Complejidad descriptiva

IIC3263

Lógicas que no pueden ser evaluadas eficientemente

¿Alguna idea?

La lógica de segundo orden: Sintaxis

Dado: Vocabulario $\mathcal{L}$

Definición

La lógica de segundo orden (LSO) sobre $\mathcal{L}$ es definida como la extensión de LPO que incluye las siguientes reglas:

- *Si $t_1, \ldots, t_k$ son $\mathcal{L}$-términos y X es una variable de segundo orden de aridad k (vale decir, una relación con $k \geq 1$ argumentos), entonces $X(t_1, \ldots, t_k)$ es una fórmula en LSO*
- *Si φ es una fórmula en LSO y X es una variable de segundo orden de aridad k, entonces $\exists X \varphi$ y $\forall X \varphi$ son fórmulas en LSO*

La lógica de segundo orden: Semántica

Notación

Dada una estructura $\mathfrak{A}$ con dominio A, una asignación σ es una función que asigna:

- *un valor en A a cada variable x de primer orden:* $\sigma(x) \in A$
- *un subconjunto de A^k a cada variable X de segundo orden con k argumentos:* $\sigma(X) \subseteq A^k$

La lógica de segundo orden: Semántica

Definición

La definición de la semántica de LSO incluye tres casos adicionales. Para una variable de segundo orden X con aridad k:

- $(\mathfrak{A}, \sigma) \models X(t_1, \ldots, t_k)$ *si y sólo si* $(\sigma(t_1), \ldots, \sigma(t_k)) \in \sigma(X)$
- $(\mathfrak{A}, \sigma) \models \exists X \varphi$ *si y sólo si existe* $S \subseteq A^k$ *tal que* $(\mathfrak{A}, \sigma[X/S]) \models \varphi$
- $(\mathfrak{A}, \sigma) \models \forall X \varphi$ *si y sólo si para todo* $S \subseteq A^k$*, se tiene que* $(\mathfrak{A}, \sigma[X/S]) \models \varphi$

La lógica de segundo orden: Ejemplo

¿Qué podemos decir en LSO?

Sea $\mathcal{L} = \{E(\cdot,\cdot)\}$ y $\mathcal{P}$ el conjunto de todas $\mathcal{L}$-estructuras $\mathfrak{A}$ tal que $E^{\mathfrak{A}}$ es una relación de sucesor.

- Vamos a demostrar que esta propiedad es expresable en LSO

La lógica de segundo orden: Ejemplo

¿Qué podemos decir en LSO?

Sea $\mathcal{L} = \{E(\cdot,\cdot)\}$ y $\mathcal{P}$ el conjunto de todas $\mathcal{L}$-estructuras $\mathfrak{A}$ tal que $E^{\mathfrak{A}}$ es una relación de sucesor.

- Vamos a demostrar que esta propiedad es expresable en LSO

Sea:

$$first(x) \quad = \quad \forall y \neg E(y, x)$$

La lógica de segundo orden: Ejemplo

Y sea:

$$conectado(x, y) = \forall P \left[\left(P(x) \wedge \forall u \forall v \left(P(u) \wedge E(u, v) \rightarrow P(v) \right) \right) \rightarrow P(y) \right]$$

La lógica de segundo orden: Ejemplo

Y sea:

$$conectado(x,y) \;=\; \forall P \Big[\Big(P(x) \wedge \forall u \forall v \, (P(u) \wedge E(u,v) \rightarrow P(v)) \Big) \rightarrow P(y) \Big]$$

Entonces la siguiente $\mathcal{L}$-oración Φ en LSO define $\mathcal{P}$:

$$\begin{aligned} \Phi \;= & \forall x \forall y \forall z \Big(E(x,y) \wedge E(x,z) \rightarrow y = z \Big) \wedge \\ & \forall x \forall y \forall z \Big(E(y,x) \wedge E(z,x) \rightarrow y = z \Big) \wedge \\ & \exists x \Big(first(x) \wedge \forall y \, (x \neq y \rightarrow conectado(x,y)) \Big) \end{aligned}$$

Lógica de Segundo Orden Existencial

Dado: Vocabulario $\mathcal{L}$

Definición

El fragmento existencial de la lógica de segundo orden, denotado como $\exists LSO$, es definido como el conjunto de todas las fórmulas en LSO sobre $\mathcal{L}$ de la forma:

$$\exists X_1 \exists X_2 \cdots \exists X_\ell \ \varphi$$

donde φ es una fórmula en LPO sobre el vocabulario $\mathcal{L}' = \mathcal{L} \cup \{X_1, \ldots, X_\ell\}$

Fragmento existencial de LSO: Ejemplo

¿Qué podemos decir en $\exists$LSO?

Sea $\mathcal{L} = \{E(\cdot, \cdot)\}$ y $\mathcal{P}$ el conjunto de todas $\mathcal{L}$-estructuras $\mathfrak{A}$ tal que $E^{\mathfrak{A}}$ es una relación de sucesor.

- ¿Podemos expresar esta propiedad en $\exists$LSO?

Fragmento existencial de LSO: Ejemplo

¿Qué podemos decir en $\exists$LSO?

Sea $\mathcal{L} = \{E(\cdot,\cdot)\}$ y $\mathcal{P}$ el conjunto de todas $\mathcal{L}$-estructuras $\mathfrak{A}$ tal que $E^{\mathfrak{A}}$ es una relación de sucesor.

- ¿Podemos expresar esta propiedad en $\exists$LSO?

La siguiente $\mathcal{L}$-oración Ψ en $\exists$LSO define $\mathcal{P}$:

$$\psi = \exists L \left(\begin{array}{l} \forall x \neg L(x,x) \wedge \\ \forall x \forall y \forall z \left(L(x,y) \wedge L(y,z) \rightarrow L(x,z) \right) \wedge \\ \forall x \forall y \left(L(x,y) \vee x = y \vee L(y,x) \right) \wedge \\ \forall x \forall y \left(E(x,y) \leftrightarrow \left(L(x,y) \wedge \neg \exists z \left(L(x,z) \wedge L(z,y) \right) \right) \right) \end{array} \right)$$

Complejidad de la $\exists$LSO

Teorema

1. *Para cada vocabulario $\mathcal{L}$ y $\mathcal{L}$-oración φ en $\exists$LSO, se tiene que L_φ está en NP*
2. *Existe un vocabulario $\mathcal{L}$ y una $\mathcal{L}$-oración φ en $\exists$LSO tal que L_φ es NP-completo*

Complejidad de la $\exists$LSO

Teorema

1. *Para cada vocabulario $\mathcal{L}$ y $\mathcal{L}$-oración φ en $\exists$LSO, se tiene que L_φ está en NP*
2. *Existe un vocabulario $\mathcal{L}$ y una $\mathcal{L}$-oración φ en $\exists$LSO tal que L_φ es NP-completo*

Ejercicio

Demuestre la primera parte del teorema

Complejidad de la $\exists$LSO

Teorema

1. *Para cada vocabulario $\mathcal{L}$ y $\mathcal{L}$-oración φ en $\exists$LSO, se tiene que L_φ está en NP*
2. *Existe un vocabulario $\mathcal{L}$ y una $\mathcal{L}$-oración φ en $\exists$LSO tal que L_φ es NP-completo*

Ejercicio

Demuestre la primera parte del teorema

Vamos a demostrar la segunda parte del teorema ...

NP-hardness de ∃LSO: Demostración

Sea $\mathcal{L} = \{E(\cdot, \cdot)\}$

- E es utilizado para almacenar grafos

NP-hardness de ∃LSO: Demostración

Sea $\mathcal{L} = \{E(\cdot, \cdot)\}$

- E es utilizado para almacenar grafos

Sea Φ la siguiente $\mathcal{L}$-oración en ∃LSO:

$$\exists A \exists B \exists C \left(\begin{array}{l} \forall x \, (A(x) \vee B(x) \vee C(x)) \wedge \\ \forall x \, (\neg A(x) \vee \neg B(x)) \wedge \\ \forall x \, (\neg A(x) \vee \neg C(x)) \wedge \\ \forall x \, (\neg B(x) \vee \neg C(x)) \wedge \\ \forall x \forall y \, ((E(x, y) \wedge A(x)) \rightarrow \neg A(y)) \wedge \\ \forall x \forall y \, ((E(x, y) \wedge B(x)) \rightarrow \neg B(y)) \wedge \\ \forall x \forall y \, ((E(x, y) \wedge C(x)) \rightarrow \neg C(y)) \end{array} \right)$$

NP-hardness de ∃LSO: Demostración

Lema

L_Φ es NP-hard

Demostración: Reducimos desde el problema de 3-coloración de grafos.

- Dado un grafo $G = (N, A)$, $\mathfrak{A}_G$ se define como una $\mathcal{L}$-estructura tal que:
  - El dominio de $\mathfrak{A}_G$ es N
  - $E^{\mathfrak{A}_G} = A$

Se tiene que G es 3-coloreable si y sólo si $\mathfrak{A}_G \in L_\Phi$ (es decir, $\mathfrak{A}_G \models \Phi$) □

Complejidad Descriptiva

Dado: Vocabulario $\mathcal{L}$

Notación

- *Lenguaje:* *Subconjunto de* $\{\mathrm{enc}(\mathfrak{A}) \mid \mathfrak{A} \in \textsc{Struct}[\mathcal{L}]\}$
  - *Ejemplo:* $\mathcal{L} = \{G(\cdot,\cdot)\}$ *y* $L = \{\mathrm{enc}(\mathfrak{A}) \mid \mathfrak{A}$ *es 3-coloreable*$\}$
- *Clase de complejidad:* *Conjunto de lenguajes*
  - *Ejemplo:* $NP = \{L \mid$ *existe una MT no determinista* M *tal que* M *acepta* L *en tiempo polinomial*$\}$

Complejidad Descriptiva

Definición

Una lógica $\mathcal{LO}$ **captura** *una clase de complejidad $\mathcal{C}$ si:*

Complejidad Descriptiva

Definición

Una lógica $\mathcal{LO}$ **captura** *una clase de complejidad $\mathcal{C}$ si:*

- *Para toda oración φ en $\mathcal{LO}$, se tiene que:*

$$L_\varphi = \{\mathrm{enc}(\mathfrak{A}) \mid \mathfrak{A} \models \varphi\} \text{ está en } \mathcal{C}$$

Complejidad Descriptiva

Definición

Una lógica $\mathcal{LO}$ **captura** *una clase de complejidad $\mathcal{C}$ si:*

- *Para toda oración φ en $\mathcal{LO}$, se tiene que:*

$$L_\varphi = \{\mathrm{enc}(\mathfrak{A}) \mid \mathfrak{A} \models \varphi\} \text{ está en } \mathcal{C}$$

- *Para cada lenguaje $L \in \mathcal{C}$, existe una oración φ en $\mathcal{LO}$ tal que*

$$L = L_\varphi$$

Vale decir: $\mathrm{enc}(\mathfrak{A}) \in L$ *si y sólo si* $\mathfrak{A} \models \varphi$

Punto de partida de la complejidad descriptiva: Teorema de Fagin

Dado: Vocabulario $\mathcal{L}$

Teorema (Fagin)

$\exists$*LSO captura NP*

Codificando fórmulas y estructuras

Dado: Vocabulario $\mathcal{L}$ sin constantes

- Constantes son reemplazadas por predicados unarios

Asumimos que las fórmulas están dadas por su árbol de parseo (como un string binario).

Codificando fórmulas y estructuras

Dado: Vocabulario $\mathcal{L}$ sin constantes

- Constantes son reemplazadas por predicados unarios

Definimos la codificación de una $\mathcal{L}$-estructura $\mathfrak{A}$ ($\mathrm{enc}(\mathfrak{A})$) de la siguiente forma.

1. Tomamos un orden arbitrario en A: $a_1 < a_2 < \cdots < a_n$

   Este orden es usado para definir un orden lexicográfico para las k-tuplas ($k \geq 1$):

$$(a_1, \ldots, a_1, a_1) < (a_1, \ldots, a_1, a_2) < \cdots$$
$$\cdots < (a_1, \ldots, a_1, a_n) < (a_1, \ldots, a_2, a_1) < \cdots$$
$$\cdots < (a_n, \ldots, a_n, a_{n-1}) < (a_n, \ldots, a_n, a_n)$$

Codificando fórmulas y estructuras

2. Para R en $\mathcal{L}$ con aridad k: $\mathrm{enc}(R)$ es una palabra w de largo n^k, tal que el i-ésimo elemento de w es 1 si y sólo si la i-ésima tupla en el orden lexicográfico de las k-tuplas está en $R^{\mathfrak{A}}$

3. Si $\mathcal{L} = \{R_1, \ldots, R_\ell\}$, entonces $\mathrm{enc}(\mathfrak{A}) = \mathrm{enc}(R_1) \cdots \mathrm{enc}(R_\ell)$

Codificando fórmulas y estructuras

2. Para R en $\mathcal{L}$ con aridad k: $\mathrm{enc}(R)$ es una palabra w de largo n^k, tal que el i-ésimo elemento de w es 1 si y sólo si la i-ésima tupla en el orden lexicográfico de las k-tuplas está en $R^{\mathfrak{A}}$

3. Si $\mathcal{L} = \{R_1, \ldots, R_\ell\}$, entonces $\mathrm{enc}(\mathfrak{A}) = \mathrm{enc}(R_1) \cdots \mathrm{enc}(R_\ell)$

Ejemplo

Si $\mathfrak{A} = \langle A = \{1,2\}, P^{\mathfrak{A}} = \{1\}, R^{\mathfrak{A}} = \{(1,1),(2,1),(2,2)\}\rangle$, entonces:

$$\mathrm{enc}(\mathfrak{A}) \quad = \quad 101011$$

Punto de partida de la complejidad descriptiva: Teorema de Fagin

Dado: Vocabulario $\mathcal{L}$

Teorema (Fagin)

$\exists$LSO captura NP

Demostración: Sabemos que para toda oración φ en $\exists$LSO, se tiene que L_φ está en NP.

Vamos a demostrar la otra condición: Dado L en NP, vamos a encontrar φ en $\exists$LSO tal que $L = L_\varphi$.

- $\mathrm{enc}(\mathfrak{A}) \in L$ si y sólo si $\mathfrak{A} \models \varphi$

Teorema de Fagin: Demostración

Consideramos el caso $\mathcal{L} = \{G(\cdot, \cdot)\}$

- Para otros vocabularios la demostración es similar

Suponemos que L es aceptado por una MT no determinista M que funciona en tiempo polinomial.

También suponemos que $\Sigma = \{0, 1\}$ y $M = (Q, \Sigma, q_0, \delta, F)$, donde

- $Q = \{q_0, \ldots, q_m\}$
- $F = \{q_m\}$

Teorema de Fagin: Demostración

- Para todo $(q, a) \in (Q \setminus \{q_m\}) \times (\Sigma \cup \{\mathtt{B}, \vdash\})$, existe $(q', b, X) \in Q \times (\Sigma \cup \{\mathtt{B}, \vdash\}) \times \{\leftarrow, \square, \rightarrow\}$ tal que $(q, a, q', b, X) \in \delta$

- Para todo $a \in (\Sigma \cup \{\mathtt{B}, \vdash\})$, no existe $(q', b, X) \in Q \times (\Sigma \cup \{\mathtt{B}, \vdash\}) \times \{\leftarrow, \square, \rightarrow\}$ tal que $(q_m, a, q', b, X) \in \delta$

- M funciona en tiempo polinomial: Existe $k > 0$ tal que si M acepta $\mathrm{enc}(\mathfrak{A})$, entonces existe una ejecución de M que acepta $\mathrm{enc}(\mathfrak{A})$ en un número de pasos menor a $|\mathrm{enc}(\mathfrak{A})|^k$

Teorema de Fagin: Demostración

¿Por qué podemos hacer los supuestos anteriores?

Teorema de Fagin: Demostración

¿Por qué podemos hacer los supuestos anteriores?

- ¿Por qué podemos suponer que M ejecuta un número de pasos menor a n^k?

Teorema de Fagin: Demostración

¿Por qué podemos hacer los supuestos anteriores?

- ¿Por qué podemos suponer que M ejecuta un número de pasos menor a n^k?
- ¿Por qué podemos usar n^k en lugar de $c \cdot n^k$?

Teorema de Fagin: Demostración

¿Por qué podemos hacer los supuestos anteriores?

- ¿Por qué podemos suponer que M ejecuta un número de pasos menor a n^k?
- ¿Por qué podemos usar n^k en lugar de $c \cdot n^k$?

Ejercicio

Suponiendo que L es un lenguaje finito, construya una $\mathcal{L}$-oración φ en $\exists$LSO tal que $L = L_\varphi$.

Teorema de Fagin: Demostración

Dada una estructura $\mathfrak{A}$ con $|A| = n$, se tiene que $|\mathrm{enc}(\mathfrak{A})| = n^2$

- M recibe como entrada $\mathrm{enc}(\mathfrak{A})$: Funciona en tiempo $(n^2)^k = n^{2k}$

Para una estructura con n elementos, M funciona en tiempo n^{2k}

- Hay que tener esto en cuenta en la construcción de la oración que representa a L

Teorema de Fagin: Demostración

Definimos oración φ como:

$$\exists L\, \exists O\, \exists S\, \exists P\, \exists H\, \exists T_0\, \exists T_1\, \exists T_{\mathrm{B}}\, \exists T_{\vdash}\, \exists E_{q_0} \cdots \exists E_{q_m} (\varphi_L \wedge \varphi_O \wedge \varphi_S \wedge \varphi_P \wedge \varphi_I \wedge \varphi_C \wedge \varphi_A \wedge \varphi_\delta)$$

φ_L: L es un orden lineal

$$\forall x\, (\neg L(x,x)) \wedge \forall x \forall y\, (x = y \vee L(x,y) \vee L(y,x)) \wedge \forall x \forall y \forall z (L(x,y) \wedge L(y,z) \rightarrow L(x,z))$$

Teorema de Fagin: Demostración

φ_O: O es un orden lexicográfico, construido a partir de L, para las tuplas con $2k$ elementos

$$\forall x_1 \cdots \forall x_{2k} \forall y_1 \cdots \forall y_{2k} \left(O(x_1, \ldots, x_{2k}, y_1, \ldots, y_{2k}) \leftrightarrow \bigvee_{i=1}^{2k} \left(\left(\bigwedge_{j=1}^{i-1} x_j = y_j \right) \wedge L(x_i, y_i) \right) \right)$$

Teorema de Fagin: Demostración

φ_S: S es la relación de sucesor asociada a O

$$\forall \bar{x} \forall \bar{y} \, (S(\bar{x}, \bar{y}) \quad \leftrightarrow \quad (O(\bar{x}, \bar{y}) \wedge \neg \exists \bar{z} \, (O(\bar{x}, \bar{z}) \wedge O(\bar{z}, \bar{y}))))$$

Notación

Cada tupla de variables tiene largo $2k$

- $|\bar{x}| = |\bar{y}| = |\bar{z}| = 2k$ *en* φ_S

Teorema de Fagin: Demostración

φ_S: S es la relación de sucesor asociada a O

$$\forall \bar{x} \forall \bar{y} \, (S(\bar{x}, \bar{y}) \quad \leftrightarrow \quad (O(\bar{x}, \bar{y}) \wedge \neg \exists \bar{z} \, (O(\bar{x}, \bar{z}) \wedge O(\bar{z}, \bar{y}))))$$

Notación

Cada tupla de variables tiene largo $2k$

- $|\bar{x}| = |\bar{y}| = |\bar{z}| = 2k$ *en* φ_S

φ_P: P almacena el primer elemento de O

$$\forall \bar{x} \, (P(\bar{x}) \quad \leftrightarrow \quad \neg \exists \bar{y} \, O(\bar{y}, \bar{x}))$$

Teorema de Fagin: Demostración

φ_I: Representa el estado inicial de M

$$\forall\bar{x}\forall\bar{y}\left[\left(P(\bar{x})\wedge S(\bar{x},\bar{y})\right)\rightarrow\left(H(\bar{x},\bar{y})\wedge E_{q_0}(\bar{x})\wedge T_{\vdash}(\bar{x},\bar{x})\wedge\right.\right.$$
$$\exists u\exists v\Big(\neg\exists w\,(L(w,u))\wedge\neg\exists w\,(L(v,w))\wedge$$
$$\forall u_1\cdots\forall u_{2k}\forall\bar{v}\left(S(u_1,\ldots,u_{2k},\bar{v})\ \rightarrow\right.$$
$$\Big(\neg O(u,\ldots,u,v,v,u_1,\ldots,u_{2k})\ \rightarrow$$
$$((\neg G(u_{2k-1},u_{2k})\rightarrow T_0(\bar{x},\bar{v}))\wedge$$
$$\left.(G(u_{2k-1},u_{2k})\rightarrow T_1(\bar{x},\bar{v})))\right)\wedge$$
$$\left.\left.\left.\left.\Big(O(u,\ldots,u,v,v,u_1,\ldots,u_{2k})\rightarrow T_{\mathrm{B}}(\bar{x},\bar{v})\Big)\right)\right)\right)\right]$$

Teorema de Fagin: Demostración

φ_I: Representa el estado inicial de M

$$\forall\bar{x}\forall\bar{y}\left[\left(P(\bar{x})\wedge S(\bar{x},\bar{y})\right)\rightarrow\left(H(\bar{x},\bar{y})\wedge E_{q_0}(\bar{x})\wedge T_{\vdash}(\bar{x},\bar{x})\wedge\right.\right.$$
$$\exists u\exists v\left(\neg\exists w\,(L(w,u))\wedge\neg\exists w\,(L(v,w))\wedge\right.$$

(u es el primer elemento del orden lexicografico y v es el último)

Teorema de Fagin: Demostración

φ_I: Representa el estado inicial de M

$$\forall\bar{x}\forall\bar{y}\left[\left(P(\bar{x})\wedge S(\bar{x},\bar{y})\right)\rightarrow\left(H(\bar{x},\bar{y})\wedge E_{q_0}(\bar{x})\wedge T_{\vdash}(\bar{x},\bar{x})\wedge\right.\right.$$
$$\exists u\exists v\Big(\neg\exists w\,(L(w,u))\wedge\neg\exists w\,(L(v,w))\wedge$$
$$\forall u_1\cdots\forall u_{2k}\forall\bar{v}\left(S(u_1,\ldots,u_{2k},\bar{v})\ \rightarrow\right.$$
$$\left.\left(\neg O(u,\ldots,u,v,v,u_1,\ldots,u_{2k})\right.\right.$$

Si no es verdad que $u_1,\ldots,u_{2k}$ es mayor a n^2...
(u es el elemento 1 y v es el n-ésimo)

Teorema de Fagin: Demostración

φ_I: Representa el estado inicial de M

$$\forall\bar{x}\forall\bar{y}\left[\left(P(\bar{x})\wedge S(\bar{x},\bar{y})\right)\rightarrow\left(H(\bar{x},\bar{y})\wedge E_{q_0}(\bar{x})\wedge T_{\vdash}(\bar{x},\bar{x})\wedge\right.\right.$$
$$\exists u\exists v\Big(\neg\exists w\,(L(w,u))\wedge\neg\exists w\,(L(v,w))\wedge$$
$$\forall u_1\cdots\forall u_{2k}\forall\bar{v}\left(S(u_1,\ldots,u_{2k},\bar{v})\ \rightarrow\right.$$
$$\Big(\neg O(u,\ldots,u,v,v,u_1,\ldots,u_{2k})\ \rightarrow$$
$$((\neg G(u_{2k-1},u_{2k})\rightarrow T_0(\bar{x},\bar{v}))\wedge$$
$$\left.\left.(G(u_{2k-1},u_{2k})\rightarrow T_1(\bar{x},\bar{v})))\Big)\right)$$

Para la posición representada por $u,\ldots,u,u_{2k-1},u_{2k}$. Si el elemento u_{2k-1} está contectado con u_{2k}, pongo un 1 en esa posición en la cinta, de lo contrario pongo un 0

Teorema de Fagin: Demostración

φ_I: Representa el estado inicial de M

$$\forall\bar{x}\forall\bar{y}\left[\left(P(\bar{x})\wedge S(\bar{x},\bar{y})\right)\rightarrow\left(H(\bar{x},\bar{y})\wedge E_{q_0}(\bar{x})\wedge T_{\vdash}(\bar{x},\bar{x})\wedge\right.\right.$$
$$\exists u\exists v\Big(\neg\exists w\,(L(w,u))\wedge\neg\exists w\,(L(v,w))\wedge$$
$$\forall u_1\cdots\forall u_{2k}\forall\bar{v}\left(S(u_1,\ldots,u_{2k},\bar{v})\ \rightarrow\right.$$
$$\Big(\neg O(u,\ldots,u,v,v,u_1,\ldots,u_{2k})\ \rightarrow$$
$$((\neg G(u_{2k-1},u_{2k})\rightarrow T_0(\bar{x},\bar{v}))\wedge$$
$$\left.(G(u_{2k-1},u_{2k})\rightarrow T_1(\bar{x},\bar{v})))\right)\wedge$$
$$\left.\left.\left.\left.\Big(O(u,\ldots,u,v,v,u_1,\ldots,u_{2k})\rightarrow T_{\mathrm{B}}(\bar{x},\bar{v})\Big)\right)\right)\right)\right]$$

La cinta tiene un blanco en una posición mayor que n^2.

Teorema de Fagin: Demostración

φ_C: La máquina funciona correctamente. Se define como la conjunción de cuatro fórmulas

Primero, cada celda siempre contiene un único símbolo:

$$\forall\bar{x}\forall\bar{y}\Big[\Big(T_0(\bar{x},\bar{y}) \vee T_1(\bar{x},\bar{y}) \vee T_{\mathtt{B}}(\bar{x},\bar{y}) \vee T_{\vdash}(\bar{x},\bar{y})\Big) \wedge \Big(\neg T_0(\bar{x},\bar{y}) \vee \neg T_1(\bar{x},\bar{y})\Big) \wedge \Big(\neg T_0(\bar{x},\bar{y}) \vee \neg T_{\mathtt{B}}(\bar{x},\bar{y})\Big) \wedge \Big(\neg T_0(\bar{x},\bar{y}) \vee \neg T_{\vdash}(\bar{x},\bar{y})\Big) \wedge \Big(\neg T_1(\bar{x},\bar{y}) \vee \neg T_{\mathtt{B}}(\bar{x},\bar{y})\Big) \wedge \Big(\neg T_1(\bar{x},\bar{y}) \vee \neg T_{\vdash}(\bar{x},\bar{y})\Big) \wedge \Big(\neg T_{\mathtt{B}}(\bar{x},\bar{y}) \vee \neg T_{\vdash}(\bar{x},\bar{y})\Big)\Big]$$

Teorema de Fagin: Demostración

Segundo, la máquina siempre está en un único estado:

$$\forall \bar{x} \left(\bigvee_{q \in Q} \left(E_q(\bar{x}) \wedge \bigwedge_{q' \in (Q \setminus \{q\})} \neg E_{q'}(\bar{x}) \right) \right)$$

Teorema de Fagin: Demostración

Segundo, la máquina siempre está en un único estado:

$$\forall \bar{x} \left(\bigvee_{q \in Q} \left(E_q(\bar{x}) \wedge \bigwedge_{q' \in (Q \setminus \{q\})} \neg E_{q'}(\bar{x}) \right) \right)$$

Notación

$(y_1, \ldots, y_{2k}) = (z_1, \ldots, z_{2k})$ *representa a la fórmula* $\bigwedge_{i=1}^{2k} y_i = z_i$

Teorema de Fagin: Demostración

Segundo, la máquina siempre está en un único estado:

$$\forall \bar{x} \left(\bigvee_{q \in Q} \left(E_q(\bar{x}) \wedge \bigwedge_{q' \in (Q \setminus \{q\})} \neg E_{q'}(\bar{x}) \right) \right)$$

Notación

$(y_1, \ldots, y_{2k}) = (z_1, \ldots, z_{2k})$ *representa a la fórmula* $\bigwedge_{i=1}^{2k} y_i = z_i$

Tercero, la cabeza siempre está en una única posición:

$$\forall \bar{x} \exists \bar{y} \left(H(\bar{x}, \bar{y}) \wedge \forall \bar{z} \left(H(\bar{x}, \bar{z}) \rightarrow \bar{y} = \bar{z} \right) \right)$$

Teorema de Fagin: Demostración

Cuarto, el valor de una celda no cambia si no es apuntada por la cabeza lectora:

$$\forall \bar{x} \forall \bar{y} \forall \bar{z} \left[\left(S(\bar{x}, \bar{y}) \wedge \neg H(\bar{x}, \bar{z}) \right) \rightarrow \right.$$
$$\left((T_0(\bar{x}, \bar{z}) \wedge T_0(\bar{y}, \bar{z})) \vee (T_1(\bar{x}, \bar{z}) \wedge T_1(\bar{y}, \bar{z})) \vee \right.$$
$$\left. \left. (T_{\mathtt{B}}(\bar{x}, \bar{z}) \wedge T_{\mathtt{B}}(\bar{y}, \bar{z})) \vee (T_{\vdash}(\bar{x}, \bar{z}) \wedge T_{\vdash}(\bar{y}, \bar{z})) \right) \right]$$

Teorema de Fagin: Demostración

φ_A: La máquina acepta la entrada.

$$\exists \bar{x}\, E_{q_m}(\bar{x})$$

φ_δ: Representa la función de transición δ.

Teorema de Fagin: Demostración

φ_A: La máquina acepta la entrada.

$$\exists \bar{x}\, E_{q_m}(\bar{x})$$

φ_δ: Representa la función de transición δ.

- φ_δ se construye como la conjunción de las fórmulas $\varphi_{(q,a)}$, donde $q \neq q_m$ y $a \in (\Sigma \cup \{\mathtt{B}, \vdash\})$

Teorema de Fagin: Demostración

$\varphi_{(q,a)}$:

$$\forall\bar{x}\forall\bar{y}\Bigg[\Big(E_q(\bar{x}) \wedge T_a(\bar{x},\bar{y}) \wedge H(\bar{x},\bar{y})\Big) \rightarrow$$

$$\Bigg(\bigvee_{(q,a,q',b,\leftarrow)\in\delta} \forall\bar{u}\forall\bar{v}\,(S(\bar{x},\bar{u}) \wedge S(\bar{v},\bar{y}) \rightarrow$$

$$E_{q'}(\bar{u}) \wedge H(\bar{u},\bar{v}) \wedge T_b(\bar{u},\bar{y}))\Bigg) \vee$$

$$\Bigg(\bigvee_{(q,a,q',b,\rightarrow)\in\delta} \forall\bar{u}\forall\bar{v}\,(S(\bar{x},\bar{u}) \wedge S(\bar{y},\bar{v}) \rightarrow$$

$$E_{q'}(\bar{u}) \wedge H(\bar{u},\bar{v}) \wedge T_b(\bar{u},\bar{y}))\Bigg) \vee$$

$$\Bigg(\bigvee_{(q,a,q',b,\Box)\in\delta} \forall\bar{u}\,(S(\bar{x},\bar{u}) \rightarrow E_{q'}(\bar{u}) \wedge H(\bar{u},\bar{y}) \wedge T_b(\bar{u},\bar{y}))\Bigg)\Bigg]$$

Teorema de Fagin: Algunas consecuencias

Definimos el fragmento universal de la lógica de segundo orden ($\forall$LSO) como el conjunto de todas las fórmula en LSO de la forma:

$$\forall X_1 \forall X_2 \cdots \forall X_\ell \ \varphi,$$

donde φ es una fórmula en LPO.

Para un lenguaje L, definimos $\overline{L}$ como $\{\mathrm{enc}(\mathfrak{A}) \mid \mathrm{enc}(\mathfrak{A}) \notin L\}$

- coNP $= \{\overline{L} \mid L \in \mathrm{NP}\}$

Teorema de Fagin: Algunas consecuencias

Corolario

$\forall$*LSO captura coNP*

Teorema de Fagin: Algunas consecuencias

Corolario
$\forall LSO$ *captura coNP*

Corolario
$NP \neq coNP$ *si y sólo si* $\exists LSO \neq \forall LSO$

Teorema de Fagin: Algunas consecuencias

Corolario

$\forall$LSO captura coNP

Corolario

NP $\neq$ coNP si y sólo si $\exists$LSO $\neq$ $\forall$LSO, vale decir, si alguna de las siguientes condiciones es verdadera:

- *Existe una oración en $\exists$LSO que no es expresable en $\forall$LSO*
- *Existe una oración en $\forall$LSO que no es expresable en $\exists$LSO*