Sistemas de demostración interactivos

Marcos Kiwi

U. Chile

Semestre Otoño 2012

Interacción determinista

Definición (Interacción entre funciones deterministas)

Sean $V, P : \{0,1\}^* \to \{0,1\}^*$ *y* $k : \mathbb{N} \to \mathbb{N}$. *Una interacción de* $k(n)$*-etapas entre* V *y* P *en la entrada* $\omega \in \{0,1\}^*$, *denotada* $(V \leftrightarrow P)(\omega)$, *es una secuencia* $q_1, a_1, \ldots, q_k, a_k \in \{0,1\}^*$, $k = k(|\omega|)$, *tal que*

$$
\begin{aligned}
q_1 &= V(\omega)\,, \\
a_1 &= P(\omega, q_1)\,, \\
&\ldots \\
q_{i+1} &= V(\omega, q_1, a_1, \ldots, q_i, a_i)\,, \text{ para } i < k, \\
a_{i+1} &= P(\omega, q_1, a_1, \ldots, a_i, q_{i+1})\,, \text{ para } i < k.
\end{aligned}
$$

Se define el resultado (o salida) de la interacción, también denotado $(V \leftrightarrow P)(\omega)$, *como* $V(\omega, q_1, a_1, \ldots, q_k, a_k)$ *la que se asume igual a* 1 *(que se interpreta como* acep*), y en caso contrario se interpreta como* rech.

Interacción determinista (cont.)

Definición (Sistemas interactivos deterministas de demostración)

Decimos que un lenguaje L tiene un sistema interactivo de demostración determinista si existe una máquina de Turing V que calcula una función de $\{0,1\}^*$ en $\{0,1\}^*$, que abusando notación también denotamos por V, que al tener una interacción de $k = k(n)$ etapas con cualquier función $P : \{0,1\}^* \to \{0,1\}^*$ en una entrada $\omega \in \{0,1\}^*$, satisface:

(Completitud) $\omega \in L \Longrightarrow \exists P^* : \{0,1\}^* \to \{0,1\}^*, (V \leftrightarrow P^*)(\omega) = 1,$
(Consistencia) $\omega \notin L \Longrightarrow \forall P : \{0,1\}^* \to \{0,1\}^*, (V \leftrightarrow P)(\omega) = 0.$

Además, en la entrada $(\omega, q_1, a_1, \ldots, q_i, a_i)$, $0 \leq i \leq k(n)$, la evaluación de V toma tiempo polinomial en $|\omega|$.

Comentario

Observación

La P es por "probador" (prover) y la V es por "verificador" (verifier).

Clase dIP

Definición

Se define la clase dIP como el conjunto de lenguajes L para los que existe un sistema interactivo de demostración de $k(n)$-etapas en que $k(\cdot)$ es un polinomio.

Teorema

$\mathrm{dIP} = \mathrm{NP}$.

Interacción probabilista

(con lanzamientos de monedas privados)

Definición (Interacción probabilista)

Sean $V, P : \{0,1\}^* \to \{0,1\}^*$ *y* $k : \mathbb{N} \to \mathbb{N}$. *Una interacción probabilista de* $k(n)$*-etapas entre* V *y* P *en la entrada* $\omega \in \{0,1\}^*$ *y la secuencia aleatoria* $\rho \in \{0,1\}^{r(|\omega|)}$, *denotada* $(V \leftrightarrow P)(\omega, \rho)$, *es una secuencia* $q_1, a_1, \ldots, q_k, a_k \in \{0,1\}^*$, $k = k(|\omega|)$, *tal que*

$$
\begin{aligned}
q_1 &= V(\omega, \rho)\,, \\
a_1 &= P(\omega, q_1)\,, \\
&\ldots \\
q_{i+1} &= V(\omega, \rho, q_1, a_1, \ldots, q_i, a_i)\,, \quad \text{para } i < k, \\
a_{i+1} &= P(\omega, q_1, a_1, \ldots, a_i, q_{i+1})\,, \quad \text{para } i < k.
\end{aligned}
$$

Se define el resultado (o salida) de la interacción, también denotado $(V \leftrightarrow P)(\omega, \rho)$, *como* $V(\omega, \rho, q_1, a_1, \ldots, q_k, a_k)$ *la que se asume igual a* 1 *(que se interpreta como* acep*), y en caso contrario se interpreta como* rech. *Para* ρ *escogido al azar uniformemente en* $\{0,1\}^{r(|\omega|)}$ *se tiene que* $(V \leftrightarrow P)(\omega, \rho)$ *es una variable aleatoria.*

Interacción probabilista

(con lanzamientos de monedas públicos)

Se definen de manera análoga a las interacciones probabilistas con lanzamientos de monedas privadas salvo porque ahora $r = \rho_1 || \ldots || \rho_k$,

$$
\begin{aligned}
q_1 &= \rho_1 \,, \\
a_1 &= P(\omega, q_1) \,, \\
&\ldots \\
q_{i+1} &= \rho_{i+1} \,, \text{ para } i < k, \\
a_{i+1} &= P(\omega, q_1, a_1, \ldots, a_i, q_{i+1}) \,, \text{ para } i < k.
\end{aligned}
$$

Interacción probabilista (cont.)

Definición (Sistemas interactivos de demostración)

Decimos que un lenguaje L está en $\mathrm{IP}[k]$ si existe una máquina de Turing V que calcula una función de $\{0,1\}^*$ en $\{0,1\}^*$, que abusando notación también denotamos por V, que al tener una interacción de k-etapas con cualquier función $P : \{0,1\}^* \to \{0,1\}^*$ en una entrada $\omega \in \{0,1\}^*$, satisface:

(Completitud)
$\omega \in L \Longrightarrow \exists P^* : \{0,1\}^* \to \{0,1\}^* , \mathbb{P}_\rho \left((V \leftrightarrow P^*)(\omega, \rho) = 1\right) \geq 2/3,$
(Consistencia)
$\omega \notin L \Longrightarrow \forall P : \{0,1\}^* \to \{0,1\}^* , \mathbb{P}_\rho \left((V \leftrightarrow P)(\omega, \rho) = 1\right) \leq 1/3.$

Además, en la entrada $(\omega, \rho, q_1, a_1, \ldots, q_i, a_i)$, $0 \leq i \leq k$, la evaluación de V toma tiempo polinomial en $|\omega|$.

IP y AM

Definición (IP)

Se define $\mathrm{IP} = \cup_{c \in \mathbb{N}} \mathrm{IP}[n^c]$.

Observación (IP_{pub})

Si en vez de interacción probabilista con lanzamientos de monedas privados se consideran interacciones probabilistas con lanzamientos de monedas públicos se obtienen, de manera análoga, las clases $\mathrm{AM}[2k]$ *y* AM.

Observaciones

Lema

$\mathrm{NP} \subseteq \mathrm{AM}[2]$, $\mathrm{AM}[2k] \subseteq \mathrm{IP}[k]$, *y* $\mathrm{AM} \subseteq \mathrm{IP}$.

Lema

Si $A \leq_P B$ *y* $B \in \mathrm{IP}[k]$ *(respectivamente* $B \in \mathrm{AM}[k]$*), entonces* $A \in \mathrm{IP}[k]$ *(respectivamente* $A \in \mathrm{AM}[k]$*).*

Lema (de amplifiación para IP y AM)

El $2/3$ *y* $1/3$ *en la definición de* IP *(respectivamente* AM*) pueden sustituirse por* ρ_{acep} *y* ρ_{rech} *siempre que*

$$1 - \frac{1}{2^{O(n)}} \geq \rho_{acep} \geq \rho_{rech} + \frac{1}{n^{O(1)}} ,$$

y

$$\rho_{rech} \geq \frac{1}{2^{O(n)}} .$$

$\mathrm{IP} = \mathrm{PSPACE}$

Teorema (Shamir (FOCS'90))

$\mathrm{AM} = \mathrm{IP} = \mathrm{PSPACE}$ *(inclusive si el* $2/3$ *en la definición de* IP *se reemplaza por* 1*).*

Grafos no-isomorfos

Se dice que dos grafos G y H son isomorfos si existe $\varphi : V(G) \to V(H)$ biyección tal que

$$uv \in E(G) \iff \varphi(u)\varphi(v) \in E(H)\,.$$

Se define

$$\overline{\mathrm{GISO}} = \{\langle G_0, G_1\rangle : G_0 \text{ y } G_1 \text{ no son isomorfos}\}\,.$$

Teorema

$\overline{\mathrm{GISO}} \in \mathrm{IP}$.

Sistema de demostración interactivo para $\overline{\mathrm{GISO}}$

Dados G_0 y G_1 tales que (sin pérdida de generalidad) $V(G_0) = V(G_1) = [n]$,

V : $b \in_r \{0,1\}, \pi \in_r S_n$, y

$H = \pi(G_b) = ([n], E)$ donde $E = \{\pi(u)\pi(v) : uv \in E(G_b)\}$,

$V \to P$: H,

$V \leftarrow P$: $\widetilde{b} \in \{0,1\}$, supuestamente tal que $H = \pi(G_{\widetilde{b}})$,

V : *Aceptar* si y sólo si $\widetilde{b} = b$.

Observaciones (cont.)

Observación

Asociado a un sistema de demostración interactivo para un lenguaje L en IP (o AM) podemos suponer que existen polinomios $q(\cdot)$, $a(\cdot)$, $k(\cdot)$, y $r(\cdot)$ que acotan el largo de las consultas (queries), respuestas (answers), etapas (rounds), y largo de la secuencia aleatoria, respectivamente.

IP $\subseteq$ PSPACE

Sea $L \in \mathrm{IP}$ y V como en la definición de sistema interactivo de demostración para L. Sea $p(\cdot)$ el polinomio que acota el tiempo de ejecución de V. Sean $q(\cdot)$, $a(\cdot)$, $k(\cdot)$, y $r(\cdot)$ los polinomios asociados al sistema interactivo de demostración para L que acotan el largo de las consultas, respuestas, etapas y largo de la secuencia aleatoria, respectivamente.

Observar que existe una máquina de Turing que en $\omega \in \{0,1\}^n$, $q_1, \ldots, q_k \in \{0,1\}^{q(n)}$, $a_1, \ldots, a_k \in \{0,1\}^{a(n)}$, $q \in \{0,1\}^{q(n)}$, e $1 \leq i < k(n)$, puede calcular en espacio polinomial en n el valor de

$$P(\omega, q_1, a_1, \ldots, q_i, a_i, q) = \mathbb{P}\left(V(\omega, \rho, q_1, \ldots, a_i) = q \mid V(\omega, \rho, q_1, \ldots, a_j) = q_{j+1}, 0 \leq j < i\right),$$

donde la probabilidad está tomada sobre los $\rho \in \{0,1\}^{r(n)}$.

Diseñamos un algoritmo recursivo $\mathcal{A}$ que en la entrada $(\omega, (), 0)$ retorna la máxima (sobre las estrategias del probador) probabilidad que el sistema de demostración interactivo para L acepte ω.

De hecho $\mathcal{A}(\omega, (t_1, \ldots, t_i), i)$ será igual a la máxima (sobre las estrategias del probador) probabilidad que V acepte ω condicionado a que las primeras i preguntas/respuestas intercambiadas entre el verificador y el probador están dadas por $t_1, \ldots, t_i$.

Algorithm 1: Algoritmo $\mathcal{A}$ para el cálculo de la probabilidad de aceptación.

input : $\omega \in \{0,1\}^n$, $(t_1, \ldots, t_i)$ tal que $t_i \in \{0,1\}^{q(n)}$ si i es impar y $t_i \in \{0,1\}^{a(n)}$ si i es par, $i \in \mathbb{N}$.

output: Máxima probabilidad que V acepte ω dada la interacción $(t_1, \ldots, t_i)$.

```
/* Fondo de la recursión                                                    */
```

if $i = 2k(n)$ **then**

  return($\mathbb{P}\left(V(\omega, \rho, t_1, \ldots, t_{2k(n)}) = 1 \;\middle|\; V(\omega, \rho, t_1, \ldots, t_{2j}) = t_{2j+1}, 0 \leq j < k(n)\right)$)

```
/* Recursión                                                                */
```

if $i = 0 \pmod 2$ **then**

```
  /* Determinación de la probabilidad esperada que el verificador acepte   */
```

  $S \leftarrow 0$;

  for $q \in \{0,1\}^{q(n)}$ **do**

    $S \leftarrow S + P(\omega, t_1, \ldots, t_i, q) \cdot \mathcal{A}(\omega, (t_1, \ldots, t_i, q), i{+}1)$;

  return(S);

else

```
  /* Determinación de la mejor respuesta del probador                       */
```

  $M \leftarrow 0$;

  for $a \in \{0,1\}^{a(n)}$ **do**

    $M \leftarrow \max\{M, \mathcal{A}(\omega, (t_1, \ldots, t_i, a), i{+}1)\}$;

  return(M);

PSPACE $\subseteq$ IP

(preliminares)

Definición (FBC simple)

Es una fórmula Booleana cuantificada en la cual cada ocurrencia de una variable está separada de su punto de cuantificación por a lo más un cuantificador (sobre una única variable Booleana) del tipo universal. Además, los únicos términos negados en la fórmula son las variables y los únicos conectivos lógicos que aparecen en la fórmula son $\vee$ y $\wedge$.

Ejemplo (FBTC simple)

$\exists x_1 \in \{0,1\}, \forall x_2 \in \{0,1\}, \exists x_3 \in \{0,1\}, [(x_1 \vee \overline{x}_2) \wedge \forall x_4 \in \{0,1\}, (\overline{x}_3 \wedge x_4)]$.

Lema

Dada (la codificación de) una FBTC φ, se puede calcular en tiempo polinomial (una codificación de) una FBTC simple equivalente a φ.

Aritmetización de fórmulas Booleanas (definición)

Dado

$$\varphi = \forall x_1 \in \{0,1\}\,, \exists x_2 \in \{0,1\}\,, [(x_1 \wedge x_1) \vee \forall x_3 \in \{0,1\}\,, (\overline{x}_2 \wedge x_3)]\,,$$

su aritmetización es

$$\mathrm{A}\,(\varphi) = \prod_{z_1 \in \{0,1\}} \sum_{z_2 \in \{0,1\}} \left[(z_1 \cdot z_2) + \prod_{z_3 \in \{0,1\}} (1 - z_2) \cdot z_3 \right]$$

En general, la aritmetización de una FBC φ consiste en:

- Reemplazar cada variable Booleana x_i por una variable z_i sobre $\mathbb{Z}$.
- Reemplazar las ocurrencias de $\overline{x}_i$ por $(1 - z_i)$.
- Reemplazar $\wedge$ por multiplicación entera $\cdot$ y $\vee$ por suma entera $+$.
- Reemplazar $\forall x_i \in \{0,1\}$ por $\sum_{z_i \in \{0,1\}}$ y $\forall x_i \in \{0,1\}$ por $\prod_{z_i \in \{0,1\}}$.

Propiedades de la aritmetización

Lema

Sea φ una FBTC. Se tiene que $\mathrm{A}(\varphi) \in \mathbb{Z}$. Además, φ es verdadera si y sólo si $\mathrm{A}(\varphi) \neq 0$.

Lema

Sea φ una FBC simple con variables libres $x_1, \ldots, x_s$. Se tiene que $\mathrm{A}(\varphi)$ es un polinomio en $z_1, \ldots, z_s$ tal que $\mathrm{grd}(\mathrm{A}(\varphi)) \leq |\langle\varphi\rangle|$.

Propiedades de la aritmetización (cont.)

Lema

Sea φ una FBTC. Se tiene que $|\mathrm{A}(\varphi)| \leq 2^{2^{|\langle\varphi\rangle|}}$.

Lema

Para todo $M \neq 0$ tal que $|M| \leq 2^{2^m}$ existe un primo $p = \Theta(2^m)$ tal que $M \neq 0 \pmod{p}$.

Sistema de demostración interactivo para FBTC simples (Parte I)

En (una codificación de) la entrada φ FBTC simple en n variables:

$$
\begin{aligned}
V \leftarrow P \quad &: \quad p \leq O(1)2^{|\langle\varphi\rangle|} \text{ (supuestamente primo),} \\
&\quad \text{y } A_1 \text{ (supuestamente igual a } \mathrm{A}\,(\varphi)\,\mathrm{mód}\; p \neq 0)\,. \\
V \quad &: \quad \text{Verifica que } p \text{ es primo y Rechaza si no lo es}\,. \\
&\quad \text{Verifica que } 0 < A_1 < p \text{ y Rechaza si no lo es}\,.
\end{aligned}
$$

Sist. de demostración interactivo para FBTC simples (Preliminares Parte II)

La parte medular del protocolo tendrá $n-1$ etapas, indexadas por $i = 1, \ldots, n-1$. Al comienzo de la i-ésima etapa:

- Se habrán seleccionado $r_1, \ldots, r_{i-1} \in \mathbb{Z}_p$.
- $\varphi_i = \varphi_i(x_1, \ldots, x_i)$ representará la parte de la fórmula Booleana φ a continuación de su i-ésimo cuantificador $Q_i x_i$.
- A_i supuestamente representará la aritmetización (módulo p) de $Q_i x_i \varphi_i$ evaluada en $z_j = r_j$, $j = 1, \ldots, i-1$.

Sist. de demostración interactivo para FBTC simples (Parte II)

Para $i = 1, \ldots, n-1$

$V \leftarrow P$: $P_i(z_i)$ (supuesta aritmetización de $\varphi_i(x_1, \ldots, x_i)$ evaluada en $z_j = r_j$, $j = 1, \ldots, i-1$).

V : Si $Q_i = \exists$, verifica que $P_i(0) + P_i(1) \equiv_p A_i$.
Si $Q_i = \forall$, verifica que $P_i(0) \cdot P_i(1) \equiv_p A_i$.
Rechazar si la verificación falla.

V : $r_i \in_r \mathbb{Z}_p$,
Determinar $\varphi'_i = \varphi'_i(x_1, \ldots, x_i)$ tal que $\varphi_i = \varphi'_i \star Q_{i+1} x_{i+1} \varphi_{i+1}$ donde $\star \in \{\wedge, \vee\}$.
Calcular $P'_i(r_1, \ldots, r_i)$ aritmetización de $\varphi'_i = \varphi'_i(x_1, \ldots, x_i)$ evaluada en $z_j = r_j$, $j = 1, \ldots, i$.
Aceptar si $P'_i(r_1, \ldots, r_i) \equiv_p 0$, y en caso contrario definir

$$A_{i+1} = \begin{cases} P_i(r_i) - P'_i(r_1, \ldots, r_i) \text{ mód } p, & \text{si } \star = \vee, \text{ o} \\ P_i(r_i)/P'_i(r_1, \ldots, r_i) \text{ mód } p, & \text{si } \star = \wedge. \end{cases}$$

$V \rightarrow P$: r_i .

Sist. de demostración interactivo para FBTC simples (Parte III)

V : Aritmetizar $Q_n x_n \varphi_n(x_1, \ldots, x_n)$, evaluar módulo p en $z_j = r_j$, $j = 1, \ldots, n-1$. Aceptar si el resultado es A_n, y Rechazar en caso contrario.

Ejemplo

Consideremos $\varphi = \forall x_1 \, [\overline{x}_1 \vee \exists x_2 \forall x_3 (x_1 \wedge x_2) \vee x_3]$

- (Parte I) $A_1 = \prod_{z_1 \in \{0,1\}} \left[(1 - z_1) + \sum_{z_2 \in \{0,1\}} \prod_{z_3 \in \{0,1\}} (z_1 \cdot z_2 + z_3) \right] = 2$.
- (Parte II) $Q_1 = \forall$ y $\varphi_1 = \overline{x}_1 \vee \exists x_2 \forall x_3 (x_1 \wedge x_2) \vee x_3$.
- $P_1(z_1) = 1 + z_1^2$.
- Verificación: $P_1(0) \cdot P_1(1) = A_1$.
- $r_1 = 3$.
- $\varphi_1 = \overline{x}_1 \vee \exists x_2 \varphi_2$, luego $\varphi_1' = \overline{x}_1$ y $P_1'(z_1) = 1 - z_1$.
- $A_2 = P_1(r_1) - P_1'(r_1) = 10 - (-2) = 12$.
- $Q_2 = \exists$ y $\varphi_2 = \forall x_3 (x_1 \wedge x_2) \vee x_3$.
- $P_2(z_2) = (3z_2)(3z_2 + 1) = 9z_2^2 + 3z_2$.
- Verificación: $P_2(0) + P_2(1) = 0 + 12 = A_2$.
- $r_2 = 2$.
- $\varphi_2 = 1 \wedge \forall x_3 \varphi_3$, luego $\varphi_2' = 1$ y $P_2'(z_2) = 1$.
- $A_3 = P_2(r_2)/P_2'(r_2) = 42/1 = 42$.
- (Parte III) Aritmetización de $\forall x_3 (x_1 \wedge x_2) \vee x_3$ evaluada en $z_1 = 3$ y $z_2 = 2$ es

$$(3 \cdot 2 + 0) \cdot (3 \cdot 2 + 1) = 6 \cdot 7 = 42 = A_3 \, .$$