



Programare logică

Algoritmul de unificare

 F. Baader, T. Nipkow,
Terms Rewriting and All That,
Cambridge University Press, 1998.

Substituție

(S, Σ) semnătură multisortată, X și Y mulțimi de variabile
O **substituție** a variabilelor din X cu termeni din $T_\Sigma(Y)$ este o funcție $\nu : X \rightarrow T_\Sigma(Y)$. Substituția ν se extinde la o funcție $\tilde{\nu} : T_\Sigma(X) \rightarrow T_\Sigma(Y)$ după cum urmează:

- $\tilde{\nu}_s(x) := \nu(x)$ or. $x \in X_s$,
- $\tilde{\nu}_s(\sigma) := \sigma$ or. $\sigma : \rightarrow s$,
- $\tilde{\nu}_s(\sigma(t_1, \dots, t_n)) := \sigma(\tilde{\nu}_{s_1}(t_1), \dots, \tilde{\nu}_{s_n}(t_n))$ or.
 $\sigma : s_1 \dots s_n \rightarrow s$, or. $t_1 \in T_\Sigma(X)_{s_1}, \dots, t_n \in T_\Sigma(X)_{s_n}$.
- $\{x_1 \leftarrow t_1, \dots, x_n \leftarrow t_n\}$ e notație pt. $\nu : X \rightarrow T_\Sigma(X)$,
 $\nu(x_i) := t_i$ or. $i = 1, \dots, n$ și $\nu(x) := x$ pt. $x \neq x_i$

Ecuatii

(S, Σ) semnatură monosortată, $S = \{s\}$

X mulțime de variabile,

$T_\Sigma(X)$ termenii cu variabile din X

- O **ecuație** este o pereche de termeni $\langle t, t' \rangle$,
unde $t, t' \in T_\Sigma(X)$. Ecuația $\langle t, t' \rangle$ o vom nota $t \doteq t'$.
- Dacă $t = \sigma(t_1, \dots, t_n)$ și $t' = \tau(t'_1, \dots, t'_k)$ atunci
 $t = t' \Leftrightarrow \sigma = \tau, n = k, t_i = t'_i$ or. i

\doteq egalitate formală, $=$ egalitate efectivă

Unificare

- O **problemă de unificare** este o mulțime finită de ecuații

$$U = \{t_1 \doteq t'_1, \dots, t_n \doteq t'_n\}$$

- Un **unificator** (o **soluție**) pentru U este o substituție

$$\nu : X \rightarrow T_\Sigma(X) \text{ a.î. } \nu(t_i) = \nu(t'_i) \text{ or. } i = 1, \dots, n.$$

- **Notatii**

- $Unif(U) :=$ mulțimea unificatorilor lui U

- $Var(U) := \bigcup_{i=1}^n (Var(t_i) \cup Var(t'_i))$, unde

$$Var(t) := \text{mulțimea variabilelor care apar în } t \in T_\Sigma(X)$$

- dacă $\nu = \{x_1 \leftarrow t_1, \dots, x_n \leftarrow t_n\}$ atunci

$$\{x_1 \leftarrow t_1, \dots, x_n \leftarrow t_n\}U := \{\nu(t) \doteq \nu(t') \mid t \doteq t' \in U\}$$

Unificare

- Spunem că problema de unificare $R = \{x_1 \doteq t_1, \dots, x_n \doteq t_n\}$ este **rezolvată** dacă
 - $x_i \in X$, $x_i \neq x_j$ or. $i \neq j$
 - $x_i \notin \bigcup_{i=1}^n Var(t_i)$ or. $i = 1, \dots, n$.
- $\{x_1 \leftarrow t_1, \dots, x_n \leftarrow t_n\}$ este **cgu idempotent** pentru R .
- Algoritmul transformă o problemă de unificare U într-o problemă de unificare R . Dacă $R = \emptyset$ atunci U nu are unificatori. În caz contrar, R este rezolvată, iar substituția determinată de R este cgu idempotent pentru U .

Algoritmul de unificare

$(S = \{s\}, \Sigma)$ **signatură**,
 X **mulțime de variabile**

Intrare: $U = \{t_1 \doteq t'_1, \dots, t_n \doteq t'_n\}$ **Inițializare:** $R = U$

Se execută nedeterminist următorii pași, atâta timp cât este posibil:

Șterge: $R \cup \{t \doteq t\} \Rightarrow R$

Orientează: $R \cup \{t \doteq x\} \Rightarrow R \cup \{x \doteq t\}$ **dc.** $x \in X, t \notin X$

Descompune:

- $R \cup \{\sigma(e_1, \dots, e_n) \doteq \sigma(e'_1, \dots, e'_n)\} \Rightarrow R \cup \{e_1 \doteq e'_1, \dots, e_n \doteq e'_n\}$
- $R \cup \{\sigma(e_1, \dots, e_n) \doteq \tau(e'_1, \dots, e'_k)\} \Rightarrow \emptyset$ **dc.** $\sigma \neq \tau$

Elimină:

- $R \cup \{x \doteq t\} \Rightarrow \{x \doteq t\} \cup \{x \leftarrow t\}R$ **dc..** $x \in Var(R) \setminus Var(t)$
- $R \cup \{x \doteq t\} \Rightarrow \emptyset$ **dc.** $x \in Var(t)$ și $t \neq x$

leșire: dacă $R = \emptyset$ atunci nu există soluții pentru U

dacă $R \neq \emptyset$ atunci R este **cgu** pentru U

Algoritmul de unificare

Se execută nedeterminist următorii pași, atâta timp cât este posibil:

Șterge: $R \cup \{t \dot{=} t\} \Rightarrow R$

Orientează: $R \cup \{t \dot{=} x\} \Rightarrow R \cup \{x \dot{=} t\}$ **dc.** $x \in X, t \notin X$

Descompune:

- $R \cup \{\sigma(e_1, \dots, e_n) \dot{=} \sigma(e'_1, \dots, e'_n)\} \Rightarrow R \cup \{e_1 \dot{=} e'_1, \dots, e_n \dot{=} e'_n\}$
- $R \cup \{\sigma(e_1, \dots, e_n) \dot{=} \tau(e'_1, \dots, e'_k)\} \Rightarrow \emptyset$ **dc.** $\sigma \neq \tau$

Elimină:

- $R \cup \{x \dot{=} t\} \Rightarrow \{x \dot{=} t\} \cup \{x \leftarrow t\}R$ **dc.** $x \in Var(R) \setminus Var(t)$
- $R \cup \{x \dot{=} t\} \Rightarrow \emptyset$ **dc.** $x \in Var(t)$ și $t \neq x$

$$\Sigma = \{g : s \rightarrow s, h : s \rightarrow s, f : sss \rightarrow s\},$$

$$X = \{x, y, z, w\}$$

$$U = R = \{g(y) \dot{=} x, f(x, h(x), y) \dot{=} f(g(z), w, z)\}$$

$$(2) R = \{x \dot{=} g(y), f(x, h(x), y) \dot{=} f(g(z), w, z)\}$$

$$(4) R = \{x \dot{=} g(y), f(g(y), h(g(y)), y) \dot{=} f(g(z), w, z)\}$$

$$(3) R = \{x \dot{=} g(y), g(y) \dot{=} g(z), h(g(y)) \dot{=} w, y \dot{=} z\}$$

$$(3) R = \{x \dot{=} g(y), y \dot{=} z, h(g(y)) \dot{=} w, y \dot{=} z\}$$

$$(4) R = \{x \dot{=} g(z), y \dot{=} z, h(g(z)) \dot{=} w, z \dot{=} z\}$$

$$(2)(1) R = \{x \dot{=} g(z), y \dot{=} z, w \dot{=} h(g(z))\}$$

$$(4) R = \{x \dot{=} g(z), y \dot{=} z, w \dot{=} h(g(z))\} \text{ cgu}$$

$$\Sigma = \{a : \rightarrow s, g : s \rightarrow s, h : s \rightarrow s, f : sss \rightarrow s\},$$

$$X = \{x, z, w\}$$

$$U = R = \{g(a) \dot{=} x, f(x, h(x), a) \dot{=} f(g(z), w, z)\}$$

$$(2) R = \{x \dot{=} g(a), f(x, h(x), a) \dot{=} f(g(z), w, z)\}$$

$$(4) R = \{x \dot{=} g(a), f(g(a), h(g(a)), a) \dot{=} f(g(z), w, z)\}$$

$$(3) R = \{x \dot{=} g(a), g(a) \dot{=} g(z), h(g(a)) \dot{=} w, a \dot{=} z\}$$

$$(3) R = \{x \dot{=} g(a), a \dot{=} z, h(g(a)) \dot{=} w, a \dot{=} z\}$$

$$(2) R = \{x \dot{=} g(a), z \dot{=} a, w \dot{=} h(g(a)), z \dot{=} a\}$$

$$(4) R = \{x \dot{=} g(a), z \dot{=} a, w \dot{=} h(g(a)), a \dot{=} a\}$$

$$(1) R = \{x \dot{=} g(a), z \dot{=} a, w \dot{=} h(g(a))\} \text{ cgu}$$

$$\Sigma = \{b : \rightarrow s, g : s \rightarrow s, h : s \rightarrow s, f : sss \rightarrow s\},$$

$$X = \{x, y, z\}$$

$$U = R = \{g(y) \dot{=} x, f(x, h(x), y) \dot{=} f(g(z), b, z)\}$$

$$(2) R = \{x \dot{=} g(y), f(x, h(x), y) \dot{=} f(g(z), b, z)\}$$

$$(4) R = \{x \dot{=} g(y), f(g(y), h(g(y)), y) \dot{=} f(g(z), b, z)\}$$

$$(3) R = \{x \dot{=} g(y), g(y) \dot{=} g(z), h(g(y)) \dot{=} b, y \dot{=} z\}$$

$$(3) R = \{x \dot{=} g(y), y \dot{=} z, h(g(y)) \dot{=} b, y \dot{=} z\}$$

$$(4) R = \{x \dot{=} g(z), y \dot{=} z, h(g(z)) \dot{=} b, z \dot{=} z\}$$

$$(3) R = \emptyset \text{ deoarece } \boxed{b \neq h}$$

$$\Sigma = \{g : s \rightarrow s, h : s \rightarrow s, f : sss \rightarrow s\},$$

$$X \doteq \{x, y, z, w\}$$

$$U = R = \{g(y) \doteq x, f(x, h(x), y) \doteq f(y, w, z)\}$$

$$(2) R = \{x \doteq g(y), f(x, h(x), y) \doteq f(y, w, z)\}$$

$$(4) R = \{x \doteq g(y), f(g(y), h(g(y)), y) \doteq f(y, w, z)\}$$

$$(3) R = \{x \doteq g(y), g(y) \doteq y, h(g(y)) \doteq w, y \doteq z\}$$

$$(2) R = \{x \doteq g(z), y \doteq g(y), h(g(y)) \doteq w, y \doteq z\}$$

$$(4) R = \emptyset \text{ deoarece } \boxed{y \in Var(g(y))}$$

Terminare

Algoritmul de unificare se termină.

Fie R problema de unificare. Notăm

$$n_{sol} := |\{x \in Var(R) \mid R = R' \cup \{x \doteq t\}, x \notin Var(R') \cup Var(t)\}|,$$

$$n_1 := |Var(R)| - n_{sol}$$

$$n_2 := \sum_{s \doteq t \in R} (|s| + |t|), \quad n_3 := |R|$$

Fiecare pas al algoritmului modifică n_1, n_2, n_3

	n_1	n_2	n_3
Șterge	\geq	$>$	
Orientează	\geq	$=$	$>$
Descompune	\geq	$>$	
Elimină	$>$		

Dacă la execuția unui pas (n_1, n_2, n_3) se schimbă în (n'_1, n'_2, n'_3) , atunci $(n_1, n_2, n_3) >_{lex} (n'_1, n'_2, n'_3)$. Algoritmul se termină deoarece (n_1, n_2, n_3) descrește în ordine lexicografică la fiecare pas.

Corectitudine

Dacă $R \Rightarrow T$ și $T \neq \emptyset$, atunci $Unif(R) = Unif(T)$.

Proprietatea este evident adevărată pentru **Ștergeși Orientează**.

Descompune. $R = R' \cup \{\sigma(e_1, \dots, e_n) \dot{=} \sigma(e'_1, \dots, e'_n)\}$

$$T = R' \cup \{e_1 \dot{=} e'_1, \dots, e_n \dot{=} e'_n\}$$

• oricare $\nu : X \rightarrow T_\Sigma(X)$ substituție

$$\nu(\sigma(e_1, \dots, e_n)) = \nu(\sigma(e'_1, \dots, e'_n)) \Leftrightarrow \nu(e_i) = \nu(e'_i) \text{ or. } i$$

Elimină: $R = R' \cup \{x \dot{=} t\}$,

$$T = \{x \dot{=} t\} \cup \{x \leftarrow t\}R', x \in Var(R') \setminus Var(t)$$

$$\nu \in Unif(R) \Leftrightarrow \nu \in Unif(R') \text{ și } \nu \in Unif(\{x \dot{=} t\})$$

• dacă $\nu \in Unif(\{x \dot{=} t\})$ atunci $\{x \leftarrow t\}; \nu = \nu$

$$\begin{aligned} \nu \in Unif(R) &\Leftrightarrow \{x \leftarrow t\}; \nu \in Unif(R') \text{ și } \nu \in Unif(\{x \dot{=} t\}) \\ &\Leftrightarrow \nu \in Unif(\{x \leftarrow t\}R') \text{ și } \nu \in Unif(\{x \dot{=} t\}) \\ &\Leftrightarrow \nu \in Unif(T) \end{aligned}$$

Completitudine

Dacă $U \Rightarrow \dots \Rightarrow R \Rightarrow \emptyset$ atunci R nu are soluții.

Distingem două cazuri.

- La pasul **Descompune**, există $\sigma(e_1, \dots, e_n) \doteq \tau(e'_1, \dots, e'_k) \in R$ cu $\sigma \neq \tau$. Termenii $\sigma(e_1, \dots, e_n)$ și $\tau(e'_1, \dots, e'_k)$ nu pot fi unificați deoarece încep cu caractere diferite.
- La pasul **Elimină**, există $x \doteq t \in R$ cu $x \in Var(t)$ și $t \neq x$. Dacă ν este o substituție, atunci $|\nu(t)| > |\nu(x)|$, deci x și t nu au unificator.

R conține o ecuație care nu poate fi unificată, deci R nu are unificator. Deoarece U și R au aceeași unificatori, rezultă ca U nu are unificatori.

Complexitate

■ Problema de unificare

$U = \{x_1 \doteq f(x_0, x_0), x_2 \doteq f(x_1, x_1), \dots, x_n \doteq f(x_{n-1}, x_{n-1})\}$
are cgu $R = \{x_1 \leftarrow f(x_0, x_0), x_2 \leftarrow f(f(x_0, x_0), f(x_0, x_0)), \dots\}.$

Complexitate

- Problema de unificare

$U = \{x_1 \doteq f(x_0, x_0), x_2 \doteq f(x_1, x_1), \dots, x_n \doteq f(x_{n-1}, x_{n-1})\}$
are cgu $R = \{x_1 \leftarrow f(x_0, x_0), x_2 \leftarrow f(f(x_0, x_0), f(x_0, x_0)), \dots\}$.

- La pasul **Elimină**, pentru a verifica că o variabilă x_i nu apare în membrul drept al ecuației (**occur check**) facem 2^i comparații.

Complexitate

- Problema de unificare

$U = \{x_1 \doteq f(x_0, x_0), x_2 \doteq f(x_1, x_1), \dots, x_n \doteq f(x_{n-1}, x_{n-1})\}$
are cgu $R = \{x_1 \leftarrow f(x_0, x_0), x_2 \leftarrow f(f(x_0, x_0), f(x_0, x_0)), \dots\}$.

- La pasul **Elimină**, pentru a verifica că o variabilă x_i nu apare în membrul drept al ecuației (**occur check**) facem 2^i comparații.

- Algoritmul de unificare prezentat anterior este exponențial. Complexitatea poate fi îmbunătățită printr-o reprezentare eficientă a termenilor.

📖 *K. Knight, Unification: A Multidisciplinary Survey, ACM Computing Surveys, Vol. 21, No. 1, 1989.*

CafeObj

```
{mod! MATCH{
[ Elem ]
op a : -> Elem
op _+_   : Elem Elem -> Elem
op _*_   : Elem Elem -> Elem
vars x y : Elem
eq x + (y * y) = x * y . }
```

```
MATCH> reduce (a + y) + (x * x ).
-- reduce in MATCH : ((a + y) + (x * x)):Elem
1>[1] rule: eq (x:Elem + (y:Elem * y))= (x * y)
      { x:Elem |-> (a + y:Elem), y:Elem |-> x:Elem }
1<[1]((a + y:Elem) + (x:Elem * x))-->((a + y:Elem)* x:Elem)
((a + y) * x):Elem
(0.000 sec for parse, 1 rewrites(0.000 sec), 3 matches)
```

Matching problem

- Fie p și t termeni cu variabile din X . Spunem că
 p **matches** t (t este o **instanță** a lui p)
dacă există o substituție $\nu : X \rightarrow T_{\Sigma}(X)$ a. î. $\nu(p) = t$.
 $p = x + (y * y), t = (a + y) + (x * x)$
 $\nu(x) := a + y, \nu(y) := x$

Matching problem

- Fie p și t termeni cu variabile din X . Spunem că

p **matches** t (t este o **instanță** a lui p)

dacă există o substituție $\nu : X \rightarrow T_{\Sigma}(X)$ a. î. $\nu(p) = t$.

$$p = x + (y * y), t = (a + y) + (x * x)$$

$$\nu(x) := a + y, \nu(y) := x$$

- Fie t' termenul obținut din t prin înlocuirea fiecărei variabile $x \in Var(T)$ cu o operație constantă c_x .

Substituția ν poate fi determinată aplicînd algoritmul de unificare ecuației $p \dot{=} t'$.

$$x + (y * y) \dot{=} (a + c_y) + (c_x * c_x)$$

$$\{x \dot{=} a + c_y, y \dot{=} c_x\}$$

Matching problem

- Fie p și t termeni cu variabile din X . Spunem că

p **matches** t (t este o **instanță** a lui p)

dacă există o substituție $\nu : X \rightarrow T_{\Sigma}(X)$ a. î. $\nu(p) = t$.

$$p = x + (y * y), t = (a + y) + (x * x)$$

$$\nu(x) := a + y, \nu(y) := x$$

- Fie t' termenul obținut din t prin înlocuirea fiecărei variabile $x \in Var(T)$ cu o operație constantă c_x .

Substituția ν poate fi determinată aplicînd algoritmul de unificare ecuației $p \dot{=} t'$.

$$x + (y * y) \dot{=} (a + c_y) + (c_x * c_x)$$

$$\{x \dot{=} a + c_y, y \dot{=} c_x\}$$

- O problemă de *matching* poate fi rezolvată prin unificare.