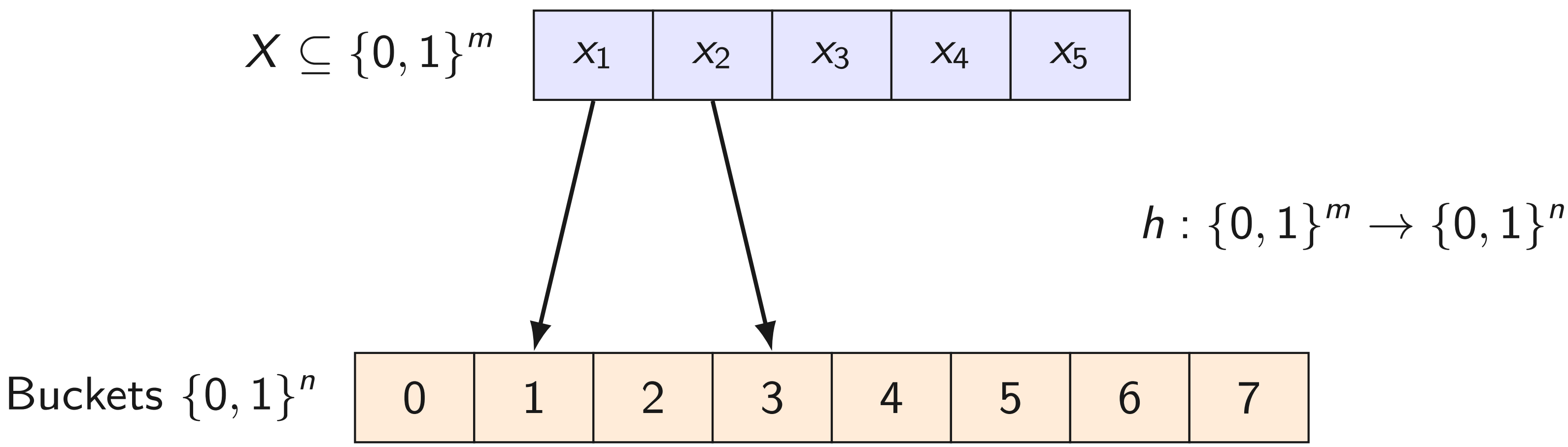
Buckets $\{0,1\}^n$

0	1	2	3	4	5	6	7

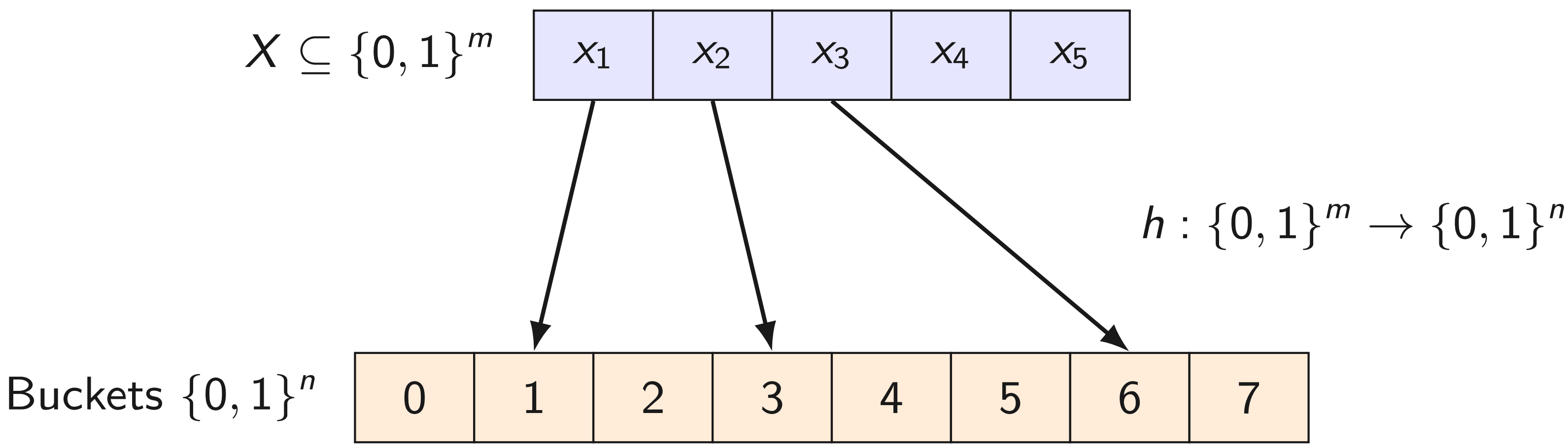
Las funciones de $\mathcal{H}(m,n)$ como funciones de hash



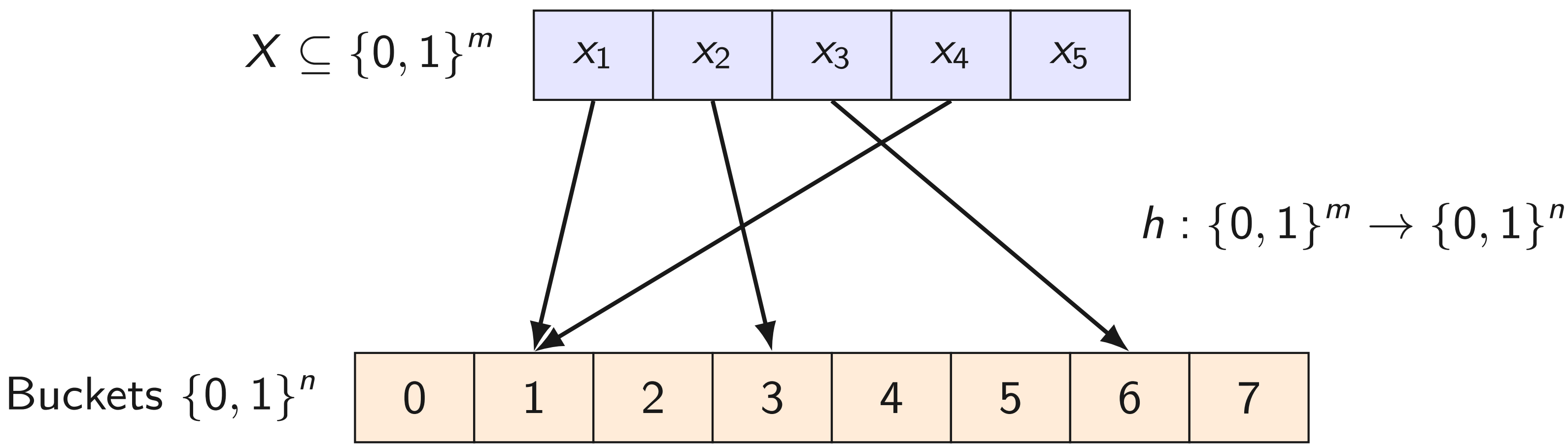
Las funciones de $\mathcal{H}(m, n)$ como funciones de hash



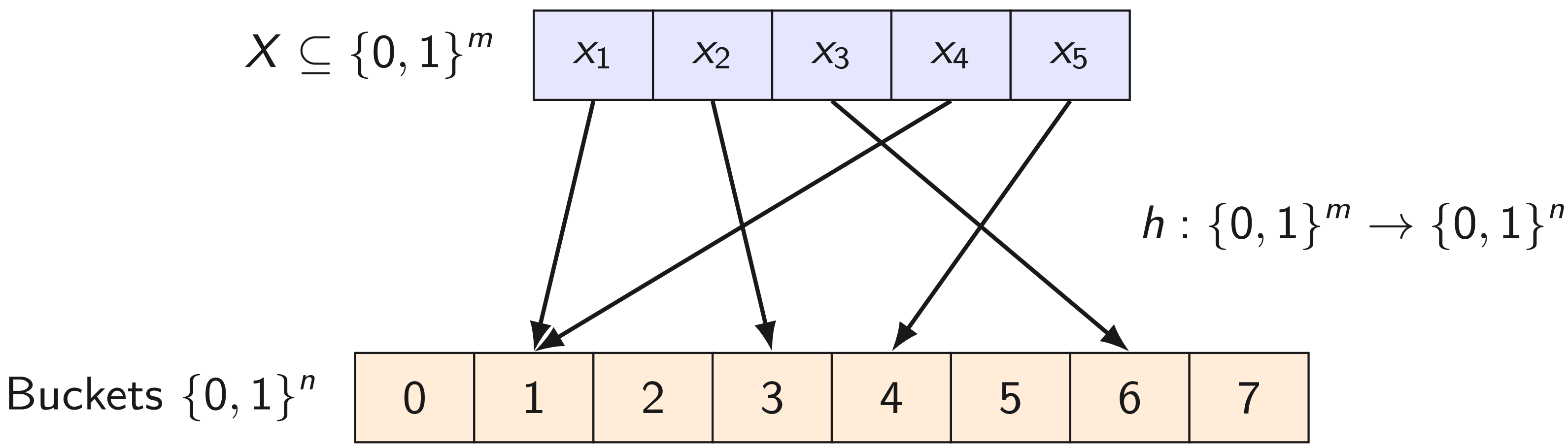
Las funciones de $\mathcal{H}(m, n)$ como funciones de hash



Las funciones de $\mathcal{H}(m, n)$ como funciones de hash



Las funciones de $\mathcal{H}(m, n)$ como funciones de hash



$\mathcal{H}(m, n)$ es una familia de funciones de hashing universal

Teorema

Para cada $u, v \in \{0, 1\}^m$ tales que $u \neq v$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = h(v)) \quad = \quad 2^{-n}$$

$\mathcal{H}(m, n)$ es una familia de funciones de hashing universal

Teorema

Para cada $u, v \in \{0, 1\}^m$ tales que $u \neq v$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = h(v)) \quad = \quad 2^{-n}$$

Elegir $h \sim \mathcal{H}(m, n)$ significa elegir al azar con distribución uniforme y de manera independiente cada uno de los elementos de la matriz Booleana que define a h

$\mathcal{H}(m, n)$ es una familia de funciones de hashing universal

Teorema

Para cada $u, v \in \{0,1\}^m$ tales que $u \neq v$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = h(v)) \quad = \quad 2^{-n}$$

Elegir $h \sim \mathcal{H}(m, n)$ significa elegir al azar con distribución uniforme y de manera independiente cada uno de los elementos de la matriz Booleana que define a h

Este teorema es consecuencia de una serie de propiedades que vamos a demostrar a continuación

$\mathcal{H}(m, n)$ es uniforme

Proposición

Para cada $u \in \{0,1\}^m$ *tal que* $u \neq 0^m$, *y para cada* $r \in \{0,1\}^n$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = r) \quad = \quad 2^{-n}$$

$\mathcal{H}(m, n)$ es uniforme

Proposición

Para cada $u \in \{0,1\}^m$ tal que $u \neq 0^m$, y para cada $r \in \{0,1\}^n$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = r) \quad = \quad 2^{-n}$$

Esta proposición es consecuencia del siguiente lema

Uniformidad: un lema necesario

Lema

Para cada $u \in \{0,1\}^m$ *tal que* $u \neq 0^m$*, y para cada* $b \in \{0,1\}$ *y* $j \in \{1,\ldots,n\}$*:*

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b) \quad = \quad \frac{1}{2}$$

Demostración del lema

Vamos a hacer la demostración por inducción en m

Demostración del lema

Vamos a hacer la demostración por inducción en m

Suponga que $m = 1$. Y sea $u \in \{0, 1\}$ tal que $u \neq 0$, $b \in \{0, 1\}$ y $j \in \{1, \ldots, n\}$

Demostración del lema

Vamos a hacer la demostración por inducción en m

Suponga que $m = 1$. Y sea $u \in \{0, 1\}$ tal que $u \neq 0$, $b \in \{0, 1\}$ y $j \in \{1, \ldots, n\}$

Dado que $u = 1$, tenemos que:

$$\begin{aligned}
\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b) &= \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} \wedge \pi_1(u) = b) \\
&= \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} \wedge u = b) \\
&= \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b) \; = \; \frac{1}{2}
\end{aligned}$$

Demostración del lema

Suponga que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m+1$

Demostración del lema

Suponga que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m + 1$

Sea $u \in \{0, 1\}^{m+1}$ tal que $u \neq 0^{m+1}$, $b \in \{0, 1\}$ y $j \in \{1, \ldots, n\}$

Demostración del lema

Suponga que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m + 1$

Sea $u \in \{0, 1\}^{m+1}$ tal que $u \neq 0^{m+1}$, $b \in \{0, 1\}$ y $j \in \{1, \ldots, n\}$

- Desde ahora en adelante usamos u_{-i} para denotar la palabra en $\{0, 1\}^m$ que resulta de remover la posición i desde u

Demostración del lema

Suponga que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m + 1$

Sea $u \in \{0,1\}^{m+1}$ tal que $u \neq 0^{m+1}$, $b \in \{0,1\}$ y $j \in \{1, \ldots, n\}$

- Desde ahora en adelante usamos u_{-i} para denotar la palabra en $\{0,1\}^m$ que resulta de remover la posición i desde u

Tenemos que:

$$\begin{aligned}\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \ =& \\ & \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b)\end{aligned}$$

Demostración del lema

Si $\pi_{m+1}(u) = 0$, entonces $u_{-(m+1)} \neq 0^m$ y concluimos por hipótesis de inducción que:

$$
\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}\big(\pi_j\big(h\big(u_{-(m+1)}\big)\big) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b\big) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}\big(\pi_j\big(h\big(u_{-(m+1)}\big)\big) = b\big) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b) \ = \ \frac{1}{2}
\end{aligned}
$$

Demostración del lema

Si $\pi_{m+1}(u) = 1$, entonces tenemos que considerar dos casos

Demostración del lema

Si $\pi_{m+1}(u) = 1$, entonces tenemos que considerar dos casos

Si $u_{-(m+1)} = 0^m$, entonces tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}(0 \oplus a_{j,m+1} = b) \ = \\
&\qquad \mathbf{Pr}_{a_{j,m+1}\sim\{0,1\}}(a_{j,m+1} = b) \ = \ \frac{1}{2}
\end{aligned}$$

Demostración del lema

Si $u_{-(m+1)} \neq 0^m$, concluimos por hipótesis de inducción que:

$$
\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \ = \\
&\quad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b) \ = \\
&\quad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus a_{j,m+1} = b) \ = \\
&\quad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) = a_{j,m+1} \oplus b) \ = \\
&\quad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) = a_{j,m+1} \oplus b \mid a_{j,m+1} = 0) \cdot \\
&\qquad\qquad\qquad \mathbf{Pr}_{a_{j,m+1} \sim \{0,1\}}(a_{j,m+1} = 0) + \\
&\qquad\quad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) = a_{j,m+1} \oplus b \mid a_{j,m+1} = 1) \cdot \\
&\qquad\qquad\qquad \mathbf{Pr}_{a_{j,m+1} \sim \{0,1\}}(a_{j,m+1} = 1) \ = \\
&\quad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) = b) \cdot \frac{1}{2} + \\
&\qquad\quad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) = 1 - b) \cdot \frac{1}{2} \ = \\
&\quad \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} \ = \ \frac{1}{2} \qquad \square
\end{aligned}
$$

La demostración de uniformidad

Sea $u \in \{0,1\}^m$ tal que $u \neq 0^m$, y $r \in \{0,1\}^n$

La demostración de uniformidad

Sea $u \in \{0,1\}^m$ tal que $u \neq 0^m$, y $r \in \{0,1\}^n$

Dado que los elementos de la matriz Booleana que define a h son escogidos de manera independientes, concluimos por el lema anterior:

$$\begin{aligned} \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = r) &= \prod_{j=1}^{n} \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = \pi_j(r)) \\ &= 2^{-n} \end{aligned}$$

Las funciones en $\mathcal{H}(m, n)$ son 2-independientes

Teorema

Para cada $u, v \in \{0,1\}^m$ *tales que* $u \neq v$, *y para cada* $r, s \in \{0,1\}^n$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = r \wedge h(v) = s) \ = \ \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = r) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(v) = s)$$

Las funciones en $\mathcal{H}(m, n)$ son 2-independientes

Teorema

Para cada $u, v \in \{0,1\}^m$ *tales que* $u \neq v$*, y para cada* $r, s \in \{0,1\}^n$*:*

$$\mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(u) = r \wedge h(v) = s) \;=\; \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(u) = r) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(v) = s)$$

Esta proposición es consecuencia del siguiente lema

2-independencia: un lema necesario

Lema

Para cada $u, v \in \{0,1\}^m$ *tales que* $u \neq v$*, y para cada* $b, c \in \{0,1\}$ *y* $j \in \{1, \ldots, n\}$*:*

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ = \\ \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(v)) = c)$$

Demostración del lema

Primero consideramos el caso $u = 0^m$

Demostración del lema

Primero consideramos el caso $u = 0^m$

- Tenemos entonces que $v \neq 0^m$

Demostración del lema

Primero consideramos el caso $u = 0^m$

- Tenemos entonces que $v \neq 0^m$

Si $b = 0$, por el lema anterior concluimos:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ = \\ \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 0 \wedge \pi_j(h(v)) = c) = \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(v)) = c) = \frac{1}{2}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b) = \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 0) = 1$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(v)) = c) = \frac{1}{2}$$

Demostración del lema

Primero consideramos el caso $u = 0^m$

- Tenemos entonces que $v \neq 0^m$

Si $b = 0$, por el lema anterior concluimos:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ =$$
$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 0 \wedge \pi_j(h(v)) = c) = \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(v)) = c) = \frac{1}{2}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b) = \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 0) = 1$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(v)) = c) = \frac{1}{2}$$

Por lo tanto, se cumple el lema en este caso

Demostración del lema

Si $b = 1$, por el lema anterior concluimos:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ = \\ \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 1 \wedge \pi_j(h(v)) = c) = 0$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b) = \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 1) = 0$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(v)) = c) = \frac{1}{2}$$

Demostración del lema

Si $b = 1$, por el lema anterior concluimos:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ = \ \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 1 \wedge \pi_j(h(v)) = c) = 0$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(u)) = b) = \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(0 = 1) = 0$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(\pi_j(h(v)) = c) = \frac{1}{2}$$

Por lo tanto, se cumple el lema en este caso

Demostración del lema

Suponemos desde ahora en adelante que $u \neq 0^m$

Demostración del lema

Suponemos desde ahora en adelante que $u \neq 0^m$

Vamos a hacer la demostración por inducción en m

Demostración del lema

Suponemos desde ahora en adelante que $u \neq 0^m$

Vamos a hacer la demostración por inducción en m

Suponga que $m = 1$. Y sean $u, v \in \{0, 1\}$ tales que $u = 1$ y $v = 0$.
Además, sean $b, c \in \{0, 1\}$ y $j \in \{1, \ldots, n\}$

Demostración del lema

Suponemos desde ahora en adelante que $u \neq 0^m$

Vamos a hacer la demostración por inducción en m

Suponga que $m = 1$. Y sean $u, v \in \{0, 1\}$ tales que $u = 1$ y $v = 0$.
Además, sean $b, c \in \{0, 1\}$ y $j \in \{1, \ldots, n\}$

Dado que $u = 1$ y $v = 0$, tenemos que:

$$\begin{aligned}
\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ = \\
\mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} \wedge u = b \wedge a_{j,1} \wedge v = c) \ = \\
\mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b \wedge 0 = c)
\end{aligned}$$

Demostración del lema

Si $c = 0$, entonces:

$$\begin{aligned}\mathbf{Pr}_{h\sim\mathcal{H}(1,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ &= \\ \mathbf{Pr}_{a_{j,1}\sim\{0,1\}}(a_{j,1} = b \wedge 0 = 0) \ &= \\ \mathbf{Pr}_{a_{j,1}\sim\{0,1\}}(a_{j,1} = b) \ &= \ \frac{1}{2}\end{aligned}$$

$$\mathbf{Pr}_{h\sim\mathcal{H}(1,n)}(\pi_j(h(u)) = b) \ = \ \mathbf{Pr}_{a_{j,1}\sim\{0,1\}}(a_{j,1} = b) \ = \ \frac{1}{2}$$

$$\mathbf{Pr}_{h\sim\mathcal{H}(1,n)}(\pi_j(h(v)) = c) \ = \ \mathbf{Pr}_{h\sim\mathcal{H}(1,n)}(0 = 0) \ = \ 1$$

Demostración del lema

Si $c = 0$, entonces:

$$\begin{aligned} \mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ &= \\ \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b \wedge 0 = 0) \ &= \\ \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b) \ &= \ \frac{1}{2} \end{aligned}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b) \ = \ \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b) \ = \ \frac{1}{2}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(v)) = c) \ = \ \mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(0 = 0) \ = \ 1$$

Por lo tanto, el lema se cumple en este caso

Demostración del lema

Si $c = 1$, entonces:

$$\begin{aligned}\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ &= \\ \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b \wedge 0 = 1) \ &= \ 0\end{aligned}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b) = \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b) \ = \ \frac{1}{2}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(v)) = c) \ = \ \mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(0 = 1) \ = \ 0$$

Demostración del lema

Si $c = 1$, entonces:

$$\begin{aligned}\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b \wedge \pi_j(h(v)) = c) \ &= \\ \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b \wedge 0 = 1) \ &= \ 0\end{aligned}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(u)) = b) = \mathbf{Pr}_{a_{j,1} \sim \{0,1\}}(a_{j,1} = b) \ = \ \frac{1}{2}$$

$$\mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(\pi_j(h(v)) = c) \ = \ \mathbf{Pr}_{h \sim \mathcal{H}(1,n)}(0 = 1) \ = \ 0$$

Por lo tanto, el lema también se cumple en este caso

Demostración del lema

Suponemos que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m+1$

Demostración del lema

Suponemos que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m+1$

Sean $u, v \in \{0,1\}^{m+1}$ tales que $u \neq v$, $b, c \in \{0,1\}$ y $j \in \{1,\ldots,n\}$

Demostración del lema

Suponemos que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m+1$

Sean $u, v \in \{0,1\}^{m+1}$ tales que $u \neq v$, $b, c \in \{0,1\}$ y $j \in \{1,\ldots,n\}$

- Sin perdida de generalidad suponemos que $\pi_{m+1}(u) \neq \pi_{m+1}(v)$

Demostración del lema

Suponemos que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m+1$

Sean $u, v \in \{0,1\}^{m+1}$ tales que $u \neq v$, $b, c \in \{0,1\}$ y $j \in \{1, \ldots, n\}$

- Sin perdida de generalidad suponemos que $\pi_{m+1}(u) \neq \pi_{m+1}(v)$

Tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \;=\\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b \wedge \\
&\qquad\qquad \pi_j(h(v_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(v)) = c)
\end{aligned}$$

Demostración del lema

Suponemos que la propiedad es cierta para m. Vamos a demostrar que es cierta para $m + 1$

Sean $u, v \in \{0,1\}^{m+1}$ tales que $u \neq v$, $b, c \in \{0,1\}$ y $j \in \{1, \ldots, n\}$

- Sin perdida de generalidad suponemos que $\pi_{m+1}(u) \neq \pi_{m+1}(v)$

Tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b \ \wedge \\
&\qquad\qquad\qquad\qquad \pi_j(h(v_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(v)) = c)
\end{aligned}$$

En este caso tenemos que considerar cinco subcasos

Demostración del lema

En primer lugar, suponemos que $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} = 0^m$

Demostración del lema

En primer lugar, suponemos que $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} = 0^m$

- Tenemos entonces que $u_{-(m+1)} \neq 0^m$

Demostración del lema

En primer lugar, suponemos que $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} = 0^m$

- Tenemos entonces que $u_{-(m+1)} \neq 0^m$

Por el lema anterior concluimos:

$$\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\quad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b \ \wedge \\
&\qquad\qquad\qquad\qquad \pi_j(h(v_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(v)) = c) \\
&\quad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}(\pi_j(h(u_{-(m+1)})) = b \wedge a_{j,m+1} = c) \ = \\
&\quad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b) \cdot \mathbf{Pr}_{a_{j,m+1}\sim\{0,1\}}(a_{j,m+1} = c) \ = \ \frac{1}{2}\cdot\frac{1}{2} \ = \ \frac{1}{4}
\end{aligned}$$

Demostración del lema

Por el lema anterior, también tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\ a_{j,m+1}\sim\{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b) \ = \ \frac{1}{2}
\end{aligned}$$

$$\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(v)) = c) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\ a_{j,m+1}\sim\{0,1\}}(\pi_j(h(v_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(v)) = b) \ = \\
&\qquad \mathbf{Pr}_{a_{j,m+1}\sim\{0,1\}}(a_{j,m+1} = b) \ = \ \frac{1}{2}
\end{aligned}$$

Demostración del lema

Por el lema anterior, también tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \; = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\; a_{j,m+1}\sim\{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b) \; = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b) \; = \; \frac{1}{2}
\end{aligned}$$

$$\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(v)) = c) \; = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\; a_{j,m+1}\sim\{0,1\}}(\pi_j(h(v_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(v)) = b) \; = \\
&\qquad \mathbf{Pr}_{a_{j,m+1}\sim\{0,1\}}(a_{j,m+1} = b) \; = \; \frac{1}{2}
\end{aligned}$$

Por lo tanto, el lema se cumple en este caso

Demostración del lema

En segundo lugar, suponemos que $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} \neq 0^m$

Demostración del lema

En segundo lugar, suponemos que $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} \neq 0^m$

- Recuerde que $u_{-(m+1)} \neq 0^m$

Demostración del lema

En segundo lugar, suponemos que $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} \neq 0^m$

- Recuerde que $u_{-(m+1)} \neq 0^m$

Tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(u)) = b \wedge \\
&\qquad\qquad\qquad\qquad \pi_j(h(v_{-(m+1)})) \oplus (a_{j,m+1} \wedge \pi_{m+1}(v)) = c) \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) = b \wedge \\
&\qquad\qquad\qquad\qquad \pi_j(h(v_{-(m+1)})) \oplus a_{j,m+1} = c) \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, a_{j,m+1} \sim \{0,1\}}(\pi_j(h(u_{-(m+1)})) = b \wedge \\
&\qquad\qquad\qquad\qquad \pi_j(h(v_{-(m+1)})) = c \oplus a_{j,m+1})
\end{aligned}$$

Demostración del lema

Si $u_{-(m+1)} \neq v_{-(m+1)}$, entonces usando la igualdad anterior, la hipótesis de inducción y el lema anterior:

$$
\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\quad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}(\pi_j(h(u_{-(m+1)})) = b \wedge \\
&\qquad\quad \pi_j(h(v_{-(m+1)})) = c \oplus a_{j,m+1} \mid a_{j,m+1} = 0) \cdot \mathbf{Pr}_{a_{j,m+1}\sim\{0,1\}}(a_{j+1,m} = 0) + \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}(\pi_j(h(u_{-(m+1)})) = b \wedge \\
&\qquad\quad \pi_j(h(v_{-(m+1)})) = c \oplus a_{j,m+1} \mid a_{j,m+1} = 1) \cdot \mathbf{Pr}_{a_{j,m+1}\sim\{0,1\}}(a_{j+1,m} = 1) \ = \\
&\quad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b \wedge \pi_j(h(v_{-(m+1)})) = c) \cdot \frac{1}{2} + \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b \wedge \pi_j(h(v_{-(m+1)})) = 1 - c) \cdot \frac{1}{2} \ = \\
&\quad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(v_{-(m+1)})) = c) \cdot \frac{1}{2} + \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u_{-(m+1)})) = b) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(v_{-(m+1)})) = 1 - c) \cdot \frac{1}{2} \ = \\
&\quad \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \ = \ \frac{1}{4}
\end{aligned}
$$

Demostración del lema

Si $u_{-(m+1)} = v_{-(m+1)}$, entonces usando la igualdad anterior y el lema anterior:

$$
\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}\big(\pi_j\big(h\big(u_{-(m+1)}\big)\big) = b \wedge \\
&\qquad\qquad\qquad\qquad \pi_j\big(h\big(v_{-(m+1)}\big)\big) = c \oplus a_{j,m+1}\big) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}\big(\pi_j\big(h\big(u_{-(m+1)}\big)\big) = b \wedge \\
&\qquad\qquad\qquad\qquad \pi_j\big(h\big(u_{-(m+1)}\big)\big) = c \oplus a_{j,m+1}\big) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}\big(\pi_j\big(h\big(u_{-(m+1)}\big)\big) = b \wedge b = c \oplus a_{j,m+1}\big) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, a_{j,m+1}\sim\{0,1\}}\big(\pi_j\big(h\big(u_{-(m+1)}\big)\big) = b \wedge b \oplus c = a_{j,m+1}\big) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}\big(\pi_j\big(h\big(u_{-(m+1)}\big)\big) = b\big) \cdot \mathbf{Pr}_{a_{j,m+1}\sim\{0,1\}}\big(b \oplus c = a_{j,m+1}\big) \ = \\
&\qquad \frac{1}{2} \cdot \frac{1}{2} \ = \ \frac{1}{4}
\end{aligned}
$$

Demostración del lema

Por otro lado, usando el lema anterior obtenemos:

$$
\begin{aligned}
\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) &= \frac{1}{2} \\
\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(v)) = c) &= \frac{1}{2}
\end{aligned}
$$

Demostración del lema

Por otro lado, usando el lema anterior obtenemos:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \;=\; \frac{1}{2}$$
$$\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(v)) = c) \;=\; \frac{1}{2}$$

Por lo tanto, concluimos que el lema se cumple en este caso

- Tanto bajo la condición de que $u_{-(m+1)} \neq v_{-(m+1)}$ como bajo la condición de que $u_{-(m+1)} = v_{-(m+1)}$

Demostración del lema

En tercer lugar, suponemos que $\pi_{m+1}(u) = 1$ y $v = 0^{m+1}$

Demostración del lema

En tercer lugar, suponemos que $\pi_{m+1}(u) = 1$ y $v = 0^{m+1}$

Si $c = 0$ tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge 0 = 0) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot 1 \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(0 = 0) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m+1,n)}(\pi_{j+1}(h(v)) = c)
\end{aligned}$$

Demostración del lema

Si $c = 1$ tenemos que:

$$\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge 0 = 1) \ = \\
&\qquad 0 \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot 0 \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(0 = 1) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_{j+1}(h(v)) = c)
\end{aligned}$$

Demostración del lema

Si $c = 1$ tenemos que:

$$
\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge \pi_{j+1}(h(v)) = c) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b \wedge 0 = 1) \ = \\
&\qquad 0 \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot 0 \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(0 = 1) \ = \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_j(h(u)) = b) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m+1,n)}(\pi_{j+1}(h(v)) = c)
\end{aligned}
$$

Por lo tanto el lema se cumple en este caso

Demostración del lema

En cuarto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} = 0^m$ y $v_{-(m+1)} \neq 0^m$

Demostración del lema

En cuarto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} = 0^m$ y $v_{-(m+1)} \neq 0^m$

- La demostración de esta caso es análoga a la del caso $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} = 0^m$, ya que sabíamos que $u_{-(m+1)} \neq 0^m$ en este caso

Demostración del lema

En cuarto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} = 0^m$ y $v_{-(m+1)} \neq 0^m$

- La demostración de esta caso es análoga a la del caso $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} = 0^m$, ya que sabíamos que $u_{-(m+1)} \neq 0^m$ en este caso

En quinto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} \neq 0^m$ y $v_{-(m+1)} \neq 0^m$

Demostración del lema

En cuarto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} = 0^m$ y $v_{-(m+1)} \neq 0^m$

- La demostración de esta caso es análoga a la del caso $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} = 0^m$, ya que sabíamos que $u_{-(m+1)} \neq 0^m$ en este caso

En quinto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} \neq 0^m$ y $v_{-(m+1)} \neq 0^m$

- La demostración de esta caso es análoga a la del caso $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} \neq 0^m$, ya que sabíamos que $u_{-(m+1)} \neq 0^m$ en este caso

Demostración del lema

En cuarto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} = 0^m$ y $v_{-(m+1)} \neq 0^m$

- La demostración de esta caso es análoga a la del caso $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} = 0^m$, ya que sabíamos que $u_{-(m+1)} \neq 0^m$ en este caso

En quinto lugar, suponemos que $\pi_{m+1}(u) = 1$, $\pi_{m+1}(v) = 0$, $u_{-(m+1)} \neq 0^m$ y $v_{-(m+1)} \neq 0^m$

- La demostración de esta caso es análoga a la del caso $\pi_{m+1}(u) = 0$, $\pi_{m+1}(v) = 1$ y $v_{-(m+1)} \neq 0^m$, ya que sabíamos que $u_{-(m+1)} \neq 0^m$ en este caso

Esto concluye la demostración del lema $\square$

La demostración de 2-independencia

Sea $u, v \in \{0,1\}^m$ tales que $u \neq v$, y $r, s \in \{0,1\}^n$

La demostración de 2-independencia

Sea $u, v \in \{0,1\}^m$ tales que $u \neq v$, y $r, s \in \{0,1\}^n$

Dado que los elementos de la matriz Booleana que define a h son escogidos de manera independientes, concluimos por el lema anterior:

$$
\begin{aligned}
&\mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(u) = r \wedge h(v) = s) \;= \\
&\qquad \prod_{j=1}^{n} \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u)) = \pi_j(r) \wedge \pi_j(h(v)) = \pi_j(s)) \;= \\
&\qquad \prod_{j=1}^{n} \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u)) = \pi_j(r)) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(v)) = \pi_j(s)) \;= \\
&\qquad \prod_{j=1}^{n} \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(u)) = \pi_j(r)) \cdot \prod_{j=1}^{n} \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(\pi_j(h(v)) = \pi_j(s)) \;= \\
&\qquad \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(u) = r) \cdot \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(v) = s)
\end{aligned}
$$

Uniformidad y 2-independencia $\Rightarrow \mathcal{H}(m, n)$ hashing universal

Tenemos que demostrar que para cada $u, v \in \{0,1\}^m$ tales que $u \neq v$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = h(v)) \quad = \quad 2^{-n}$$

Uniformidad y 2-independencia $\Rightarrow \mathcal{H}(m, n)$ hashing universal

Tenemos que demostrar que para cada $u, v \in \{0,1\}^m$ tales que $u \neq v$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = h(v)) \quad = \quad 2^{-n}$$

Vamos a demostrar que esta propiedad es consecuencia de uniformidad y 2-independencia

Uniformidad y 2-independencia $\Rightarrow \mathcal{H}(m,n)$ hashing universal

Por uniformidad y 2-independencia, tenemos que:

$$\begin{aligned}
\mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(u)=h(v)) &= \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}\Big(\bigvee_{r\in\{0,1\}^n} h(u)=r \wedge h(v)=r\Big) \\
&= \sum_{r\in\{0,1\}} \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(u)=r \wedge h(v)=r) \\
&= \sum_{r\in\{0,1\}} \mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(u)=r)\cdot\mathbf{Pr}_{h\sim\mathcal{H}(m,n)}(h(v)=r) \\
&= \sum_{r\in\{0,1\}} 2^{-n}\cdot 2^{-n} \;=\; 2^{n}\cdot 2^{-2n} \;=\; 2^{-n}
\end{aligned}$$

Algunas consecuencias útiles

Proposición

Para cada $u \in \{0,1\}^m$*:*

$$\mathbf{Pr}_{h\sim\mathcal{H}(m,n),\, r\sim\{0,1\}^n}(h(u) = r) \quad = \quad 2^{-n}$$

Algunas consecuencias útiles

Para demostrar la propiedad consideramos dos casos

Algunas consecuencias útiles

Para demostrar la propiedad consideramos dos casos

Si $u = 0^m$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r) \;=\; \mathbf{Pr}_{r \sim \{0,1\}^n}(0^n = r) \;=\; 2^{-n}$$

Algunas consecuencias útiles

Para demostrar la propiedad consideramos dos casos

Si $u = 0^m$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r) \; = \; \mathbf{Pr}_{r \sim \{0,1\}^n}(0^n = r) \; = \; 2^{-n}$$

Si $u \neq 0^m$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r) \; =$$

$$\sum_{s \in \{0,1\}^n} \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r \mid r = s) \cdot \mathbf{Pr}_{r \sim \{0,1\}^n}(r = s) \; =$$

$$\sum_{s \in \{0,1\}^n} \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = s) \cdot 2^{-n} \; =$$

$$2^{-n} \sum_{s \in \{0,1\}^n} 2^{-n} \; = \; 2^n \cdot 2^{-2n} \; = \; 2^{-n}$$

Algunas consecuencias útiles

Proposición

Para cada $u, v \in \{0,1\}^m$ tales que $u \neq v$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r \wedge h(v) = r) \quad = \quad 2^{-2n}$$

Algunas consecuencias útiles

Para demostrar la propiedad consideramos tres casos

Algunas consecuencias útiles

Para demostrar la propiedad consideramos tres casos

Si $u = 0^m$ y $v \neq 0^m$:

$$\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r \wedge h(v) = r) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(0^n = r \wedge h(v) = r) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(0^n = r \wedge h(v) = 0^n) \ = \\
&\qquad \mathbf{Pr}_{r \sim \{0,1\}^n}(0^n = r) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(v) = 0^n) \ = \ 2^{-n} \cdot 2^{-n} \ = \ 2^{-2n}
\end{aligned}$$

Algunas consecuencias útiles

Para demostrar la propiedad consideramos tres casos

Si $u = 0^m$ y $v \neq 0^m$:

$$
\begin{aligned}
&\mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r \wedge h(v) = r) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(0^n = r \wedge h(v) = r) \ = \\
&\qquad \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(0^n = r \wedge h(v) = 0^n) \ = \\
&\qquad \mathbf{Pr}_{r \sim \{0,1\}^n}(0^n = r) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(v) = 0^n) \ = \ 2^{-n} \cdot 2^{-n} \ = \ 2^{-2n}
\end{aligned}
$$

Si $u \neq 0^m$ y $v = 0^m$, la demostración se hace de forma análoga

Algunas consecuencias útiles

Si $u \neq 0^m$ y $v \neq 0^m$:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r \wedge h(v) = r) \ =$$

$$\sum_{s \in \{0,1\}^n} \mathbf{Pr}_{h \sim \mathcal{H}(m,n),\, r \sim \{0,1\}^n}(h(u) = r \wedge h(v) = r \mid r = s) \cdot$$

$$\mathbf{Pr}_{r \sim \{0,1\}^n}(r = s) \ =$$

$$\sum_{s \in \{0,1\}^n} \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = s \wedge h(v) = s) \cdot 2^{-n} \ =$$

$$2^{-n} \sum_{s \in \{0,1\}^n} \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(u) = s) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,n)}(h(v) = s) \ =$$

$$2^{-n} \sum_{s \in \{0,1\}^n} 2^{-n} \cdot 2^{-n} \ = \ 2^{-n} \cdot 2^n \cdot 2^{-2n} \ = \ 2^{-2n}$$

¿Cómo se utiliza una familia de funciones de hashing universal?

Proposición

Sea $X \subseteq \{0,1\}^m$ *tal que* $2^{k-1} \leq |X| < 2^k$ *para* $k \in \{1, \ldots, m+1\}$*. Entonces:*

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,k+1),\, r \sim \{0,1\}^{k+1}}(|\{x \in X \mid h(x) = r\}| = 1) \quad \geq \quad \frac{1}{8}$$

La demostración de la proposición

Sea $s = |X|$

La demostración de la proposición

Sea $s = |X|$

- Sabemos que $2^{k-1} \leq s < 2^k$

La demostración de la proposición

Sea $s = |X|$

- Sabemos que $2^{k-1} \leq s < 2^k$

Dado $h \in \mathcal{H}(m, k+1)$ y $r \in \{0,1\}^{k+1}$, defina:

$$Y(h, r) = |\{x \in X \mid h(x) = r\}|$$

La demostración de la proposición

Sea $s = |X|$

- Sabemos que $2^{k-1} \leq s < 2^k$

Dado $h \in \mathcal{H}(m, k+1)$ y $r \in \{0,1\}^{k+1}$, defina:

$$Y(h, r) \quad = \quad |\{x \in X \mid h(x) = r\}|$$

Tenemos que demostrar que $\mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) \geq \frac{1}{8}$

La demostración de la proposición

Dado $x \in X$, $h \in \mathcal{H}(m, k+1)$ y $r \in \{0,1\}^{k+1}$, defina:

$$I_x(h, r) = \begin{cases} 1 & h(x) = r \\ 0 & \text{en otro caso} \end{cases}$$

La demostración de la proposición

Dado $x \in X$, $h \in \mathcal{H}(m, k+1)$ y $r \in \{0,1\}^{k+1}$, defina:

$$I_x(h, r) \quad = \quad \begin{cases} 1 & h(x) = r \\ 0 & \text{en otro caso} \end{cases}$$

Tenemos que $Y = \sum_{x \in X} I_x$

La demostración de la proposición

Para la variable aleatoria I_x tenemos que:

$$\begin{aligned} E[I_x] &= 0 \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_x = 0) \ + \\ &\qquad\qquad\qquad 1 \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_x = 1) \\ &= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_x = 1) \\ &= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(h(x) = r) \\ &= 2^{-(k+1)} \end{aligned}$$

La demostración de la proposición

Para la variable aleatoria I_x tenemos que:

$$
\begin{aligned}
E[I_x] &= 0 \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_x = 0) \ + \\
&\qquad\qquad 1 \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_x = 1) \\
&= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_x = 1) \\
&= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(h(x) = r) \\
&= 2^{-(k+1)}
\end{aligned}
$$

Por lo tanto:

$$
E[Y] \ = \ E\left[\sum_{x \in X} I_x\right] \ = \ \sum_{x \in X} E[I_x] \ = \ \sum_{x \in X} 2^{-(k+1)} \ = \ s 2^{-(k+1)}
$$

La demostración de la proposición

Suponiendo que $x_1 \neq x_2$ para $x_1, x_2 \in X$, tenemos que:

$$
\begin{aligned}
E[I_{x_1} I_{x_2}] &= 0 \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_{x_1} I_{x_2} = 0) + \\
&\qquad\qquad 1 \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_{x_1} I_{x_2} = 1) \\
&= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_{x_1} I_{x_2} = 1) \\
&= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(I_{x_1} = 1 \wedge I_{x_2} = 1) \\
&= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(h(x_1) = r \wedge h(x_2) = r) \\
&= 2^{-2(k+1)}
\end{aligned}
$$

La demostración de la proposición

Considerando que $I_x = I_x^2$, obtenemos que:

$$
\begin{aligned}
E[Y^2] &= E\left[\left(\sum_{x \in X} I_x\right)^2\right] \\
&= E\left[\sum_{x_1, x_2 \in X \,:\, x_1 \neq x_2} I_{x_1} I_{x_2} + \sum_{x \in X} I_x^2\right] \\
&= E\left[\sum_{x_1, x_2 \in X \,:\, x_1 \neq x_2} I_{x_1} I_{x_2} + \sum_{x \in X} I_x\right] \\
&= \sum_{x_1, x_2 \in X \,:\, x_1 \neq x_2} E[I_{x_1} I_{x_2}] + \sum_{x \in X} E[I_x] \\
&= \sum_{x_1, x_2 \in X \,:\, x_1 \neq x_2} 2^{-2(k+1)} + \sum_{x \in X} 2^{-(k+1)} \\
&= s(s-1)2^{-2(k+1)} + s2^{-(k+1)}
\end{aligned}
$$

La demostración de la proposición

Los cálculos anteriores son útiles por la siguiente relación:

$$
\begin{aligned}
E[Y] &= \sum_{i=0}^{s} i \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = i) \\
&= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) + \\
&\qquad\qquad \sum_{i=2}^{s} i \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = i) \\
&\leq \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) + \\
&\qquad\qquad \sum_{i=0}^{s} i(i-1) \cdot \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = i) \\
&= \mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) + E[Y(Y-1)]
\end{aligned}
$$

La demostración de la proposición

De lo anterior concluimos que:

$$
\begin{aligned}
\mathbf{Pr}_{h\sim\mathcal{H}(m,k+1),r\sim\{0,1\}^{k+1}}(Y=1) &\geq E[Y]-E[Y(Y-1)]\\
&= E[Y]-E[Y^2-Y]\\
&= 2E[Y]-E[Y^2]\\
&= 2s2^{-(k+1)}-s(s-1)2^{-2(k+1)}-s2^{-(k+1)}\\
&= s2^{-(k+1)}-s(s-1)2^{-2(k+1)}\\
&\geq s2^{-(k+1)}-s^2 2^{-2(k+1)}
\end{aligned}
$$

La demostración de la proposición

Como $2^{k-1} \leq s < 2^k$, sabemos que $\frac{1}{4} \leq s2^{-(k+1)} < \frac{1}{2}$

La demostración de la proposición

Como $2^{k-1} \leq s < 2^k$, sabemos que $\frac{1}{4} \leq s2^{-(k+1)} < \frac{1}{2}$

Tenemos entonces que:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) \quad \geq \quad \mu - \mu^2,$$

donde $\mu \in [\frac{1}{4}, \frac{1}{2})$

La demostración de la proposición

Como $2^{k-1} \leq s < 2^k$, sabemos que $\frac{1}{4} \leq s2^{-(k+1)} < \frac{1}{2}$

Tenemos entonces que:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) \quad \geq \quad \mu - \mu^2,$$

donde $\mu \in [\frac{1}{4}, \frac{1}{2})$

El menor valor de $\mu - \mu^2$ se obtiene para $\mu = \frac{1}{4}$, por lo que:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) \ \geq \ \frac{1}{4} - \left(\frac{1}{4}\right)^2 \ = \ \frac{3}{16} \ > \ \frac{1}{8}$$

La demostración de la proposición

Como $2^{k-1} \leq s < 2^k$, sabemos que $\frac{1}{4} \leq s2^{-(k+1)} < \frac{1}{2}$

Tenemos entonces que:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) \quad \geq \quad \mu - \mu^2,$$

donde $\mu \in [\frac{1}{4}, \frac{1}{2})$

El menor valor de $\mu - \mu^2$ se obtiene para $\mu = \frac{1}{4}$, por lo que:

$$\mathbf{Pr}_{h \sim \mathcal{H}(m,k+1), r \sim \{0,1\}^{k+1}}(Y = 1) \ \geq \ \frac{1}{4} - \left(\frac{1}{4}\right)^2 \ = \ \frac{3}{16} \ > \ \frac{1}{8}$$

Esto concluye la demostración de la proposición $\square$

Una aplicación de las funciones de hash

Para dar intuición sobre cómo se utilizan las funciones de hash vamos a ver una aplicación sobre el problema de satisfacibilidad

Una aplicación de las funciones de hash

Para dar intuición sobre cómo se utilizan las funciones de hash vamos a ver una aplicación sobre el problema de satisfacibilidad

Considere el problema de verificar si una fórmula proposicional φ en CNF es satisfacible bajo la promesa de que hay a lo más una asignación que satisface φ

Una aplicación de las funciones de hash

Para dar intuición sobre cómo se utilizan las funciones de hash vamos a ver una aplicación sobre el problema de satisfacibilidad

Considere el problema de verificar si una fórmula proposicional φ en CNF es satisfacible bajo la promesa de que hay a lo más una asignación que satisface φ

- Este problema se conoce como unambiguous CNF-SAT (U-CNF-SAT)

Una aplicación de las funciones de hash

Para dar intuición sobre cómo se utilizan las funciones de hash vamos a ver una aplicación sobre el problema de satisfacibilidad

Considere el problema de verificar si una fórmula proposicional φ en CNF es satisfacible bajo la promesa de que hay a lo más una asignación que satisface φ

- Este problema se conoce como unambiguous CNF-SAT (U-CNF-SAT)

¿Es U-CNF-SAT más fácil de resolver que CNF-SAT?

Una aplicación de las funciones de hash

Queremos demostrar que U-CNF-SAT es tan difícil como CNF-SAT

Una aplicación de las funciones de hash

Queremos demostrar que U-CNF-SAT es tan difícil como CNF-SAT

- En particular, nos gustaría dar una reducción de tiempo polinomial de CNF-SAT a U-CNF-SAT

Una aplicación de las funciones de hash

Queremos demostrar que U-CNF-SAT es tan difícil como CNF-SAT

- En particular, nos gustaría dar una reducción de tiempo polinomial de CNF-SAT a U-CNF-SAT

Defina #CNF-SAT(φ) como el número de asignaciones que satisfacen a una fórmula φ en CNF

Una aplicación de las funciones de hash

Queremos demostrar que U-CNF-SAT es tan difícil como CNF-SAT

- En particular, nos gustaría dar una reducción de tiempo polinomial de CNF-SAT a U-CNF-SAT

Defina #CNF-SAT(φ) como el número de asignaciones que satisfacen a una fórmula φ en CNF

Para hacer la reducción necesitamos tener una definición de U-CNF-SAT sin promesa

Una aplicación de las funciones de hash

Queremos demostrar que U-CNF-SAT es tan difícil como CNF-SAT

- En particular, nos gustaría dar una reducción de tiempo polinomial de CNF-SAT a U-CNF-SAT

Defina #CNF-SAT(φ) como el número de asignaciones que satisfacen a una fórmula φ en CNF

Para hacer la reducción necesitamos tener una definición de U-CNF-SAT sin promesa

- La reducción puede generar fórmulas φ tales que #CNF-SAT(φ) ≥ 2

Una aplicación de las funciones de hash

Queremos demostrar que U-CNF-SAT es tan difícil como CNF-SAT

- En particular, nos gustaría dar una reducción de tiempo polinomial de CNF-SAT a U-CNF-SAT

Defina #CNF-SAT(φ) como el número de asignaciones que satisfacen a una fórmula φ en CNF

Para hacer la reducción necesitamos tener una definición de U-CNF-SAT sin promesa

- La reducción puede generar fórmulas φ tales que #CNF-SAT(φ) ≥ 2
- Pero la reducción no debería depender de las respuestas para las fórmulas φ tales que #CNF-SAT(φ) ≥ 2

Una aplicación de las funciones de hash

Dado $H \subseteq \{\psi \mid \psi$ es una fórmula en CNF tal que $\#\text{CNF-SAT}(\psi) \geq 2\}$, defina:

$$\text{U-CNF-SAT}_H \ = \ \text{U-CNF-SAT} \cup H$$

Una aplicación de las funciones de hash

Dado $H \subseteq \{\psi \mid \psi$ es una fórmula en CNF tal que $\#\text{CNF-SAT}(\psi) \geq 2\}$, defina:

$$\text{U-CNF-SAT}_H \ = \ \text{U-CNF-SAT} \cup H$$

Queremos demostrar que CNF-SAT se puede reducir a U-CNF-SAT_H para cada conjunto H

Una aplicación de las funciones de hash

Dado $H \subseteq \{\psi \mid \psi$ es una fórmula en CNF tal que $\#\text{CNF-SAT}(\psi) \geq 2\}$, defina:

$$\text{U-CNF-SAT}_H \ = \ \text{U-CNF-SAT} \cup H$$

Queremos demostrar que CNF-SAT se puede reducir a U-CNF-SAT$_H$ para cada conjunto H

- De hecho, queremos demostrar que la reducción es la misma para todos los problemas U-CNF-SAT$_H$

Una aplicación de las funciones de hash

Teorema (Valiant-Vazirani)

Existe una MT probabilística M con oráculo tal que $t_M(n)$ es $O(n^k)$ y para cada

$$H \subseteq \{\psi \mid \psi \text{ es una fórmula en CNF tal que } \#\text{CNF-SAT}(\psi) \geq 2\}$$

y cada fórmula φ en CNF:

Una aplicación de las funciones de hash

Teorema (Valiant-Vazirani)

Existe una MT probabilística M con oráculo tal que $t_M(n)$ es $O(n^k)$ y para cada

$$H \subseteq \{\psi \mid \psi \text{ es una fórmula en CNF tal que } \#\text{CNF-SAT}(\psi) \geq 2\}$$

y cada fórmula φ en CNF:

- *Si $\varphi \in$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) \geq \frac{3}{4}$*

Una aplicación de las funciones de hash

Teorema (Valiant-Vazirani)

Existe una MT probabilística M con oráculo tal que $t_M(n)$ es $O(n^k)$ y para cada

$$H \subseteq \{\psi \mid \psi \text{ es una fórmula en CNF tal que } \#\text{CNF-SAT}(\psi) \geq 2\}$$

y cada fórmula φ en CNF:

- *Si $\varphi \in$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) \geq \frac{3}{4}$*
- *Si $\varphi \notin$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) = 0$*

Una aplicación de las funciones de hash

Teorema (Valiant-Vazirani)

Existe una MT probabilística M con oráculo tal que $t_M(n)$ es $O(n^k)$ y para cada

$$H \subseteq \{\psi \mid \psi \text{ es una fórmula en CNF tal que } \#\text{CNF-SAT}(\psi) \geq 2\}$$

y cada fórmula φ en CNF:

- *Si $\varphi \in$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) \geq \frac{3}{4}$*
- *Si $\varphi \notin$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) = 0$*

El teorema nos dice que si U-CNF-SAT $\in$ P, entonces CNF-SAT $\in$ RP

Una aplicación de las funciones de hash

Teorema (Valiant-Vazirani)

Existe una MT probabilística M con oráculo tal que $t_M(n)$ es $O(n^k)$ y para cada

$$H \subseteq \{\psi \mid \psi \text{ es una fórmula en CNF tal que } \#\text{CNF-SAT}(\psi) \geq 2\}$$

y cada fórmula φ en CNF:

- *Si $\varphi \in$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H}$ acepte $\varphi) \geq \frac{3}{4}$*
- *Si $\varphi \notin$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H}$ acepte $\varphi) = 0$*

El teorema nos dice que si U-CNF-SAT $\in$ P, entonces CNF-SAT $\in$ RP

- Independientemente de las respuestas que damos para las fórmulas ψ tales que $\#$CNF-SAT$(\psi) \geq 2$

La demostración del teorema

El ingrediente esencial de la demostración es el siguiente lema:

Lema

Existe un algoritmo aleatorizado de tiempo polinomial que, dada una fórmula proposicional φ en CNF con n variables, genera una secuencia de fórmulas φ_1, ..., φ_n, φ_{n+1}, φ_{n+2} en CNF tales que:

1. *Si φ es consistente, entonces*

$$\mathbf{Pr}\left(\bigvee_{i=1}^{n+2} \#\textit{CNF-SAT}(\varphi_i) = 1\right) \quad \geq \quad \frac{1}{8}$$

2. *Si φ es inconsistente, entonces cada fórmula φ_i $(i \in \{1, \ldots, n+2\})$ es inconsistente.*

La demostración del lema

Sea $\varphi(x_1, \ldots, x_n)$ una fórmula en CNF que menciona a las variables proposicionales x_1, ..., x_n

La demostración del lema

Sea $\varphi(x_1, \ldots, x_n)$ una fórmula en CNF que menciona a las variables proposicionales x_1, ..., x_n

Para cada $k \in \{1, \ldots, n+2\}$, defina:

$$\varphi_k(x_1, \ldots, x_n) \ = \ \varphi(x_1, \ldots, x_n) \wedge h_k(x_1, \ldots, x_n) = r_k,$$

donde:

La demostración del lema

Sea $\varphi(x_1, \ldots, x_n)$ una fórmula en CNF que menciona a las variables proposicionales x_1, ..., x_n

Para cada $k \in \{1, \ldots, n+2\}$, defina:

$$\varphi_k(x_1, \ldots, x_n) \ = \ \varphi(x_1, \ldots, x_n) \wedge h_k(x_1, \ldots, x_n) = r_k,$$

donde:

- h_k es elegida al azar con distribución uniforme desde $\mathcal{H}(n, k)$

La demostración del lema

Sea $\varphi(x_1, \ldots, x_n)$ una fórmula en CNF que menciona a las variables proposicionales x_1, ..., x_n

Para cada $k \in \{1, \ldots, n+2\}$, defina:

$$\varphi_k(x_1, \ldots, x_n) \ = \ \varphi(x_1, \ldots, x_n) \wedge h_k(x_1, \ldots, x_n) = r_k,$$

donde:

- h_k es elegida al azar con distribución uniforme desde $\mathcal{H}(n, k)$
- r_k es elegido al azar con distribución uniforme desde $\{0, 1\}^k$

La demostración del lema

Suponiendo que h_k es definida por la matriz Booleana $A = (a_{i,j})$ de $k \times n$, la siguiente fórmula proposicional representa al término $h_k(x_1, \ldots, x_n) = r_k$:

$$\alpha_k \; = \; \bigwedge_{i=1}^{k} \Big[\big((a_{i,1} \wedge x_1) \oplus \cdots \oplus (a_{i,n} \wedge x_n) \big) \leftrightarrow \pi_i(r_k) \Big]$$

La demostración del lema

Suponiendo que h_k es definida por la matriz Booleana $A = (a_{i,j})$ de $k \times n$, la siguiente fórmula proposicional representa al término $h_k(x_1, \ldots, x_n) = r_k$:

$$\alpha_k \;=\; \bigwedge_{i=1}^{k} \Big[\big((a_{i,1} \wedge x_1) \oplus \cdots \oplus (a_{i,n} \wedge x_n)\big) \leftrightarrow \pi_i(r_k) \Big]$$

La fórmula α_k puede transformarse en tiempo polinomial en una fórmula β_k en CNF usando la reducción usual desde lógica proposicional a CNF (transformación de Tseytin)

La demostración del lema

Suponiendo que h_k es definida por la matriz Booleana $A = (a_{i,j})$ de $k \times n$, la siguiente fórmula proposicional representa al término $h_k(x_1, \ldots, x_n) = r_k$:

$$\alpha_k \;=\; \bigwedge_{i=1}^{k} \Big[\big((a_{i,1} \wedge x_1) \oplus \cdots \oplus (a_{i,n} \wedge x_n)\big) \leftrightarrow \pi_i(r_k) \Big]$$

La fórmula α_k puede transformarse en tiempo polinomial en una fórmula β_k en CNF usando la reducción usual desde lógica proposicional a CNF (transformación de Tseytin)

- ¿Cuál es la relación entre α_k y β_k?

La demostración del lema

Suponiendo que h_k es definida por la matriz Booleana $A = (a_{i,j})$ de $k \times n$, la siguiente fórmula proposicional representa al término $h_k(x_1, \ldots, x_n) = r_k$:

$$\alpha_k \;=\; \bigwedge_{i=1}^{k} \Big[\big((a_{i,1} \wedge x_1) \oplus \cdots \oplus (a_{i,n} \wedge x_n)\big) \leftrightarrow \pi_i(r_k)\Big]$$

La fórmula α_k puede transformarse en tiempo polinomial en una fórmula β_k en CNF usando la reducción usual desde lógica proposicional a CNF (transformación de Tseytin)

- ¿Cuál es la relación entre α_k y β_k?

Por lo tanto las fórmulas φ_1, ..., φ_{n+2} están bien definidas

La demostración del lema

Si φ es inconsistente, entonces φ_1, ..., φ_{n+2} son todas fórmulas inconsistentes

La demostración del lema

Si φ es inconsistente, entonces φ_1, ..., φ_{n+2} son todas fórmulas inconsistentes

Suponga que φ es consistente, y defina:

$$X = \{x \in \{0,1\}^n \mid \sigma(\varphi_{k+1}) = 1, \text{ donde } \sigma(x_i) = \pi_i(x) \text{ para cada } i \in \{1, \dots, n\}\}$$

La demostración del lema

Si φ es inconsistente, entonces φ_1, ..., φ_{n+2} son todas fórmulas inconsistentes

Suponga que φ es consistente, y defina:

$$X \;=\; \{x \in \{0,1\}^n \mid \sigma(\varphi_{k+1}) = 1, \\ \text{donde } \sigma(x_i) = \pi_i(x) \text{ para cada } i \in \{1,\ldots,n\}\}$$

Sabemos que $2^{k-1} \leq |X| < 2^k$ para algún $k \in \{1,\ldots,n+1\}$.

La demostración del lema

Si φ es inconsistente, entonces φ_1, ..., φ_{n+2} son todas fórmulas inconsistentes

Suponga que φ es consistente, y defina:

$$X \;=\; \{x \in \{0,1\}^n \mid \sigma(\varphi_{k+1}) = 1, \\ \text{donde } \sigma(x_i) = \pi_i(x) \text{ para cada } i \in \{1,\ldots,n\}\}$$

Sabemos que $2^{k-1} \leq |X| < 2^k$ para algún $k \in \{1,\ldots,n+1\}$. Por lo tanto, por la proposición anterior tenemos que:

$$\mathbf{Pr}_{h\sim\mathcal{H}(n,k+1),r\in\{0,1\}^{k+1}}(|\{x \in X \mid h(x) = r\}| = 1) \quad \geq \quad \frac{1}{8}$$

La demostración del lema

Concluimos que: $\mathbf{Pr}(\#\text{CNF-SAT}(\varphi_{k+1}) = 1) \;\geq\; \dfrac{1}{8}$

La demostración del lema

Concluimos que: $\mathbf{Pr}(\#\text{CNF-SAT}(\varphi_{k+1}) = 1) \;\geq\; \frac{1}{8}$

De esto se deduce que:

$$\mathbf{Pr}\left(\bigvee_{i=1}^{n+2} \#\text{CNF-SAT}(\varphi_i) = 1\right) \;\geq\; \mathbf{Pr}(\#\text{CNF-SAT}(\varphi_{k+1}) = 1) \;\geq\; \frac{1}{8}$$

La demostración del lema

Concluimos que: $\mathbf{Pr}(\#\text{CNF-SAT}(\varphi_{k+1}) = 1) \ \geq \ \dfrac{1}{8}$

De esto se deduce que:

$$\mathbf{Pr}\left(\bigvee_{i=1}^{n+2} \#\text{CNF-SAT}(\varphi_i) = 1\right) \ \geq \ \mathbf{Pr}(\#\text{CNF-SAT}(\varphi_{k+1}) = 1) \ \geq \ \frac{1}{8}$$

Esto concluye la demostración del lema $\square$

La demostración del teorema de Valiant-Vazirani

Para terminar con la demostración necesitamos construir una MT probabilística M con oráculo tal que $t_M(n)$ es $O(n^k)$ y para cada

$$H \subseteq \{\psi \mid \psi \text{ es una fórmula en CNF tal que } \#\text{CNF-SAT}(\psi) \geq 2\}$$

y cada fórmula φ en CNF:

- Si $\varphi \in$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) \geq \frac{3}{4}$
- Si $\varphi \notin$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) = 0$

La demostración del teorema de Valiant-Vazirani

Para terminar con la demostración necesitamos construir una MT probabilística M con oráculo tal que $t_M(n)$ es $O(n^k)$ y para cada

$$H \ \subseteq \ \{\psi \mid \psi \text{ es una fórmula en CNF tal que } \#\text{CNF-SAT}(\psi) \geq 2\}$$

y cada fórmula φ en CNF:

- Si $\varphi \in$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) \geq \frac{3}{4}$
- Si $\varphi \notin$ CNF-SAT, entonces $\mathbf{Pr}(M^{\text{U-CNF-SAT}_H} \text{ acepte } \varphi) = 0$

Teniendo el lema ya demostrado, usted va a construir esta MT M en la tarea 🙂

Otra propiedad útil de las funciones de hash

Vamos a demostrar una propiedad que será fundamental para la demostración de que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

Otra propiedad útil de las funciones de hash

Vamos a demostrar una propiedad que será fundamental para la demostración de que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

Lema

Sea $X \subseteq \{0,1\}^m$, y suponga que se escoge con distribución uniforme y de manera independiente $n+1$ funciones de hash aleatorias $h_1, \ldots, h_{n+1}$ desde el conjunto $\mathcal{H}(m,n)$.

1. *Si $|X| \leq 2^{n-1}$, entonces:*

$$\mathbf{Pr}(\exists x \in X\, \forall k \in \{1,\ldots,n+1\}\, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \ \leq \ \frac{1}{4}$$

2. *Si $|X| > (n+1)2^n$, entonces:*

$$\mathbf{Pr}(\exists x \in X\, \forall k \in \{1,\ldots,n+1\}\, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \ = \ 1$$

Otra propiedad útil de las funciones de hash

En este lema queremos razonar sobre la siguiente probabilidad:

$$\mathbf{Pr}_{h_1 \sim \mathcal{H}(m,n), \ldots, h_{n+1} \sim \mathcal{H}(m,n)} \big(\\ \exists x \in X \, \forall k \in \{1, \ldots, n+1\} \, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y)) \big)$$

Otra propiedad útil de las funciones de hash

En este lema queremos razonar sobre la siguiente probabilidad:

$$\mathbf{Pr}_{h_1 \sim \mathcal{H}(m,n),\ldots,h_{n+1} \sim \mathcal{H}(m,n)}\big(\\ \exists x \in X \, \forall k \in \{1,\ldots,n+1\} \, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))\big)$$

Para simplificar la notación omitimos los subíndices en las probabilidades

Demostración de la parte 1 del lema

Por los lemas anteriores sabemos que:

$$\mathbf{Pr}(h_k(x) = h_k(y)) \quad = \quad 2^{-n}$$

Demostración de la parte 1 del lema

Por los lemas anteriores sabemos que:

$$\mathbf{Pr}(h_k(x) = h_k(y)) \quad = \quad 2^{-n}$$

Así, dado $x \in X$:

$$\begin{aligned}
\mathbf{Pr}(\exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \quad &= \quad \mathbf{Pr}\Big(\bigvee_{y \in X \,:\, y \neq x} h_k(x) = h_k(y) \Big) \\
&\leq \quad \sum_{y \in X \,:\, y \neq x} \mathbf{Pr}(h_k(x) = h_k(y)) \\
&\leq \quad |X| \cdot 2^{-n} \\
&\leq \quad 2^{n-1} \cdot 2^{-n} \\
&= \quad \frac{1}{2}
\end{aligned}$$

Demostración de la parte 1 del lema

Dado que las funciones $h_1, \ldots, h_{n+1}$ son escogidas de manera independiente, concluimos que:

$$
\begin{aligned}
\mathbf{Pr}(\forall k \in \{1, \ldots, n+1\}\, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \ &= \\
&\prod_{k=1}^{n+1} \mathbf{Pr}(\exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \ \leq \\
&\prod_{k=1}^{n+1} \frac{1}{2} \ \leq \\
&2^{-n-1}
\end{aligned}
$$

Demostración de la parte 1 del lema

Así, finalmente tenemos que:

$$
\begin{aligned}
&\mathbf{Pr}(\exists x \in X\, \forall k \in \{1, \ldots, n+1\}\, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \;=\\
&\qquad \mathbf{Pr}\Big(\bigvee_{x \in X} \forall k \in \{1, \ldots, n+1\}\, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))\Big) \;\leq\\
&\qquad \sum_{x \in X} \mathbf{Pr}(\forall k \in \{1, \ldots, n+1\}\, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \;\leq\\
&\qquad \sum_{x \in X} 2^{-n-1} \;=\\
&\qquad |X| \cdot 2^{-n-1} \;\leq\\
&\qquad 2^{n-1} \cdot 2^{-n-1} \;=\\
&\qquad \frac{1}{4}
\end{aligned}
$$

Demostración de la parte 2 del lema

Definimos una secuencia de conjuntos X_1, ..., X_{n+1} tales $X_i \subseteq X$ para cada $i \in \{1, \ldots, n+1\}$

Demostración de la parte 2 del lema

Definimos una secuencia de conjuntos X_1, ..., X_{n+1} tales $X_i \subseteq X$ para cada $i \in \{1, \ldots, n+1\}$

Primero, X_1 es definido como:

$$X_1 \quad = \quad \{x \in X \mid \exists y \in X : (y \neq x \wedge h_1(x) = h_1(y))\}$$

Demostración de la parte 2 del lema

Definimos una secuencia de conjuntos X_1, ..., X_{n+1} tales $X_i \subseteq X$ para cada $i \in \{1, \ldots, n+1\}$

Primero, X_1 es definido como:

$$X_1 \quad = \quad \{x \in X \mid \exists y \in X : (y \neq x \wedge h_1(x) = h_1(y))\}$$

Dado $i \in \{1, \ldots, n\}$, definimos X_{i+1} como:

$$X_{i+1} \quad = \quad \{x \in X_i \mid \exists y \in X : (y \neq x \wedge h_{i+1}(x) = h_{i+1}(y))\}$$

Demostración de la parte 2 del lema

Si $x \in X_{n+1}$, entonces tenemos que la siguiente condición se satisface:

$$\forall k \in \{1, \ldots, n+1\}\, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))$$

Demostración de la parte 2 del lema

Si $x \in X_{n+1}$, entonces tenemos que la siguiente condición se satisface:

$$\forall k \in \{1, \ldots, n+1\} \, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))$$

Por lo tanto, si demostramos que $|X_{n+1}| > 0$ concluimos que:

$$\mathbf{Pr}(\exists x \in X \, \forall k \in \{1, \ldots, n+1\} \, \exists y \in X : (y \neq x \wedge h_k(x) = h_k(y))) \quad = \quad 1$$

Demostración de la parte 2 del lema

Dado que el conjunto $\{0,1\}^n$ tiene 2^n strings y $h_1 : \{0,1\}^m \to \{0,1\}^n$, si $|X| > 2^n$ entonces existen elementos $x, y \in X$ tales que:

$$x \neq y \quad \text{y} \quad h_1(x) = h_1(y)$$

Demostración de la parte 2 del lema

Dado que el conjunto $\{0,1\}^n$ tiene 2^n strings y $h_1 : \{0,1\}^m \to \{0,1\}^n$, si $|X| > 2^n$ entonces existen elementos $x, y \in X$ tales que:

$$x \neq y \quad \text{y} \quad h_1(x) = h_1(y)$$

Usando este argumento es posible concluir que $|X_1| \geq |X| - 2^n$

- ¿Por qué?

Demostración de la parte 2 del lema

Para cada $i \in \{1, \ldots, n\}$ podemos aplicar el mismo argumento obteniendo:

$$\begin{aligned} |X_{i+1}| &\geq |X_i| - 2^n \\ &\geq |X| - (i+1)2^i \end{aligned}$$

Demostración de la parte 2 del lema

Para cada $i \in \{1, \ldots, n\}$ podemos aplicar el mismo argumento obteniendo:

$$\begin{aligned} |X_{i+1}| &\geq |X_i| - 2^n \\ &\geq |X| - (i+1)2^i \end{aligned}$$

Concluimos que:

$$\begin{aligned} |X_{n+1}| &\geq |X| - (n+1)2^n \\ &> (n+1)2^n - (n+1)2^n \\ &= 0 \end{aligned}$$

Demostración de la parte 2 del lema

Para cada $i \in \{1, \ldots, n\}$ podemos aplicar el mismo argumento obteniendo:

$$\begin{aligned} |X_{i+1}| &\geq |X_i| - 2^n \\ &\geq |X| - (i+1)2^i \end{aligned}$$

Concluimos que:

$$\begin{aligned} |X_{n+1}| &\geq |X| - (n+1)2^n \\ &> (n+1)2^n - (n+1)2^n \\ &= 0 \end{aligned}$$

Esto era lo que teníamos que demostrar $\square$

Volvemos a la demostración de que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

Las funciones de hash aleatorias son el ingrediente necesario para demostrar que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

Volvemos a la demostración de que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

Las funciones de hash aleatorias son el ingrediente necesario para demostrar que $\overline{\text{GRAPH-ISO}} \in \text{AM}$

En el resto de esta presentación nos vamos a dedicar a esta demostración

Un poco de notación para grafos

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

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

Un poco de notación para grafos

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

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

Notación

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

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

Un poco de notación para grafos

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

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

Notación

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

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

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* $f : N \to N$, *decimos que* f *es un automorfismo para* G *si* $f(G) = G$

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

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

Los automorfismos de un grafo

Definición

Dado un grafo $G = (N, A)$ y una biyección $f : N \to N$, 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 $G = (N, A)$:

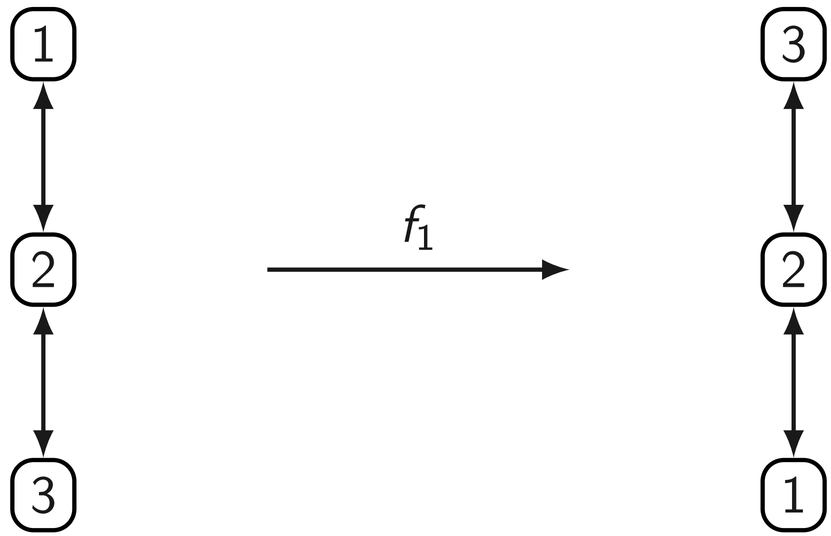
En este caso tenemos que $N = \{1, 2, 3\}$ y $A = \{(1, 2), (2, 1), (2, 3), (3, 2)\}$

Contando el número de grafos isomorfos a un grafo

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

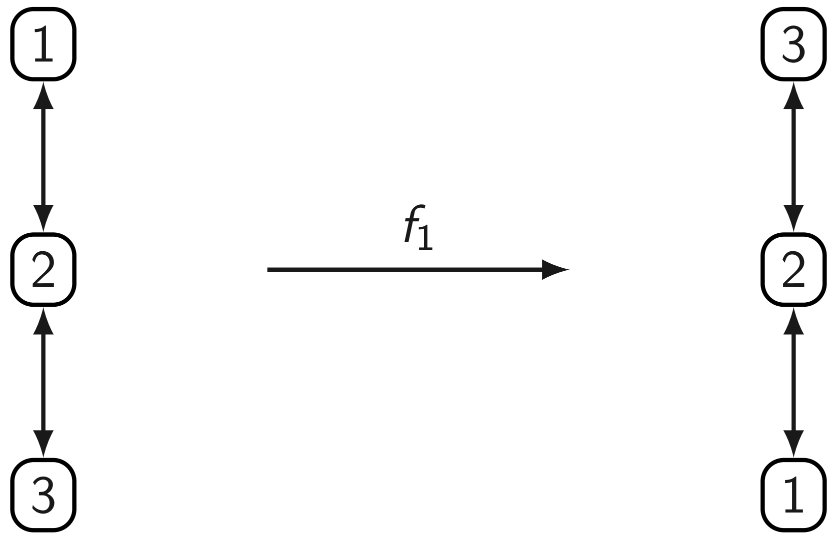
Contando el número de grafos isomorfos a un grafo

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 $f_1(G) = G$

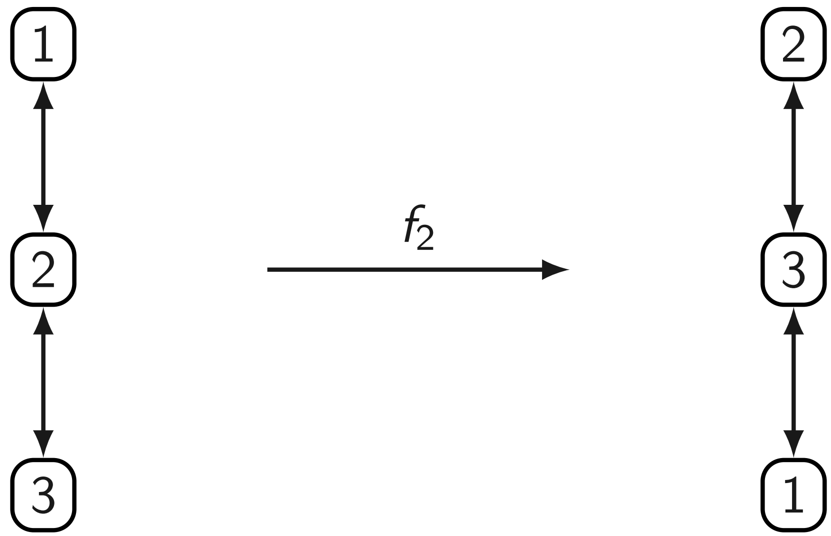
- En particular, si $f_1(G) = (N, A')$ entonces $A = A'$

Contando el número de grafos isomorfos a un grafo

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

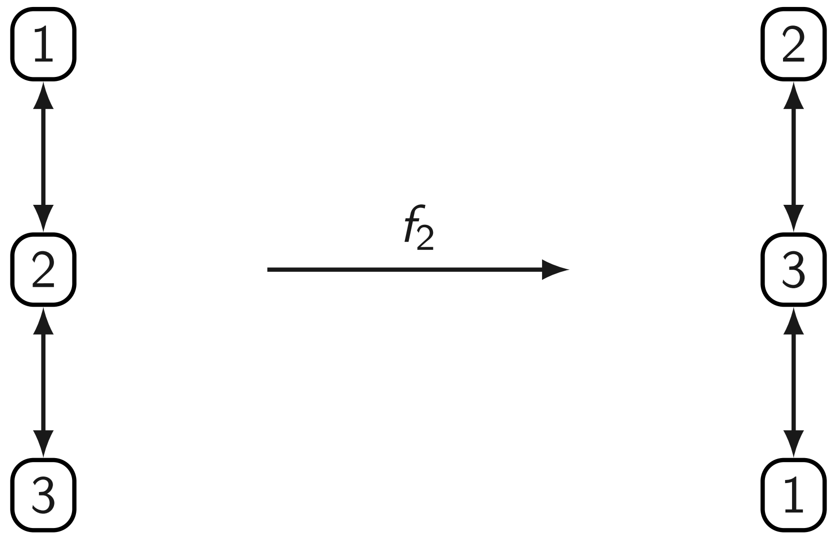
Contando el número de grafos isomorfos a un grafo

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 $f_2(3) = 1$:

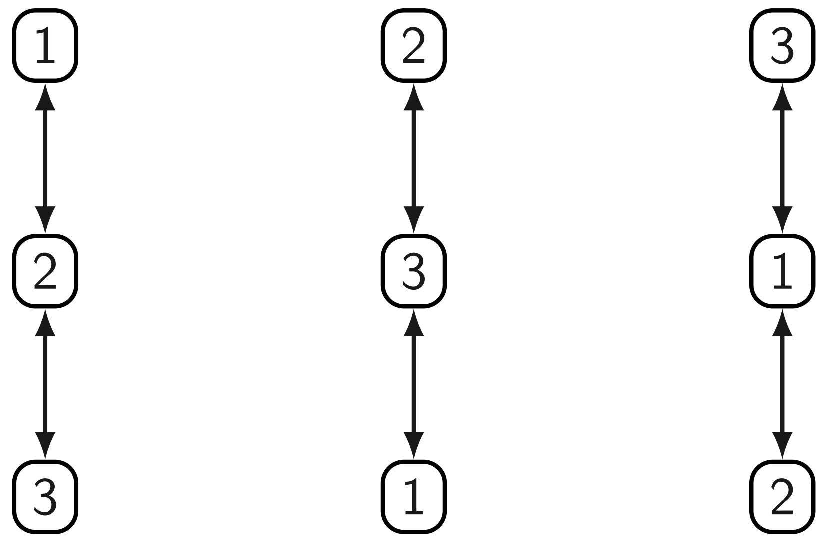


f_2 no es un automorfismo para G ya que $f_2(G) \neq G$

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

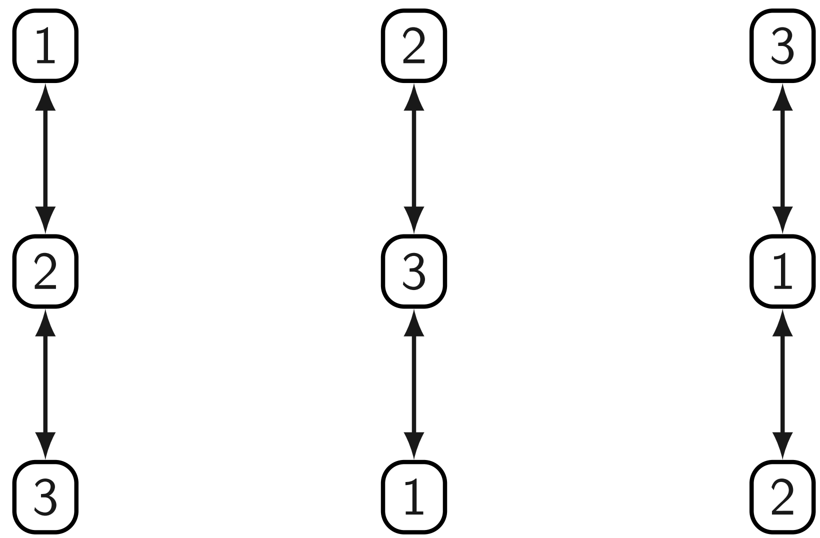
Contando el número de grafos isomorfos a un grafo

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



Contando el número de grafos isomorfos a un grafo

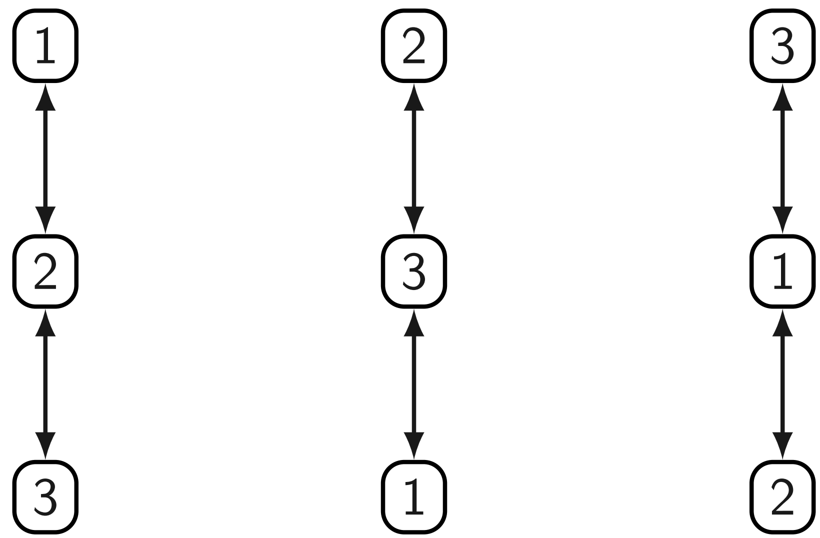
Para el caso de G tenemos seis biyecciones posibles que generan tres grafos distintos:



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 $1, \ldots, n$

El número de grafos isomorfos a un grafo

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

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 $1, \ldots, n$

Lema

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

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

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

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 $\{1, \ldots, n\}$: $[\mathrm{id}]_\sim = \mathrm{Aut}(G)$
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 $\{1, \ldots, n\}$: $[\mathrm{id}]_\sim = \mathrm{Aut}(G)$
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 $\frac{n!}{|\mathrm{Aut}(G)|}$, que es lo que teníamos que demostrar.

- ¿Por qué?

Demostración del lema

En primer lugar tenemos que:

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

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 $\mathcal{T} : [f_1]_\sim \to [f_2]_\sim$

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 $\mathcal{T} : [f_1]_\sim \to [f_2]_\sim$

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

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

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:

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

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:

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

Demostración del lema

Además, tenemos que:

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

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

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:

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

Definiendo un testigo (probabilístico) para grafos no isomorfos

Ejemplo

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

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

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 $\text{num}(G_1, G_2)$:

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

$\text{num}(G_1, G_2)$ 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 $\text{num}(G_1, G_2)$:

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

$\text{num}(G_1, G_2)$ 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 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 $|num(G_1, G_2)| = 2 \cdot n!$, si no se tiene que $|num(G_1, G_2)| \geq 4 \cdot n!$

El conjunto 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 $|num(G_1, G_2)| = 2 \cdot n!$, si no se tiene que $|num(G_1, G_2)| \geq 4 \cdot n!$

¿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 $\frac{n!}{|\mathrm{Aut}(G)|}$

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:

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

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:

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

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:

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

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:

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

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

Tenemos los ingredientes necesarios para la demostración

Teorema (Schöning)

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

Tenemos los ingredientes necesarios para la demostración

Teorema (Schöning)

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

Corolario

$\textit{GRAPH-ISO} \in \textit{co-AM}$

Demostración del teorema: notación inicial

Antes de definir el protocolo, vamos a discutir algunas nociones y herramientas útiles para la demostración

Demostración del teorema: notación inicial

Antes de definir el protocolo, vamos a discutir algunas nociones y herramientas útiles para la demostración

Suponga que G_1 y G_2 son dos grafos con $m > 0$ nodos cada uno

Demostración del teorema: notación inicial

Antes de definir el protocolo, vamos a discutir algunas nociones y herramientas útiles para la demostración

Suponga que G_1 y G_2 son dos grafos con $m > 0$ nodos cada uno

- Recuerde que $\text{num}(G_1, G_2)$ fue definido como:

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

Demostración del teorema: notación inicial

Antes de definir el protocolo, vamos a discutir algunas nociones y herramientas útiles para la demostración

Suponga que G_1 y G_2 son dos grafos con $m > 0$ nodos cada uno

- Recuerde que $\text{num}(G_1, G_2)$ fue definido como:

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

Para utilizar las herramientas desarrolladas primero tenemos que representar cada $(H, i, f) \in \text{num}(G_1, G_2)$ como un string en $\{0, 1\}^\ell$

Demostración del teorema: notación inicial

Antes de definir el protocolo, vamos a discutir algunas nociones y herramientas útiles para la demostración

Suponga que G_1 y G_2 son dos grafos con $m > 0$ nodos cada uno

- Recuerde que $\text{num}(G_1, G_2)$ fue definido como:

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

Para utilizar las herramientas desarrolladas primero tenemos que representar cada $(H, i, f) \in \text{num}(G_1, G_2)$ como un string en $\{0, 1\}^\ell$

- ¿Cuál es el valor de ℓ?

Demostración del teorema: notación inicial

¿Cuántos bits necesitamos para representar una tupla $(H, i, f) \in \mathrm{num}(G_1, G_2)$?

Demostración del teorema: notación inicial

¿Cuántos bits necesitamos para representar una tupla $(H, i, f) \in \text{num}(G_1, G_2)$?

- Podemos representar H usando su matriz de adyacencia, para lo cual necesitamos m^2 bits

Demostración del teorema: notación inicial

¿Cuántos bits necesitamos para representar una tupla $(H, i, f) \in \text{num}(G_1, G_2)$?

- Podemos representar H usando su matriz de adyacencia, para lo cual necesitamos m^2 bits
- Para almacenar el valor de i necesitamos un bit

Demostración del teorema: notación inicial

¿Cuántos bits necesitamos para representar una tupla $(H, i, f) \in \text{num}(G_1, G_2)$?

- Podemos representar H usando su matriz de adyacencia, para lo cual necesitamos m^2 bits
- Para almacenar el valor de i necesitamos un bit
- Podemos almacenar la biyección f como una lista de m números $a_1 \ldots a_m$ tal que $f(i) = a_i$

Demostración del teorema: notación inicial

¿Cuántos bits necesitamos para representar una tupla $(H, i, f) \in \text{num}(G_1, G_2)$?

- Podemos representar H usando su matriz de adyacencia, para lo cual necesitamos m^2 bits
- Para almacenar el valor de i necesitamos un bit
- Podemos almacenar la biyección f como una lista de m números $a_1 \ldots a_m$ tal que $f(i) = a_i$
  - Dado que cada $a_i \leq m$, basta con utilizar $1 + \lfloor \log_2(m) \rfloor$ bits para almacenar a_i

Demostración del teorema: notación inicial

¿Cuántos bits necesitamos para representar una tupla $(H, i, f) \in \text{num}(G_1, G_2)$?

- Podemos representar H usando su matriz de adyacencia, para lo cual necesitamos m^2 bits
- Para almacenar el valor de i necesitamos un bit
- Podemos almacenar la biyección f como una lista de m números $a_1 \ldots a_m$ tal que $f(i) = a_i$
  - Dado que cada $a_i \leq m$, basta con utilizar $1 + \lfloor \log_2(m) \rfloor$ bits para almacenar a_i
  - Por lo tanto necesitamos $m(1 + \lfloor \log_2(m) \rfloor)$ bits para almacenar la lista $a_1 \ldots a_m$

Demostración del teorema: notación inicial

¿Cuántos bits necesitamos para representar una tupla $(H, i, f) \in \text{num}(G_1, G_2)$?

- Podemos representar H usando su matriz de adyacencia, para lo cual necesitamos m^2 bits
- Para almacenar el valor de i necesitamos un bit
- Podemos almacenar la biyección f como una lista de m números $a_1 \ldots a_m$ tal que $f(i) = a_i$
  - Dado que cada $a_i \leq m$, basta con utilizar $1 + \lfloor \log_2(m) \rfloor$ bits para almacenar a_i
  - Por lo tanto necesitamos $m(1 + \lfloor \log_2(m) \rfloor)$ bits para almacenar la lista $a_1 \ldots a_m$

Suponemos entonces que $\ell = m^2 + 1 + m(1 + \lfloor \log_2(m) \rfloor)$

Demostración del teorema: notación inicial

Desde ahora en adelante consideramos a cada elemento de $\text{num}(G_1, G_2)$ como un string de ℓ bits

- Tenemos que $\text{num}(G_1, G_2) \subseteq \{0,1\}^\ell$

Demostración del teorema: notación inicial

Desde ahora en adelante consideramos a cada elemento de $\text{num}(G_1, G_2)$ como un string de ℓ bits

- Tenemos que $\text{num}(G_1, G_2) \subseteq \{0,1\}^\ell$

Defina $X(G_1, G_2)$ como $\text{num}(G_1, G_2)^m$

- Cada elemento de $\text{num}(G_1, G_2)^m$ es de la forma $w_1 w_2 \cdots w_m$, donde para cada $i \in \{1, \ldots, m\}$ se tiene que w_i es un string en $\text{num}(G_1, G_2)$

Demostración del teorema: notación inicial

Desde ahora en adelante consideramos a cada elemento de $\text{num}(G_1, G_2)$ como un string de ℓ bits

- Tenemos que $\text{num}(G_1, G_2) \subseteq \{0,1\}^\ell$

Defina $X(G_1, G_2)$ como $\text{num}(G_1, G_2)^m$

- Cada elemento de $\text{num}(G_1, G_2)^m$ es de la forma $w_1 w_2 \cdots w_m$, donde para cada $i \in \{1, \ldots, m\}$ se tiene que w_i es un string en $\text{num}(G_1, G_2)$

Tenemos que:

Demostración del teorema: notación inicial

Desde ahora en adelante consideramos a cada elemento de $\text{num}(G_1, G_2)$ como un string de ℓ bits

- Tenemos que $\text{num}(G_1, G_2) \subseteq \{0, 1\}^\ell$

Defina $X(G_1, G_2)$ como $\text{num}(G_1, G_2)^m$

- Cada elemento de $\text{num}(G_1, G_2)^m$ es de la forma $w_1 w_2 \cdots w_m$, donde para cada $i \in \{1, \ldots, m\}$ se tiene que w_i es un string en $\text{num}(G_1, G_2)$

Tenemos que:

- $X(G_1, G_2) \subseteq \{0, 1\}^{\ell \cdot m}$

Demostración del teorema: notación inicial

Desde ahora en adelante consideramos a cada elemento de $\text{num}(G_1, G_2)$ como un string de ℓ bits

- Tenemos que $\text{num}(G_1, G_2) \subseteq \{0, 1\}^\ell$

Defina $X(G_1, G_2)$ como $\text{num}(G_1, G_2)^m$

- Cada elemento de $\text{num}(G_1, G_2)^m$ es de la forma $w_1 w_2 \cdots w_m$, donde para cada $i \in \{1, \ldots, m\}$ se tiene que w_i es un string en $\text{num}(G_1, G_2)$

Tenemos que:

- $X(G_1, G_2) \subseteq \{0, 1\}^{\ell \cdot m}$
- Si G_1 no es isomorfo a G_2, entonces $|X(G_1, G_2)| \geq (4 \cdot m!)^m$

Demostración del teorema: notación inicial

Desde ahora en adelante consideramos a cada elemento de $\text{num}(G_1, G_2)$ como un string de ℓ bits

- Tenemos que $\text{num}(G_1, G_2) \subseteq \{0, 1\}^\ell$

Defina $X(G_1, G_2)$ como $\text{num}(G_1, G_2)^m$

- Cada elemento de $\text{num}(G_1, G_2)^m$ es de la forma $w_1 w_2 \cdots w_m$, donde para cada $i \in \{1, \ldots, m\}$ se tiene que w_i es un string en $\text{num}(G_1, G_2)$

Tenemos que:

- $X(G_1, G_2) \subseteq \{0, 1\}^{\ell \cdot m}$
- Si G_1 no es isomorfo a G_2, entonces $|X(G_1, G_2)| \geq (4 \cdot m!)^m$
- Si G_1 es isomorfo a G_2, entonces $|X(G_1, G_2)| = (2 \cdot m!)^m$

Demostración del teorema: notación inicial

Finalmente defina $n = 1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil$

Demostración del teorema: notación inicial

Finalmente defina $n = 1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil$

Tenemos que:

$$\begin{aligned} 1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil &= 1 + \lceil m \cdot (1 + \log_2(m!)) \rceil \\ &\leq 1 + \lceil m \cdot (1 + \log_2(m^m)) \rceil \\ &= 1 + \lceil m \cdot (1 + m \log_2(m)) \rceil \\ &\leq 1 + \lceil m \cdot (1 + m^2) \rceil \\ &= 1 + m + m^3 \end{aligned}$$

Demostración del teorema: notación inicial

Finalmente defina $n = 1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil$

Tenemos que:

$$
\begin{aligned}
1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil \quad &= \quad 1 + \lceil m \cdot (1 + \log_2(m!)) \rceil \\
&\leq \quad 1 + \lceil m \cdot (1 + \log_2(m^m)) \rceil \\
&= \quad 1 + \lceil m \cdot (1 + m \log_2(m)) \rceil \\
&\leq \quad 1 + \lceil m \cdot (1 + m^2) \rceil \\
&= \quad 1 + m + m^3
\end{aligned}
$$

Concluimos que $n + 1 = 2 + \lceil m \cdot \log_2(2 \cdot m!) \rceil < 2^{m-2}$ para todo $m \geq 14$

- Vamos a utilizar esta propiedad en las siguientes láminas

Demostración del teorema: notación inicial

Suponga que G_1 no es isomorfo a G_2 y que $m \geq 14$

- Tenemos que $2^{m-2} > (n+1)$

Demostración del teorema: notación inicial

Suponga que G_1 no es isomorfo a G_2 y que $m \geq 14$

- Tenemos que $2^{m-2} > (n+1)$

Concluimos que $|X(G_1, G_2)| > (n+1)2^n$, puesto que:

$$
\begin{aligned}
|X(G_1, G_2)| &\geq (4 \cdot m!)^m \\
&= 2^{\log_2((4 \cdot m!)^m)} \\
&= 2^{m \cdot \log_2(4 \cdot m!)} \\
&= 2^{m + m \cdot \log_2(2 \cdot m!)} \\
&= 2^{m-1+(1+m \cdot \log_2(2 \cdot m!))} \\
&\geq 2^{m-1+\lceil m \cdot \log_2(2 \cdot m!) \rceil} \\
&= 2^{m-2+n} \\
&= 2^{m-2} \cdot 2^n \\
&> (n+1)2^n
\end{aligned}
$$

Demostración del teorema: notación inicial

Si G_1 es isomorfo a G_2 tenemos que $|X(G_1, G_2)| \leq 2^{n-1}$, puesto que:

$$\begin{array}{rcl}
|X(G_1, G_2)| & = & (2 \cdot m!)^m \\
 & = & 2^{\log_2((2 \cdot m!)^m)} \\
 & = & 2^{m \cdot \log_2(2 \cdot m!)} \\
 & \leq & 2^{\lceil m \cdot \log_2(2 \cdot m!) \rceil} \\
 & = & 2^{1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil - 1} \\
 & = & 2^{n-1}
\end{array}$$

Demostración del teorema: notación inicial

En la demostración vamos a considerar funciones de hash aleatorias $h \in \mathcal{H}(\ell \cdot m, n)$

Demostración del teorema: notación inicial

En la demostración vamos a considerar funciones de hash aleatorias $h \in \mathcal{H}(\ell \cdot m, n)$

Estas funciones están dadas por matrices Booleanas A de $n \times (\ell \cdot m)$

Demostración del teorema: notación inicial

En la demostración vamos a considerar funciones de hash aleatorias $h \in \mathcal{H}(\ell \cdot m, n)$

Estas funciones están dadas por matrices Booleanas A de $n \times (\ell \cdot m)$

- Los elementos son de A son escogidos con distribución uniforme y de manera independiente

Demostración del teorema: notación inicial

En la demostración vamos a considerar funciones de hash aleatorias $h \in \mathcal{H}(\ell \cdot m, n)$

Estas funciones están dadas por matrices Booleanas A de $n \times (\ell \cdot m)$

- Los elementos son de A son escogidos con distribución uniforme y de manera independiente

Necesitamos $(\ell \cdot m \cdot n)$ bits para representar A

Demostración del teorema: notación inicial

Necesitamos entonces $(\ell \cdot m \cdot n)$ bits para representar una función de hash aleatoria $h : \{0,1\}^{\ell \cdot m} \rightarrow \{0,1\}^n$

Demostración del teorema: notación inicial

Necesitamos entonces $(\ell \cdot m \cdot n)$ bits para representar una función de hash aleatoria $h : \{0,1\}^{\ell \cdot m} \to \{0,1\}^n$

Vale decir, necesitamos la siguiente cantidad de bits para representar h:

$$[m^2 + 1 + m(1 + \lfloor \log_2(m) \rfloor)] \cdot m \cdot [1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil]$$

Demostración del teorema: notación inicial

Necesitamos entonces $(\ell \cdot m \cdot n)$ bits para representar una función de hash aleatoria $h : \{0,1\}^{\ell \cdot m} \to \{0,1\}^n$

Vale decir, necesitamos la siguiente cantidad de bits para representar h:

$$[m^2 + 1 + m(1 + \lfloor \log_2(m) \rfloor)] \cdot m \cdot [1 + \lceil m \cdot \log_2(2 \cdot m!) \rceil]$$

El valor $(\ell \cdot m \cdot n)$ es polinomial en m, de lo cual concluimos que es polinomial en el tamaño de (G_1, G_2)

Demostración del teorema: la definición del protocolo

Tenemos que definir un protocolo interactivo con bits aleatorios públicos que recibe como entrada un par de grafos (G_1, G_2), tiene dos rondas, y satisface las siguientes condiciones:

Demostración del teorema: la definición del protocolo

Tenemos que definir un protocolo interactivo con bits aleatorios públicos que recibe como entrada un par de grafos (G_1, G_2), tiene dos rondas, y satisface las siguientes condiciones:

- Si G_1 y G_2 no son isomorfos:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D}) \text{ acepte } (G_1, G_2)) = 1$$

Demostración del teorema: la definición del protocolo

Tenemos que definir un protocolo interactivo con bits aleatorios públicos que recibe como entrada un par de grafos (G_1, G_2), tiene dos rondas, y satisface las siguientes condiciones:

- Si G_1 y G_2 no son isomorfos:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D}) \text{ acepte } (G_1, G_2)) \quad = \quad 1$$

- Si G_1 y G_2 son isomorfos, entonces para todo **D'**:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D'}) \text{ acepte } (G_1, G_2)) \quad \leq \quad \frac{1}{4}$$

Demostración del teorema: la definición del protocolo

Con entrada (G_1, G_2) el protocolo funciona de la siguiente forma:

Demostración del teorema: la definición del protocolo

Con entrada (G_1, G_2) el protocolo funciona de la siguiente forma:

1. **V** revisa si G_1 y G_2 no tienen el mismo número de nodos. Si es así acepta, si no va al paso 2

Demostración del teorema: la definición del protocolo

Con entrada (G_1, G_2) el protocolo funciona de la siguiente forma:

1. **V** revisa si G_1 y G_2 no tienen el mismo número de nodos. Si es así acepta, si no va al paso 2
2. Sea m el número de nodos de G_1 y G_2

Demostración del teorema: la definición del protocolo

Con entrada (G_1, G_2) el protocolo funciona de la siguiente forma:

1. **V** revisa si G_1 y G_2 no tienen el mismo número de nodos. Si es así acepta, si no va al paso 2

2. Sea m el número de nodos de G_1 y G_2

3. Si $m < 14$ entonces **V** va al paso 3.1, si no va al paso 4

Demostración del teorema: la definición del protocolo

Con entrada (G_1, G_2) el protocolo funciona de la siguiente forma:

1. **V** revisa si G_1 y G_2 no tienen el mismo número de nodos. Si es así acepta, si no va al paso 2
2. Sea m el número de nodos de G_1 y G_2
3. Si $m < 14$ entonces **V** va al paso 3.1, si no va al paso 4
   - 3.1 **V** construye todas las posibles biyecciones $f : \{1, \ldots, m\} \to \{1, \ldots, m\}$

Demostración del teorema: la definición del protocolo

Con entrada (G_1, G_2) el protocolo funciona de la siguiente forma:

1. **V** revisa si G_1 y G_2 no tienen el mismo número de nodos. Si es así acepta, si no va al paso 2
2. Sea m el número de nodos de G_1 y G_2
3. Si $m < 14$ entonces **V** va al paso 3.1, si no va al paso 4
   - 3.1 **V** construye todas las posibles biyecciones $f : \{1, \ldots, m\} \rightarrow \{1, \ldots, m\}$
   - 3.2 **V** verifica si alguna de estas biyecciones f es un isomorfismo de G_1 en G_2. Si es así rechaza, si no acepta

Demostración del teorema: la definición del protocolo

Con entrada (G_1, G_2) el protocolo funciona de la siguiente forma:

1. **V** revisa si G_1 y G_2 no tienen el mismo número de nodos. Si es así acepta, si no va al paso 2
2. Sea m el número de nodos de G_1 y G_2
3. Si $m < 14$ entonces **V** va al paso 3.1, si no va al paso 4
   - 3.1 **V** construye todas las posibles biyecciones $f : \{1, \ldots, m\} \to \{1, \ldots, m\}$
   - 3.2 **V** verifica si alguna de estas biyecciones f es un isomorfismo de G_1 en G_2. Si es así rechaza, si no acepta
4. **V** envía a **D** los primeros $\ell \cdot m \cdot n \cdot (n + 1)$ bits de su cinta de bits aleatorios, los cuales representan $n + 1$ funciones h_1, ..., h_{n+1} en $\mathcal{H}(\ell \cdot m, n)$

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

6. Los siguientes pasos se repiten para $k = 0, \ldots, n+1$

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

6. Los siguientes pasos se repiten para $k = 0, \ldots, n+1$

   6.1 Para cada $j \in \{1, \ldots, m\}$, **V** verifica que $g_{k,j}$ es un isomorfismo de $H_{k,j}$ en G_1 o G_2, $i_{k,j} \in \{0, 1\}$ y $f_{k,j} \in \text{Aut}(G_{i_{k,j}})$. Si no es así, entonces rechaza

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

6. Los siguientes pasos se repiten para $k = 0, \ldots, n+1$

   6.1 Para cada $j \in \{1, \ldots, m\}$, **V** verifica que $g_{k,j}$ es un isomorfismo de $H_{k,j}$ en G_1 o G_2, $i_{k,j} \in \{0, 1\}$ y $f_{k,j} \in \mathrm{Aut}(G_{i_{k,j}})$. Si no es así, entonces rechaza

   6.2 Si $k = 0$, entonces define $x = (H_{0,1}, i_{0,1}, f_{0,1}, \ldots, H_{0,m}, i_{0,m}, f_{0,m})$

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

6. Los siguientes pasos se repiten para $k = 0, \ldots, n+1$

   6.1 Para cada $j \in \{1, \ldots, m\}$, **V** verifica que $g_{k,j}$ es un isomorfismo de $H_{k,j}$ en G_1 o G_2, $i_{k,j} \in \{0, 1\}$ y $f_{k,j} \in \text{Aut}(G_{i_{k,j}})$. Si no es así, entonces rechaza

   6.2 Si $k = 0$, entonces define $x = (H_{0,1}, i_{0,1}, f_{0,1}, \ldots, H_{0,m}, i_{0,m}, f_{0,m})$

$\boxed{x \in X(G_1, G_2)}$

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

6. Los siguientes pasos se repiten para $k = 0, \ldots, n+1$

   6.1 Para cada $j \in \{1, \ldots, m\}$, **V** verifica que $g_{k,j}$ es un isomorfismo de $H_{k,j}$ en G_1 o G_2, $i_{k,j} \in \{0, 1\}$ y $f_{k,j} \in \mathrm{Aut}(G_{i_{k,j}})$. Si no es así, entonces rechaza

   6.2 Si $k = 0$, entonces define $x = (H_{0,1}, i_{0,1}, f_{0,1}, \ldots, H_{0,m}, i_{0,m}, f_{0,m})$

   $\boxed{x \in X(G_1, G_2)}$

   6.3 Si $k > 0$, entonces define $y_k = (H_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, i_{k,m}, f_{k,m})$

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

6. Los siguientes pasos se repiten para $k = 0, \ldots, n+1$

   6.1 Para cada $j \in \{1, \ldots, m\}$, **V** verifica que $g_{k,j}$ es un isomorfismo de $H_{k,j}$ en G_1 o G_2, $i_{k,j} \in \{0,1\}$ y $f_{k,j} \in \mathrm{Aut}(G_{i_{k,j}})$. Si no es así, entonces rechaza

   6.2 Si $k = 0$, entonces define $x = (H_{0,1}, i_{0,1}, f_{0,1}, \ldots, H_{0,m}, i_{0,m}, f_{0,m})$

   $\boxed{x \in X(G_1, G_2)}$

   6.3 Si $k > 0$, entonces define $y_k = (H_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, i_{k,m}, f_{k,m})$

   $\boxed{y \in X(G_1, G_2)}$

Demostración del teorema: la definición del protocolo

5. **D** responde a **V** con una secuencia de strings $(H_{k,1}, g_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, g_{k,m}, i_{k,m}, f_{k,m})$ para $k \in \{0, \ldots, n+1\}$

6. Los siguientes pasos se repiten para $k = 0, \ldots, n+1$

   6.1 Para cada $j \in \{1, \ldots, m\}$, **V** verifica que $g_{k,j}$ es un isomorfismo de $H_{k,j}$ en G_1 o G_2, $i_{k,j} \in \{0, 1\}$ y $f_{k,j} \in \text{Aut}(G_{i_{k,j}})$. Si no es así, entonces rechaza

   6.2 Si $k = 0$, entonces define $x = (H_{0,1}, i_{0,1}, f_{0,1}, \ldots, H_{0,m}, i_{0,m}, f_{0,m})$

   $\boxed{x \in X(G_1, G_2)}$

   6.3 Si $k > 0$, entonces define $y_k = (H_{k,1}, i_{k,1}, f_{k,1}, \ldots, H_{k,m}, i_{k,m}, f_{k,m})$

   $\boxed{y \in X(G_1, G_2)}$

7. **V** verifica si $h_k(x) = h_k(y_k)$ para cada $k \in \{1, \ldots, n+1\}$. Si es así, entonces acepta, y si no rechaza

Demostración del teorema: la probabilidad de error

El protocolo utiliza bit aleatorios públicos y tiene dos rondas

Demostración del teorema: la probabilidad de error

El protocolo utiliza bit aleatorios públicos y tiene dos rondas

Tenemos que determinar ahora la probabilidad de error del protocolo

Demostración del teorema: la probabilidad de error

El protocolo utiliza bit aleatorios públicos y tiene dos rondas

Tenemos que determinar ahora la probabilidad de error del protocolo

Vale decir, dados dos grafos G_1 y G_2, queremos determinar la probabilidad:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D}) \text{ acepte } (G_1, G_2))$$

dependiendo de si G_1 y G_2 son o no son isomorfos

Demostración del teorema: la probabilidad de error

Suponemos primero que G_1 y G_2 **no** son isomorfos

Demostración del teorema: la probabilidad de error

Suponemos primero que G_1 y G_2 **no** son isomorfos

Si G_1 no tiene el mismo número de nodos que G_2 entonces **V** acepta con probabilidad 1 (no invoca a **D**)

Demostración del teorema: la probabilidad de error

Suponemos primero que G_1 y G_2 **no** son isomorfos

Si G_1 no tiene el mismo número de nodos que G_2 entonces **V** acepta con probabilidad 1 (no invoca a **D**)

- Suponemos entonces que G_1 y G_2 tienen el mismo número de nodos m

Demostración del teorema: la probabilidad de error

Suponemos primero que G_1 y G_2 **no** son isomorfos

Si G_1 no tiene el mismo número de nodos que G_2 entonces **V** acepta con probabilidad 1 (no invoca a **D**)

- Suponemos entonces que G_1 y G_2 tienen el mismo número de nodos m

Si $m < 14$, entonces **V** también acepta con probabilidad 1 (no invoca a **D**)

Demostración del teorema: la probabilidad de error

Suponemos primero que G_1 y G_2 **no** son isomorfos

Si G_1 no tiene el mismo número de nodos que G_2 entonces **V** acepta con probabilidad 1 (no invoca a **D**)

- Suponemos entonces que G_1 y G_2 tienen el mismo número de nodos m

Si $m < 14$, entonces **V** también acepta con probabilidad 1 (no invoca a **D**)

- Suponemos entonces que $m \geq 14$

Demostración del teorema: la probabilidad de error

Dado que $m \geq 14$, tenemos que $|X(G_1, G_2)| > (n+1)2^n$

Demostración del teorema: la probabilidad de error

Dado que $m \geq 14$, tenemos que $|X(G_1, G_2)| > (n+1)2^n$

Entonces, por el último lema, sabemos que para cualquier secuencia h_1, $\ldots$, h_{n+1} de funciones de hash aleatorias en $\mathcal{H}(\ell \cdot m, n)$:

$$\exists x \in X(G_1, G_2) \forall k \in \{1, \ldots, n+1\}$$
$$\exists y \in X(G_1, G_2) : (x \neq y \wedge h_k(x) = h_k(y))$$

Demostración del teorema: la probabilidad de error

Dado que $m \geq 14$, tenemos que $|X(G_1, G_2)| > (n+1)2^n$

Entonces, por el último lema, sabemos que para cualquier secuencia h_1, ..., h_{n+1} de funciones de hash aleatorias en $\mathcal{H}(\ell \cdot m, n)$:

$$\exists x \in X(G_1, G_2) \forall k \in \{1, \ldots, n+1\} \\ \exists y \in X(G_1, G_2) : (x \neq y \wedge h_k(x) = h_k(y))$$

Por lo tanto, en este caso existe un demostrador **D** tal que:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D}) \text{ acepte } (G_1, G_2)) = 1$$

Demostración del teorema: la probabilidad de error

Consideramos ahora el caso en que G_1 y G_2 son grafos isomorfos

Demostración del teorema: la probabilidad de error

Consideramos ahora el caso en que G_1 y G_2 son grafos isomorfos

- Suponemos que G_1 y G_2 tienen el mismo número de nodos m

Demostración del teorema: la probabilidad de error

Consideramos ahora el caso en que G_1 y G_2 son grafos isomorfos

- Suponemos que G_1 y G_2 tienen el mismo número de nodos m

Si $m < 14$ entonces $\mathbf{V}$ no se puede equivocar al decidir si G_1 es isomorfo a G_2, y tenemos que para todo demostrador $\mathbf{D}$:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D}) \text{ acepte } (G_1, G_2)) \quad = \quad 0$$

Demostración del teorema: la probabilidad de error

Consideramos ahora el caso en que G_1 y G_2 son grafos isomorfos

- Suponemos que G_1 y G_2 tienen el mismo número de nodos m

Si $m < 14$ entonces **V** no se puede equivocar al decidir si G_1 es isomorfo a G_2, y tenemos que para todo demostrador **D**:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D}) \text{ acepte } (G_1, G_2)) \quad = \quad 0$$

Suponemos entonces que $m \geq 14$

Demostración del teorema: la probabilidad de error

En este caso tenemos que $|X(G_1, G_2)| \leq 2^{n-1}$

Demostración del teorema: la probabilidad de error

En este caso tenemos que $|X(G_1, G_2)| \leq 2^{n-1}$

Concluimos por el último lema que:

$$\mathbf{Pr}(\exists x \in X(G_1, G_2)\, \forall k \in \{1, \ldots, n+1\}$$
$$\exists y \in X(G_1, G_2) : (x \neq y \wedge h_k(x) = h_k(y))) \ \leq \ \frac{1}{4}$$

Demostración del teorema: la probabilidad de error

En este caso tenemos que $|X(G_1, G_2)| \leq 2^{n-1}$

Concluimos por el último lema que:

$$\mathbf{Pr}(\exists x \in X(G_1, G_2)\, \forall k \in \{1, \ldots, n+1\}$$
$$\exists y \in X(G_1, G_2) : (x \neq y \wedge h_k(x) = h_k(y))) \ \leq \ \frac{1}{4}$$

Por lo tanto, para todo demostrador **D**:

$$\mathbf{Pr}((\mathbf{V}, \mathbf{D}) \text{ acepte } (G_1, G_2)) \quad \leq \quad \frac{1}{4}$$

$\square$