Tema AR–5: Refinamientos de resolución

José A. Alonso Jiménez
José L. Ruiz Reina

Dpto. de Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

P–resolución

- **Cláusulas positivas:**
  - Def.: C es positiva si todos sus literales son positivos
  - Ejemplos:
    $\{p, q, r\}$ es positiva
    $\{p, \neg q\}$ no es positiva

- **Demostración por P–resolución**
  - Def.: La sucesión $(C_1, \dots, C_n)$ es una demostración por P–resolución de C a partir de S si $C_n = C$ y para todo $i \in \{1, \dots, n\}$ se verifica una de las siguientes condiciones:
    * $C_i \in S$;
    * existen $j, k < i$ tales que $C_i \in$ resolventes(C_j, C_k) y alguna de C_j y C_k es positiva
  - Def.: C es demostrable por P–resolución a partir de S si existe una demostración por P-resolución de C a partir de S
  - Representación: $S \vdash_P C$

P–resolución

- **Ejemplo:**

```
> (prueba-por-P-resolucion
   '(((- p1) p2)
     ((- p2) p3)
     ((- p3) p4)
     (p3)
     ((- p4))))

========== Soporte ==========
1 NIL {-P1,P2}
2 NIL {-P2,P3}
3 NIL {-P3,P4}
4 NIL {P3}
5 NIL {-P4}
====== Fin del soporte ======

1 NIL {-P1,P2}
2 NIL {-P2,P3}
3 NIL {-P3,P4}
4 NIL {P3}
  ** 6 (4 3) {P4}

========== Prueba ==========
3 NIL {-P3,P4}
4 NIL {P3}
5 NIL {-P4}
6 (4 3) {P4}
7 (6 5) {}
===== Fin de la prueba =====

T
```

P–resolución

- **Propiedades del cálculo por P–resolución:**
  - **Adecuación:** $S \vdash_P \{\} \Longrightarrow S$ es inconsistente
  - **Completitud:** S es inconsistente $\Longrightarrow S \vdash_P \{\}$
- **N–resolución**
- **Resolución semántica**

Estrategia del conjunto soporte

- **Notación:** Sean S un conjunto de cláusulas y $T \subseteq S$ tal que $S - T$ es consistente
- **Demostración por resolución con soporte:**
  - Def.: La sucesión $(C_1, \dots, C_n)$ es una demostración por resolución de C a partir de S con soporte T si $C_n = C$ y para todo $i \in \{1, \dots, n\}$ se verifica una de las siguientes condiciones:
    * $C_i \in S$;
    * existen $j, k < i$ tales que $C_i \in \text{resolventes}(C_j, C_k)$ y alguna de C_j y C_k no pertenece a $S - T$
  - Def.: C es demostrable por resolución con soporte T a partir de S si existe una demostración por resolución de C a partir de S con soporte T
  - Representación: $S \vdash_{\text{soporte } T} C$
- **Adecuación y completitud**
  - Si $T \subseteq S$ y $S - T$ es consistente, entonces S es inconsistente $\Longleftrightarrow S \vdash_{\text{soporte } T} \{\}$

Estrategia del conjunto soporte

- **Ejemplo:**
  $\{\{\neg p_1, p_2\}, \{\neg p_2, p_3\}, \{\neg p_3, p_4\}, \{p_1\}, \{\neg p_2\}\}$
  $\vdash_{\textbf{soporte}\ \{\neg p_2\}} \{\}$

```
> (prueba-por-resolucion-con-soporte
   '(((- p1) p2)
     ((- p2) p3)
     ((- p3) p4)
     (p1))
   '(((- p2))))

========== Usables ==========
1 NIL {-P1,P2}
2 NIL {-P2,P3}
3 NIL {-P3,P4}
4 NIL {P1}
====== Fin de usables ======

========== Soporte ==========
5 NIL {-P2}
====== Fin del soporte ======

5 NIL {-P2}
  ** 6 (5 1) {-P1}

========== Prueba ==========
1 NIL {-P1,P2}
4 NIL {P1}
5 NIL {-P2}
6 (5 1) {-P1}
7 (6 4) {}
===== Fin de la prueba =====

T
```

Estrategia del conjunto soporte

```
> (prueba
   '(((- p1) p2)
     ((- p2) p3)
     ((- p3) p4)
     (p1)
     ((- p2))))

========== Soporte ==========
1 NIL {-P1,P2}
2 NIL {-P2,P3}
3 NIL {-P3,P4}
4 NIL {P1}
5 NIL {-P2}
====== Fin del soporte ======

1 NIL {-P1,P2}
2 NIL {-P2,P3}
  ** 6 (2 1) {P3,-P1}
3 NIL {-P3,P4}
  ** 7 (3 2) {P4,-P2}
4 NIL {P1}
  ** 8 (4 1) {P2}

========== Prueba ==========
1 NIL {-P1,P2}
4 NIL {P1}
5 NIL {-P2}
8 (4 1) {P2}
9 (8 5) {}
===== Fin de la prueba =====

T
```

Resolución unidad

- **Demostración por resolución unidad:**
  - Def.: La sucesión $(C_1, \dots, C_n)$ es una demostración por resolución unidad de C a partir de S si $C_n = C$ y para todo $i \in \{1, \dots, n\}$ se verifica una de las siguientes condiciones:
    * $C_i \in S$;
    * existen $j, k < i$ tales que $C_i \in$ resolventes(C_j, C_k) y alguna de C_j y C_k es unitaria
  - Def.: C es demostrable por resolución unidad a partir de S si existe una demostración por resolución unidad de C a partir de S
  - Representación: $S \vdash_U C$
- **Propiedades del cálculo por resolución unidad:**
  - Adecuación: $S \vdash_U \{\} \Longrightarrow S$ es inconsistente
  - S es inconsistente $\not\Longrightarrow S \vdash_U \{\}$
  - Ejemplo: Sea $S = \{\{p, q\}, \{p, \neg q\}, \{\neg p, q\}, \{\neg p, \neg q\}\}$: Entonces S es inconsistente, pero $S \not\vdash_U \{\}$

Resolución unidad

- **Ejemplo:**

```
> (prueba-por-resolucion-unidad
   '((p q)
     ((- p) r)
     ((- q) r)
     ((- q) s)
     ((- r))))
========== Soporte ==========
1 NIL {P,Q}
2 NIL {-P,R}
3 NIL {-Q,R}
4 NIL {-Q,S}
5 NIL {-R}
====== Fin del soporte ======
1 NIL {P,Q}
2 NIL {-P,R}
3 NIL {-Q,R}
4 NIL {-Q,S}
5 NIL {-R}
  ** 6 (5 2) {-P}
  ** 7 (5 3) {-Q}
6 (5 2) {-P}
  ** 8 (6 1) {Q}
========== Prueba ==========
1 NIL {P,Q}
2 NIL {-P,R}
3 NIL {-Q,R}
5 NIL {-R}
6 (5 2) {-P}
7 (5 3) {-Q}
8 (6 1) {Q}
9 (8 7) {}
===== Fin de la prueba =====
T
```

Resolución por entradas

- **Demostración por resolución por entradas:**
  - Def.: La sucesión $(C_1, \dots, C_n)$ es una demostración por resolución por entradas de C a partir de S si $C_n = C$ y para todo $i \in \{1, \dots, n\}$ se verifica una de las siguientes condiciones:
    * $C_i \in S$;
    * existen $j, k < i$ tales que $C_i \in$ resolventes(C_j, C_k) y alguna de C_j y C_k pertenece a S
  - Def.: C es demostrable por resolución por entradas a partir de S si existe una demostración por resolución por entradas de C a partir de S
  - Representación: $S \vdash_E C$
- **Propiedades del cálculo por resolución por entradas:**
  - Adecuación: $S \vdash_E \{\} \Longrightarrow S$ es inconsistente
  - S es inconsistente $\not\Longrightarrow S \vdash_E \{\}$
  - Ejemplo: Sea $S = \{\{p, q\}, \{p, \neg q\}, \{\neg p, q\}, \{\neg p, \neg q\}\}$: Entonces S es inconsistente, pero $S \not\vdash_E \{\}$

Resolución por entradas

- **Ejemplo:**

```
> (prueba-por-resolucion-por-entradas
   '((p q)
     ((- p) r)
     ((- q) r)
     ((- q) s)
     ((- r))))
========== Soporte ==========
1 NIL {P,Q}
2 NIL {-P,R}
3 NIL {-Q,R}
4 NIL {-Q,S}
5 NIL {-R}
====== Fin del soporte ======
1 NIL {P,Q}
2 NIL {-P,R}
  ** 6 (2 5) {-P}
3 NIL {-Q,R}
  ** 7 (3 5) {-Q}
4 NIL {-Q,S}
5 NIL {-R}
6 (2 5) {-P}
  ** 8 (6 1) {Q}
========== Prueba ==========
1 NIL {P,Q}
2 NIL {-P,R}
3 NIL {-Q,R}
5 NIL {-R}
6 (2 5) {-P}
7 (3 5) {-Q}
8 (6 1) {Q}
9 (8 7) {}
===== Fin de la prueba =====
T
```

Resolución lineal

- **Demostración por resolución lineal**
  - Def.: La sucesión $(C_0, \dots, C_n)$ es una demostración por resolución lineal de C a partir de S con base $C_0 \in S$ si $C_n = C$ y para cada $i \in \{1, \dots, n\}$ existe un $B_{i-1} \in S \cup \{C_0, \dots, C_{i-1}\}$ tal que $C_i \in$ resolventes(C_{i-1}, B_{i-1})
  - C_0 es la cláusula base
    C_i se llaman cláusulas centrales
    B_i se llaman cláusulas laterales
  - Def.: C es demostrable por resolución lineal a partir de S con base C_0 si existe una demostración lineal de C a partir de S con base C_0
  - Representación: $S \vdash_{L,C_0} C$
- **Adecuación y completitud**
  - Si $C_0 \in S$ y $S - \{C_0\}$ es consistente, entonces
    S es inconsistente $\iff S \vdash_{L,C_0} \{\}$

Resolución lineal

- **Ejemplo:** $\{\{p, \neg q\}, \{p, \neg r\}, \{r\}, \{\neg p\}\} \vdash_{L,\{\neg p\}} \{\}$

```
> (prueba-por-resolucion-lineal
    '((p (- q))
      (p (- r))
      (r))
    '((- p)))

========== Entradas ==========
1 NIL {P,-Q}
2 NIL {P,-R}
3 NIL {R}
====== Fin del entradas ======

4 NIL {-P}
5 (4 1) {-Q}
··· Fallo ···

6 (4 2) {-R}
7 (6 3) {}

========== Prueba ==========
2 NIL {P,-R}
3 NIL {R}
4 NIL {-P}
6 (4 2) {-R}
7 (6 3) {}
===== Fin de la prueba =====

T
```

Resolución lineal

- **Procedimiento**

```
(defun prueba-por-resolucion-lineal (S C)
  (setf *contador* 0)
  (let ((S-anotado (mapcar #'(lambda (C1)
                                (crea-clausula-anotada :numero (incf *contador*)
                                                       :clausula C1))
                            S))
        (C-anotada (crea-clausula-anotada :clausula C)))
    (format t "~%========== Entradas ==========")
    (loop for C1 in S-anotado do (print C1))
    (format t "~%====== Fin del entradas ======~%~%")
    (catch 'prueba
      (prueba-por-resolucion-lineal-aux S-anotado C-anotada))))
```

Resolución lineal

```
(defun prueba-por-resolucion-lineal-aux (SA CA)
  (numera CA) (format t "~&~a" CA)
  (cond ((and (es-unitaria (ca-clausula CA))
              (complementaria CA SA ()))
         (procesa-unitaria CA SA ())
         (throw 'prueba t))
        (t (some #'(lambda (CA1)
                     (prueba-por-resolucion-lineal-aux
                      (n-union SA (list CA) :test #'igual-clausula-anotada)
                      CA1))
                 (nuevas-resolventes-anotadas CA SA)))))

(defun nuevas-resolventes-anotadas (CA SA)
  (let ((res ()))
    (loop for CA1 in (resolventes-ca-conjunto CA SA) do
      (when (and (not (es-tautologia (ca-clausula CA1)))
                 (not (member CA1 SA :test #'igual-clausula-anotada)))
            (push CA1 res)))
    (when (null res) (format t "~&... Fallo ...~%~%"))
    (nreverse res)))
```

Referencias

- Chang, C–L y Lee, R. C–T. *Symbolic Logic and Mechanical Theorem Proving* (Academic Press, 1973)
  - Cap. 6: "Semantic resolution and lock resolution"
  - Cap. 7: "Linear resolution"
- Genesereth, M.R. y Nilsson, N.J. *Logical Foundations of Artificial Intelligence* (Morgan Kaufmann, 1987)
  - Cap. 4 "Resolution"
- Lucas, P. y Gaag, L.v.d. *Principles of Expert Systems* (Addison–Wesley, 1991).
  - Cap. 2 "Logic and resolution"
- Russell, S. y Norvig, P. *Inteligencia artificial (un enfoque moderno)* (Prentice–Hall, 1996)
  - Cap. 9.5 "Resolución: un procedimiento completo de inferencia"

Referencias

- Thayse, A. y otros *Aproche logique de l'Intelligence Artificielle. (Vol 1: de la logique classique à la programmation logique).* (Dunod, 1988)
  - Cap. 1.1 "Calcul des propositions"