Lògica en la Informàtica

Definició de la Lògica de Primer Ordre (LPO)

Deducció en Lògica de Primer Ordre

José Miguel Rivero Robert Nieuwenhuis

Dept. Ciències de la Computació
Facultat de Informàtica
Universitat Politècnica de Catalunya (UPC)

Tardor 2022

FIB

Definició de la Lògica de Primer Ordre

Contiguts:

- Quins problemes són decidibles en LPO (o LPOI) i quins no?
- Tema 5: Deducció en LPO
  - Transformació a forma clausal EQUISATISFACTIBLE
  - Resolució en LPO

FIB

Definició de la Lògica de Primer Ordre

Quins problemes són decidibles en LPO (o LPOI) i quins no?

➤ Considerem problemes Booleans = problemes de decisió = problemes amb resposta si/no.

Definició

Un problema és DECIDIBLE si:
existeix algun procediment que sempre contesta correctament, en temps finit (és a dir, acaba).

Dins dels problemes decidibles, distingim classes de complexitat (en temps): logarítmic, lineal, quadràtic, polinòmic, exponencial, NP-complet, ...

Dins dels problemes INdecidibles, distingim altres classes: semi-decidibles, co-semi-decidibles, ...

FIB

Definició de la Lògica de Primer Ordre

Quins problemes són decidibles en LPO (o LPOI) i quins no?

En el context de la lògica, dos problemes importants són:

1: evaluació d'una formula: donades I i F, tenim $I \models F$?
2: SAT: donada una F, existeix alguna I tal que $I \models F$?

	avaluació	SAT
LProp:	lineal	NP-complet
LPO:	indecidible	indecidible

SAT en LPO és *co-semi-decidible*: existeix algun procediment que,

- si la resposta és NO (és a dir, F és insat), llavors contesta correctament "NO" en temps finit
- si la resposta és SI (és a dir, F és sat), o contesta correctament "SI" o no acaba

FIB

Definició de la Lògica de Primer Ordre

Quins problemes són decidibles en LPO (o LPOI) i quins no?

En general:

Un problema és *semi-decidible* si existeix algun procediment que,

- si la resposta és SI, llavors contesta correctament en temps finit
- si la resposta és NO, o contesta correctament "NO" o no acaba

Un problema és *CO-semi-decidible* si existeix algun procediment que,

- si la resposta és NO, llavors contesta correctament en temps finit
- si la resposta és SI, o contesta correctament "SI" o no acaba

FIB

Definició de la Lògica de Primer Ordre

Avaluació en LPO amb domini **finit** (és a dir, la I donada té domini finit) sí que és *decidible*. Per què?

Exemple d'avaluació en LPO amb domini finit:

Sigui la I següent:

$D_I = \{a, b\}$
$p_I(a, a) = 1$
$p_I(a, b) = 1$
$p_I(b, a) = 1$
$p_I(b, b) = 0$

Sigui la F següent:

$\forall x \, \exists y \; p(x, y)$

$\forall x_1 \, \exists y_1 \, \forall x_2 \, \exists y_2 \cdots \forall x_n \, \exists y_n \; F$

decidible però pot ser exponencial.

FIB

Definició de la Lògica de Primer Ordre

Avaluació en LPO amb domini **INfinit** és *INdecidible* en general. Per què?

El halting problem (el problema de la parada):
- donat un programa P, (o el que és el mateix, una màquina de Turing), P acaba?

Aquest problema es va demostrar que era indecidible.

A partir de aqui, es van demostrar indecidibles altres problemes, mitjançant reduccions entre problemes.
Per exemple, si pots reduir el "halting problem" a "SAT en LPO" (fer-ho mitjançant SAT en LPO),... llavors SAT en LPO també ha de ser indecidible!

FIB

Definició de la Lògica de Primer Ordre

Avaluació en LPO amb domini **INfinit** és *INdecidible* en general. Per què?

El problema "Arrell": donat un polinomi com a $x^3y^2 + 3x^4 + \cdots = 0$, té solucions ("arrells") senceres?
(trobar arrells de polinomis sobre diverses variables de grau arbitrari i amb productes entre variables).

Aquest problema es va demostrar que era indecidible. Es diu Hilbert's tenth problem.

FIB

Definició de la Lògica de Primer Ordre

Avaluació en LPO amb domini **INfinit** és *INdecidible* en general. Per què?

Podem reduir "Arrell" a l'avaluació en LPO amb domini infinit (fer "Arrell" mitjançant avaluació en LPO amb domini infinit):
Sigui I la interpretació amb $D_I = \mathbb{Z}$ (els enters) on $\{f^2, g^2\}$ s'interpreten com a suma i producte.

$$F \; = \; \exists x \, \exists y$$
$$\exists z \, (\forall y \; f(z, y) = y$$
$$\wedge$$
$$f(g(\underbrace{g(x, g(x,x)), g(y,y)}_{x^3 \qquad y^2})), \underbrace{f(f(g(g(x,x), g(x,x)), g(g(x,x), g(x,x))), g(g(x,x), g(x,x)))}_{+ \qquad x^4+x^4+x^4}\underbrace{, \ldots) = z)}_{+\cdots=0}$$

Tenim $I \models F$ ssi $x^3y^2 + 3x^4 + \cdots = 0$ té arrells senceres.

FIB

Avaluació en LPO amb domini **INfinit** és *INdecidible* en general. Per què?

Reduir "Arrell" al problema d'avaluació en LPO amb domini infinit: Si em donen un polinomi P, puc construir una fórmula F_P, tal que si I és la interpretació: $D_I = \mathbb{Z}$ (els enters) on $\{f^2, g^2\}$ s'interpreten com a suma i producte tenim $I \models F_P$ ssi P té arrells senceres.

Tema 5: Deducció en LPO

En L.Prop. teníem diversos mètodes per a SAT: $p \vee q \vee \neg r$

- El millor mètode per a SAT estava basat en un algorisme de backtracking amb propagació, etc., que explora el conjunt de possibles models (totes les interpretacions).
  En LPO aquest mètode no existeix.
  No hi ha manera de "enumerar" totes les I's.
- Però en L.Prop. vam veure un altre, basat en resolució, amb el teorema:
  un cjto de clausulas S és insat ssi $\square \in Res(S)$.
  En LPO, l'únic mètode per a SAT que estudiarem és el basat en resolució.

Una clàusula en LPO és una disjunció de literals, com en L.Prop, però en LPO els literals ja no són símbols de predicat o símbols de predicat negats, sinó que són ÀTOMS, o ÀTOMS NEGATS.

Poden contenir variables, que TOTES s'entenen que estan universalment quantificades:

$\forall x_1 \cdots \forall x_m\ L_1 \vee \cdots \vee L_n$ però normalment els $\forall x_1 \cdots \forall x_m$ no els escrivim.

Un exemple de Prolog:

`tio(S,T) :- padre(S,P), hermano(P,T).`

`tio(S,T)` $\leftarrow$ `padre(S,P)` $\wedge$ `hermano(P,T)`

`tio(S,T)` $\vee \neg($ `padre(S,P)` $\wedge$ `hermano(P,T)` $)$

`tio(S,T)` $\vee \neg$`padre(S,P)` $\vee \neg$`hermano(P,T)` és una clàusula de Horn de LPO (i no escrivim els "per a tot" $\forall S \forall T \forall P$)

Un literal és un àtom $p(t_1, \ldots, t_n)$ o un àtom negat $\neg p(t_1, \ldots, t_n)$.

Per a què volem SAT en LPO?

Per al mateix en L.Prop, per a les aplicacions pràctiques, i tenim les propietats:

F SAT?			
F insat?			
F Taut?	ssi	$\neg F$ insat	$\to$ (1)
$F \models G$?	ssi	$F \wedge \neg G$ insat	
$F \equiv G$?	ssi	$F \wedge \neg G \ \vee\ G \wedge \neg F$ insat	

(1) Taut en LPO és *semi-decidible* (pq és equivalent a un problema de INsat)

Per a poder fer SAT en LPO mitjançant resolució:

Transformació a forma clausal EQUISATISFACTIBLE

(com passava amb la transformació de Tseitin en L.Prop)

"Forma clausal" = conjunt (conjunció, un AND) de clàusules.

Equisatisfactible: Si una formula F té el cjto de clàusules S com a forma clausal, tenim que: F sat ssi S sat.

Per a poder fer SAT en LPO mitjançant resolució:

Transformació a forma clausal EQUISATISFACTIBLE

(com passava amb la transformació de Tseitin en L.Prop)

1. Moviment de les negacions cap a dins
2. Eliminació de conflictes de nom de variable
3. [Opcional] Moviment de quantificadors cap a dins mentre sigui possible
4. Eliminació de quantificadors existencials o Skolemización
5. Moviment de quantificadors universals cap a fora
6. Aplicar distributivitat

Transformació a forma clausal EQUISATISFACTIBLE:

1. Moviment de les negacions cap a dins:
   $\neg(F \wedge G) \Rightarrow \neg F \vee \neg G$
   $\neg(F \vee G) \Rightarrow \neg F \wedge \neg G$
   $\neg\neg F \Rightarrow F$
   $\neg \exists x\, F \Rightarrow \forall x \neg F$
   $\neg \forall x\, F \Rightarrow \exists x \neg F$
2. Eliminació de conflictes de nom de variable:
   per exemple: $\forall x\, p(x) \wedge \exists x\, q(x) \implies \forall x\, p(x) \wedge \exists x'\, q(x')$
3. [Opcional; és només per eficiència] Moviment de quantificadors cap a dins mentre sigui possible:
   per exemple: $\forall x\,(p(a) \wedge q(x)) \implies p(a) \wedge \forall x\, q(x)$

Transformació a forma clausal EQUISATISFACTIBLE:

4. Eliminació de quantificadors existencials o Skolemització:
   (és l'únic pas dels 6 que NO preserva l'equivalència logica; però sí la equisatisfactibilitat)
   2 exemples:
   1. $\forall x \exists y\ p(x, y) \xrightarrow{sk} \forall x\ p(x, f_y(x))$ on f_y és un símbol de funció nou "fresc"
   2. $\exists y \forall x\ p(x, y) \xrightarrow{sk} \forall x\ p(x,\ c_y\ )$ on c_y és un símbol de funció nou "fresc" (en aquest cas, una cte)

Transformació a forma clausal EQUISATISFACTIBLE:

4. Eliminació de quantificadors existencials o Skolemització:
(és l'únic pas dels 6 que NO preserva l'equivalència logica; però sí la equisatisfactibilitat)

2 exemples:

1. $\forall x \exists y\ p(x,y)$
2. $\exists y \forall x\ p(x,y)$

Si tenim la interpretació I tal que:

$D_I = \{a, b\}$
$p_I(a,a) = 0$
$p_I(a,b) = 1$
$p_I(b,a) = 1$
$p_I(b,b) = 0$

tenim que $I \models \forall x \exists y\ p(x,y)$,
però $I \not\models \exists y \forall x\ p(x,y)$.

Intuïtivament, tenim que $\exists y \forall x\ p(x,y) \models \forall x \exists y\ p(x,y)$.

Transformació a forma clausal EQUISATISFACTIBLE:

4. Eliminació de quantificadors existencials o Skolemització:

La Skolemización NO dona una fórmula lògicament equivalent:
tenim $\forall x \exists y\ p(x,y) \xrightarrow{sk} \forall x\ p(x, f_y(x))$
donem una I tal que:

- $I \models \forall x \exists y\ p(x,y)$
- $I \not\models \forall x\ p(x, f_y(x))$

$D_I = \{a, b\}$
$p_I(a,a) = 0$ $\quad f_{y_I}(a) = a$
$p_I(a,b) = 1$ $\quad f_{y_I}(b) = a$
$p_I(b,a) = 1$
$p_I(b,b) = 0$

això és model de $\forall x \exists y\ p(x,y)$ però NO és model de $\forall x\ p(x, f_y(x))$.

Transformació a forma clausal EQUISATISFACTIBLE:

4. Eliminació de quantificadors existencials o Skolemització:

La Skolemización NO dona una fórmula lògicament equivalent:
tenim $\forall x \exists y\ p(x,y) \xrightarrow{sk} \forall x\ p(x, f_y(x))$
donem una I tal que:

- $I \models \forall x \exists y\ p(x,y)$
- $I \not\models \forall x\ p(x, f_y(x))$

$D_I = \{a, b\}$
$p_I(a,a) = 0$ $\quad f_{y_I}(a) = b$
$p_I(a,b) = 1$ $\quad f_{y_I}(b) = a$
$p_I(b,a) = 1$
$p_I(b,b) = 0$

En canvi, si interpreto f_y així (és a dir, "bé", com ho feia l'existeix en $\forall x \exists y\ p(x,y)$), llavors f_{y_I} "tria" el valor adequat perquè I SÍ QUE sigui model de $\forall x\ p(x, f_y(x))$.

Transformació a forma clausal EQUISATISFACTIBLE:

4. Eliminació de quantificadors existencials o Skolemització:

La Skolemización NO dona una fórmula lògicament equivalent

En general:

Si $F \xrightarrow{sk} F'$ llavors donat un model de F puc construir un model de F', i viceversa: F i F' són **equisatisfactibles**.

Transformació a forma clausal EQUISATISFACTIBLE:

5. Moviment de quantificadors universals cap a fora
Per exemple:
$F \wedge \forall x\ G \implies \forall x\ F \wedge G$
6. Distributivitat amb: $(F \wedge G) \vee H \Rightarrow (F \vee H) \wedge (G \vee H)$
(això pot fer créixer la fórmula exponencialment, perquè la part H es duplica; hi ha mètodes similars a Tseitin per a evitar aquest problema).

Resolució en LPO

- Resolució en L.Proposicional
- Regla de resolució en LPO
- No terminació de la resolució en LPO

Recordem: Resolució en L.Proposicional:

$$\frac{p \vee C \qquad \neg p \vee D}{C \vee D}$$

Teorema

S insat ssi $\square \in Res(S)$

Aquest teorema també és cert en LPO (bé, "gaire bé cert"; veure més endavant)

Recordem: Resolució en L.Proposicional:

$$\frac{p \vee C \qquad \neg p \vee D}{C \vee D}$$

Teorema

S insat ssi $\square \in Res(S)$

$S_0 = S$
$S_1 = S_0 \cup Res_1(S_0)$ $\quad$ ($Res_1(S_0)$ = el que puc obtenir en 1 pas de resolució a partir de S_0)
$S_2 = S_1 \cup Res_1(S_1)$ $\quad$ ($Res_1(S_1)$ = el que puc obtenir en 1 pas de resolució a partir de S_1)
... ...
$Res(S) = \bigcup_{i=0}^{\infty} S_i$

Resolució en LPO

$$\frac{A \vee C \qquad \neg B \vee D}{(C \vee D)\sigma}$$

A, B són àtoms
si $\sigma = mgu(A, B)$
(*most general unifier*)

Exemple: x, y són vars $\quad a, b$ són ctes:

$$\frac{p(a,x) \vee q(x) \qquad \neg p(y,b) \vee r(y)}{(\,q(x) \vee r(y)\,)\sigma}$$

si $\sigma = \{x = b, y = a\}$
$q(b) \vee r(a)$

$$\frac{p(a,b) \vee q(b) \qquad \neg p(a,b) \vee r(a)}{q(b) \vee r(a)}$$

En LPO, la resolució pot no acabar:

$$\frac{p(a) \qquad \neg p(x) \vee p(f(x))}{p(f(a))} \qquad mgu(\ p(a), p(x)\ ) = \{x\!=\!a\}$$

$$\frac{p(f(a)) \qquad \neg p(x) \vee p(f(x))}{p(f(f(a)))} \qquad mgu(\ p(f(a)), p(x)\ ) = \{x\!=\!f(a)\}$$

1. $p(a)$
2. $\neg p(x) \vee p(f(x))$

puc obtenir,

3. $p(f(a))$ amb $mgu(p(a), p(x)) = \{x\!=\!a\}$:
4. $p(f(f(a)))$ 3. amb la 2. amb $mgu(\ p(f(a)), p(x)\ ) = \{x\!=\!f(a)\}$:
5. $p(f(f(f(a))))$ 4. amb la 2. amb $mgu(\ p(f(f(a))), p(x)\ ) = \{x\!=\!f(f(a))\}$:
6. ...

En LPO, la resolució pot no acabar:

$$\frac{p(a) \qquad \neg p(x) \vee p(f(x))}{p(f(a))} \qquad mgu(\ p(a), p(x)\ ) = \{x\!=\!a\}$$

$$\frac{p(f(a)) \qquad \neg p(x) \vee p(f(x))}{p(f(f(a)))} \qquad mgu(\ p(f(a)), p(x)\ ) = \{x\!=\!f(a)\}$$

El mateix exemple, de forma més "natural":

1. $nat(0)$
2. $\neg nat(x) \vee nat(succ(x))$

En LPO, la resolució pot no acabar:

Un altre exemple de no-terminació, sense símbol de funció:

1. $\neg p(x,y) \vee \neg p(y,z) \vee p(x,z)$

1. $\neg p(x,y) \ \vee\ \neg p(y,z) \ \vee\ \underline{p(x,z)}$

   $\underline{\neg p(x',y')} \vee \neg p(y',z') \vee p(x',z')$

el *mgu* és $\{x'\!=\!x, y'\!=\!z\}$ i obtenim:

2. $\neg p(x,y) \vee \neg p(y,z) \vee \neg p(z,z') \vee p(x,z')$
   (una mena de "transitivitat de 4")

... etc

Per al proper dia de classe:

- Unificació
- Veure el capítol 5 dels apunts, i els exercicis.

  ☞ `p5.pdf`