

Sémantique dénotationnelle

Luigi Santocanale

Laboratoire d'Informatique Fondamentale,
Centre de Mathématiques et Informatique,
39, rue Joliot-Curie - F-13453 Marseille

À chaque expression/commande du langage on associe
une relation, son interprétation :

$$\| \cdot \|_{\mathcal{A}exp} : \mathcal{A}exp \longrightarrow \mathcal{P}(\mathcal{S} \times \mathcal{N})$$

$$\| \cdot \|_{\mathcal{B}exp} : \mathcal{B}exp \longrightarrow \mathcal{P}(\mathcal{S} \times \{1, 0\})$$

$$\| \cdot \|_{\mathcal{C}om} : \mathcal{C}om \longrightarrow \mathcal{P}(\mathcal{S} \times \mathcal{S})$$

Plan

- 1 Sémantique dénotationnelle du langage IML
 - Définition
 - Sémantique dénotationnelle vs. sémantique opérationnelle
- 2 Treillis complets et le théorème de Tarski
 - Treillis complets
 - Les théorèmes de point fixe
- 3 Sémantique abstraite : introduction à la théorie des domaines

Interprétation des expressions arithmétiques

$$\| \hat{n} \| = \{ (\sigma, n) \mid \sigma \in \mathcal{S} \}$$

$$\| X \| = \{ (\sigma, \sigma(X)) \mid \sigma \in \mathcal{S} \}$$

$$\| a_1 + a_2 \| = \{ (\sigma, n_1 + n_2) \mid (\sigma, n_1) \in \| a_1 \|, (\sigma, n_2) \in \| a_2 \| \}$$

$$\| a_1 - a_2 \| = \{ (\sigma, n_1 - n_2) \mid (\sigma, n_1) \in \| a_1 \|, (\sigma, n_2) \in \| a_2 \| \}$$

$$\| a_1 * a_2 \| = \{ (\sigma, n_1 n_2) \mid (\sigma, n_1) \in \| a_1 \|, (\sigma, n_2) \in \| a_2 \| \}$$

Interprétation des expressions booléennes

$$\| \text{true} \| = \{ (\sigma, 1) \mid \sigma \in \mathcal{S} \}$$

$$\| \text{false} \| = \dots$$

$$\| \text{not } b \| = \{ (\sigma, 1) \mid (\sigma, 0) \in \| b \| \} \cup \{ (\sigma, 0) \mid (\sigma, 1) \in \| b \| \}$$

$$\| b_1 \text{ and } b_2 \| = \dots$$

$$\| b_1 \text{ or } b_2 \| = \dots$$

$$\| b_1 \leq b_2 \| = \{ (\sigma, 1) \mid \exists n_1 \leq n_2 \text{ t.c. } (\sigma, n_1) \in \| b_1 \| \text{ et } (\sigma, n_2) \in \| b_2 \| \}$$

$$\cup \{ (\sigma, 0) \mid \exists n_1 > n_2 \text{ t.c. } (\sigma, n_1) \in \| b_1 \| \text{ et } (\sigma, n_2) \in \| b_2 \| \}$$

$$\| b_1 = b_2 \| = \dots$$

Exercice : compléter la définition.

Interprétation des commandes

$$\| \text{skip} \| = \{ (\sigma, \sigma) \mid \sigma \in \mathcal{S} \}$$

$$\| c_1; c_2 \| = \{ (\sigma_1, \sigma_3) \mid \exists \sigma_2 \text{ t.q. } (\sigma_1, \sigma_2) \in \| c_1 \| \text{ et } (\sigma_2, \sigma_3) \in \| c_2 \| \}$$

$$\| X := a \| = \{ (\sigma, \sigma[n/X]) \mid (\sigma, n) \in \| a \|_{\mathcal{A}_{exp}} \}$$

$$\| \text{if } (b) \text{ then } c_1 \text{ else } c_2 \|$$

$$= \{ (\sigma_1, \sigma_2) \in \| c_1 \| \mid (\sigma_1, 1) \in \| b \|_{\mathcal{B}_{exp}} \}$$

$$\cup \{ (\sigma_1, \sigma_2) \in \| c_2 \| \mid (\sigma_1, 0) \in \| b \|_{\mathcal{B}_{exp}} \}$$

Interprétation du commande while ... do ...

Soit $b \in \mathcal{B}_{exp}$ et $comm \in \mathcal{C}om$.

Posons :

$$f(R) = \{ (\sigma, \sigma) \mid (\sigma, 0) \in \| b \|_{\mathcal{B}_{exp}} \}$$

$$\cup \{ (\sigma_1, \sigma_3) \mid (\sigma, 1) \in \| b \|_{\mathcal{B}_{exp}} \text{ et}$$

$$\exists \sigma_2 \text{ t.q. } (\sigma_1, \sigma_2) \in \| comm \| \text{ et } (\sigma_2, \sigma_3) \in R \}$$

Observons :

- $f : \mathcal{P}(\mathcal{S} \times \mathcal{S}) \rightarrow \mathcal{P}(\mathcal{S} \times \mathcal{S})$ est une fonction croissante,
- si $c \in \mathcal{C}om$, alors

$$f(\| c \|) = \| \text{if } (b) \text{ then } comm ; c \text{ else skip} \| \quad (1)$$

Interprétation de while ... do ... (II)

Définissons

$$\| \text{while } b \text{ do } comm \| = \text{fix}(f)$$

$$= \text{plus petit point fixe de } f.$$

De cette façon :

$$\| \text{while } b \text{ do } comm \|$$

$$= f(\| \text{while } b \text{ do } comm \|)$$

$$= \| \text{if } (b) \text{ then } comm ; \text{while } b \text{ do } comm \text{ else skip} \|$$

à cause de (1).

Équivalences

Définition

$$a \equiv_{\mathcal{A}exp} a' \text{ ssi } \| a \|_{\mathcal{A}exp} = \| a' \|_{\mathcal{A}exp}$$

$$b \equiv_{\mathcal{B}exp} b' \text{ ssi } \| b \|_{\mathcal{B}exp} = \| b' \|_{\mathcal{A}exp}$$

$$c \equiv_{\mathcal{C}om} c' \text{ ssi } \| c \|_{\mathcal{C}om} = \| c' \|_{\mathcal{C}om}$$

Exemple : on vient de montrer que

while b do comm

$$\equiv_{\mathcal{C}om} \textit{if } b \textit{ then comm; while } b \textit{ do comm else skip}$$



Rapport entre les sémantiques

- sémantique opérationnelle,
- sémantique dénotationnelle,
- sémantique axiomatique,
- sémantique ...

Quelle cohérence ?

Proposition

Le sémantique opérationnelle est la même que la sémantique dénotationnelle :

$$c \sim c' \text{ ssi } c \equiv c'.$$



Rapport entre les sémantiques (II)

Plus précisément :

Proposition

$$\forall a \in \mathcal{A}exp \quad \| a \|_{\mathcal{A}exp} = \{ (\sigma, n) \mid (a, \sigma) \rightarrow_{\mathcal{A}exp} n \}$$

$$\forall b \in \mathcal{B}exp \quad \| b \|_{\mathcal{B}exp} = \{ (\sigma, v) \mid (b, \sigma) \rightarrow_{\mathcal{B}exp} v \}$$

$$\forall c \in \mathcal{C}om \quad \| c \|_{\mathcal{C}om} = \{ (\sigma, \sigma') \mid (c, \sigma) \rightarrow_{\mathcal{C}om} \sigma' \}.$$



Preuve : $\| a \|_{\mathcal{A}exp} = \{ (\sigma, n) \mid (a, \sigma) \rightarrow_{\mathcal{A}exp} n \}$

Par induction sur la structure de a .

$$(\sigma, n) \in \| \hat{m} \|_{\mathcal{A}exp} \text{ ssi } m = n \\ \text{ssi } (\hat{m}, \sigma) \rightarrow_{\mathcal{A}exp} n$$

$$(\sigma, n) \in \| X \|_{\mathcal{A}exp} \text{ ssi } \sigma(X) = n \\ \text{ssi } (X, \sigma) \rightarrow_{\mathcal{A}exp} n$$

Supposons la propriété vraie pour a_0, a_1 .

$$(\sigma, n) \in \| a_0 + a_1 \|_{\mathcal{A}exp} \\ \text{ssi } \exists n_0, n_1 \ (\sigma, n_0) \in \| a_0 \|_{\mathcal{A}exp}, \ (\sigma, n_1) \in \| a_1 \|_{\mathcal{A}exp} \\ \text{et } n = n_0 + n_1 \\ \text{ssi } \exists n_0, n_1 \ (a_0, \sigma) \rightarrow_{\mathcal{A}exp} n_0, \ (a_1, \sigma) \rightarrow_{\mathcal{A}exp} n_1, \\ \text{et } n = n_0 + n_1 \\ \text{ssi } (a_0 + a_1, \sigma) \rightarrow_{\mathcal{A}exp} n.$$



Suite preuve

On raisonne de façon semblable pour $a_0 - a_1$ et $a_0 * a_1$:

$$\begin{aligned} (\sigma, n) \in \| a_0 - a_1 \|_{\mathcal{A}exp} \\ \text{ssi } \exists n_0, n_1 \ (\sigma, n_0) \in \| a_0 \|_{\mathcal{A}exp}, \ (\sigma, n_1) \in \| a_1 \|_{\mathcal{A}exp} \\ \text{et } n = n_0 - n_1 \\ \text{ssi } \exists n_0, n_1 \ (a_0, \sigma) \rightarrow_{\mathcal{A}exp} n_0, \ (a_1, \sigma) \rightarrow_{\mathcal{A}exp} n_1, \\ \text{et } n = n_0 - n_1 \\ \text{ssi } (a_0 - a_1, \sigma) \rightarrow_{\mathcal{A}exp} n. \end{aligned}$$

□

Exercice :
montrer le résultat analogue

$$\| b \|_{\mathcal{B}exp} = \{ (\sigma, \nu) \mid (b, \sigma) \rightarrow_{\mathcal{B}exp} \nu \}$$

pour les expression Booléennes.

Preuve : $\| c \|_{\mathcal{C}om} = \{ (\sigma, \sigma') \mid (c, \sigma) \rightarrow_{\mathcal{C}om} \sigma' \}$

Par induction sur la structure de c .

$$\begin{aligned} (\sigma, \sigma') \in \| skip \|_{\mathcal{C}om} \text{ ssi } \sigma = \sigma' \\ \text{ssi } (skip, \sigma) \rightarrow_{\mathcal{C}om} \sigma' \end{aligned}$$

$$\begin{aligned} (\sigma, \sigma') \in \| X := a \|_{\mathcal{C}om} \text{ ssi } \sigma' = \sigma[n/X] \text{ et } (\sigma, n) \in \| a \|_{\mathcal{A}exp} \\ \text{ssi } \sigma' = \sigma[n/X] \text{ et } (a, \sigma) \rightarrow_{\mathcal{A}exp} n \\ \text{ssi } (X, \sigma) \rightarrow_{\mathcal{C}om} \sigma' \end{aligned}$$

Preuve : $\| c \|_{\mathcal{C}om} = \{ (\sigma, \sigma') \mid (c, \sigma) \rightarrow_{\mathcal{C}om} \sigma' \}$

Hypothèse d'induction : la propriété est vrai pour c, c_0, c_1 .

$$\begin{aligned} (\sigma_0, \sigma_2) \in \| c_0; c_1 \|_{\mathcal{C}om} \\ \text{ssi } \exists \sigma_1 \ (\sigma_0, \sigma_1) \in \| c_0 \|_{\mathcal{C}om} \text{ et } (\sigma_1, \sigma_2) \in \| c_1 \|_{\mathcal{C}om} \\ \text{ssi } \exists \sigma_1 \ (c_0, \sigma_0) \rightarrow_{\mathcal{C}om} \sigma_1 \text{ et } (c_1, \sigma_1) \rightarrow_{\mathcal{C}om} \sigma_2 \\ \text{ssi } (c_0; c_1, \sigma_0) \rightarrow_{\mathcal{C}om} \sigma_2 \end{aligned}$$

$$\begin{aligned} (\sigma_0, \sigma_1) \in \| \text{if } b \text{ then } c_0 \text{ else } c_1 \|_{\mathcal{C}om} \\ \text{ssi } (\sigma_0, 1) \in \| b \|_{\mathcal{B}exp} \text{ et } (\sigma_0, \sigma_1) \in \| c_0 \|_{\mathcal{C}om} \text{ ou} \\ (\sigma_0, 0) \in \| b \|_{\mathcal{B}exp} \text{ et } (\sigma_0, \sigma_1) \in \| c_1 \|_{\mathcal{C}om} \\ \text{ssi } (b, \sigma_0) \rightarrow_{\mathcal{B}exp} 1 \text{ et } (c_0, \sigma_0) \rightarrow_{\mathcal{C}om} \sigma_1 \text{ ou} \\ (b, \sigma_0) \rightarrow_{\mathcal{B}exp} 0 \text{ et } (c_1, \sigma_0) \rightarrow_{\mathcal{C}om} \sigma_1 \\ \text{ssi } (\text{if } b \text{ then } c_0 \text{ else } c_1, \sigma_0) \rightarrow_{\mathcal{C}om} \sigma_1. \end{aligned}$$

Preuve : $\| c \|_{\mathcal{C}om} = \{ (\sigma, \sigma') \mid (c, \sigma) \rightarrow_{\mathcal{C}om} \sigma' \}$

Rappel :

$$\| \text{while } b \text{ do } c \|_{\mathcal{C}om} = \text{fix}(f)$$

où

$$\begin{aligned} f(X) = \{ (\sigma, \sigma) \mid (\sigma, 0) \in \| b \|_{\mathcal{B}exp} \} \\ \cup \{ (\sigma_1, \sigma_3) \mid (\sigma_1, 1) \in \| b \|_{\mathcal{B}exp} \text{ et} \\ \exists \sigma_2 \text{ t.q. } (\sigma_1, \sigma_2) \in \| c \|_{\mathcal{C}om} \text{ et } (\sigma_2, \sigma_3) \in X \} \end{aligned}$$

On a

$$f(X) = f_R(X)$$

où R est le système de règles suivant :

$$\begin{cases} \emptyset / (\sigma, \sigma) & (\sigma, 0) \in \| b \|_{\mathcal{B}exp} \\ (\sigma_2, \sigma_3) / (\sigma_1, \sigma_3) & (\sigma_1, 1) \in \| b \|_{\mathcal{B}exp} \text{ et } (\sigma_1, \sigma_2) \in \| c \|_{\mathcal{C}om} \end{cases}$$

et donc $\text{fix}(f) = I_R$.

Sémantique dénotationnelle du langage IML
Treillis complets et le théorème de Tarski
Sémantique abstraite : introduction à la théorie des domaines

Définition
Sémantique dénotationnelle vs. sémantique opérationnelle

Preuve : $\|c\|_{Com} = \{(\sigma, \sigma') \mid (c, \sigma) \rightarrow_{Com} \sigma'\}$

Considérons maintenant l'ensemble

$$A = \{(\sigma_1, \sigma_3) \mid (while\ b\ do\ c, \sigma_1) \rightarrow \sigma_3\}$$

Obtenu à l'aide du système de règles :

$$\frac{(\sigma_1, 0) \in \|b\|_{Bexp}}{(b, \sigma_1) \rightarrow 0}$$

$$\frac{(\sigma_1, 1) \in \|b\|_{Bexp} \quad (\sigma_1, \sigma_2) \in \|c\|_{Com}}{(b, \sigma_1) \rightarrow 1 \quad (c, \sigma_1) \rightarrow \sigma_2 \quad (while\ b\ do\ c, \sigma_2) \rightarrow \sigma_3}$$

Luigi Santocanale Sémantique dénotationnelle 19

Sémantique dénotationnelle du langage IML
Treillis complets et le théorème de Tarski
Sémantique abstraite : introduction à la théorie des domaines

Définition
Sémantique dénotationnelle vs. sémantique opérationnelle

Preuve : $\|c\|_{Com} = \{(\sigma, \sigma') \mid (c, \sigma) \rightarrow_{Com} \sigma'\}$

Formulation concise des mêmes règles :

$$\frac{}{(\dots, \sigma_1) \rightarrow \sigma_3} (\sigma_1, 0) \in \|b\|_{Bexp}$$

$$\frac{(\dots, \sigma_2) \rightarrow \sigma_3}{(\dots, \sigma_1) \rightarrow \sigma_3} (\sigma_1, 1) \in \|b\|_{Bexp} \text{ et } (\sigma_1, \sigma_2) \in \|c\|_{Com}$$

c.-à-d. :

Pour montrer que $(\sigma_1, \sigma_3) \in A$
il faut avoir un arbre de dérivation étiquette par les règles de R .

On a donc :

$$A = I_R = fix(f) = \|while\ b\ do\ c\|_{Com}.$$

Luigi Santocanale Sémantique dénotationnelle 20

Sémantique dénotationnelle du langage IML
Treillis complets et le théorème de Tarski
Sémantique abstraite : introduction à la théorie des domaines

Treillis complets
Les théorèmes de point fixe

Ensembles ordonnés

Définition

Un ensemble ordonné est un couple (P, \leq) t.q. $\leq \subseteq P \times P$ est :

- réflexive : $x \leq x, \forall x \in P$,
- transitive : $x \leq y$ et $y \leq z$ implique $x \leq z, \forall x, y, z \in P$,
- antisymétrique : $x \leq y$ et $y \leq x$ implique $x = y, \forall x, y \in P$

Définition

Soit $X \subseteq P$ et $p \in P$. p est un *plus petit majorant* de X s'il satisfait

$$\forall x \in X \ x \leq y \text{ ssi } p \leq y$$

On écrit alors $p = \bigvee X$. p est un *plus grand minorant* ssi ... alors $p = \bigwedge X$.

Luigi Santocanale Sémantique dénotationnelle 22

Sémantique dénotationnelle du langage IML
Treillis complets et le théorème de Tarski
Sémantique abstraite : introduction à la théorie des domaines

Treillis complets
Les théorèmes de point fixe

Treillis complets

Remarque : un plus petit (grand) majorant (minorant) est unique.

Définition

Un ensemble ordonné (P, \leq) est un treillis complet ssi pour tout sousensemble $X \subseteq P$ il existe $\bigvee X$.

Lemma

Un treillis complet est aussi « co-complet » : $\bigwedge X$ existe pour tout $X \subseteq P$.

Preuve : il suffit d'observer que

$$\bigwedge X = \bigvee \{y \mid \forall x \in X \ x \leq y\}.$$

Luigi Santocanale Sémantique dénotationnelle 23

Théorème de Tarski

Théorème de Tarski

Soit (P, \leq) un treillis complet. Alors toute fonction monotone $f : P \rightarrow P$ possède un point fixe.

Théorème de Davis

Soit (P, \leq) un ensemble ordonné tel que toute fonction monotone $f : P \rightarrow P$ possède un point fixe. Alors (P, \leq) est un treillis complet.

Preuve du théorème de Tarski

Posons

$$\text{Préf}_f = \{x \mid f(x) \leq x\}$$

Le plus petit point préfixe $\mu.f$:

$$\begin{aligned} \mu.f &\in \text{Préf}_f \\ x \in \text{Préf}_f &\text{ implique } \mu.f \leq x. \end{aligned}$$

Preuve du théorème de Tarski

Lemma

$\mu.f$, s'il existe, est bien un point fixe et donc il est le plus petit point fixe.

Car

$$\begin{aligned} f(\mu.f) &\leq \mu.f && \text{implique} \\ f(f(\mu.f)) &\leq f(\mu.f) && \text{c.-à-d.} \\ f(\mu.f) &\in \text{Préf}_f && \text{implique} \\ \mu.f &\leq f(\mu.f) && \end{aligned}$$

Preuve du théorème de Tarski

Lemma

Dans un treillis complet, l'élément

$$\bigwedge \text{Préf}_f$$

est le plus petit point préfixe de f , et donc le plus petit point fixe de f .

Car si $x \in \text{Préf}_f$ alors $\bigwedge \text{Préf}_f \leq x$, par conséquent :

$$\begin{aligned} f(\bigwedge \text{Préf}_f) &\leq f(x) && f \text{ croissante} \\ &\leq x && x \in \text{Préf}_f \end{aligned}$$

On obtient $f(\bigwedge \text{Préf}_f) \leq \bigwedge \text{Préf}_f$.

Théorème de Tarski-Knaster, Park

Soit (P, \leq) un treillis complet et $f : P \rightarrow P$ une fonction monotone.

Posons

$$\begin{aligned} \perp &= \bigvee \emptyset && \text{Remarque : } \perp \leq y \text{ pour tout } y \in P \\ f^0(\perp) &= \perp \\ f^{\alpha+1}(\perp) &= f(f^\alpha(\perp)) \\ f^\alpha(\perp) &= \bigvee_{\beta < \alpha} f^\beta(\perp) && \alpha \text{ ordinal limite.} \end{aligned}$$

On a

$$\bigwedge Pref_f = f^\alpha(\perp) \quad \text{pour quelque ordinal } \alpha$$

CPOs

Définition

Un CPO est un ensemble ordonné (P, \leq) tel que

- $\perp \in P$,
- toute chaîne dénombrable ascendante

$$x_0 \leq x_1 \leq \dots x_n \leq \dots$$

possède un p.p.borne supérieure $\bigvee \{x_i \mid i \geq 0\}$.

Exemple :

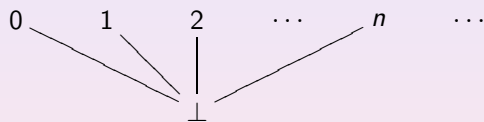
Soit $A \rightarrow B$ l'ensemble des fonctions partielles de A vers B .
 On pose $f \leq g$ ssi $f(a)$ est défini implique $g(a)$ est défini aussi.

Schéma d'interprétation : les types

La sémantique d'un type est un CPO.

Exemples :

$\|nat\|$ est le CPO :



$\|bool\|$ est le CPO :

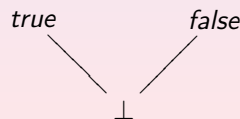


Schéma d'interprétation : les expressions

La sémantique d'une expression

$$e : t_1 \rightarrow t_2$$

est une fonction continue

$$\|e\| : \|t_1\| \rightarrow \|t_2\|$$

Définition

Soient P_0, P_1 des CPOs. Une fonction monotone est *continue* ssi pour toute chaîne

$$x_0 \leq x_1 \leq \dots x_n \leq \dots$$

on a

$$f(\bigvee \{x_i \mid i \geq 0\}) = \bigvee \{f(x_i) \mid i \geq 0\}.$$

Fonctions continues ?

On s'intéresse aux fonctions continues :

- G est l'interprétation d'une commande,
- $G(x)$ est le résultat des calculs avec en entrée x ,
- si x est un objet infini (ex: fonction, liste infinie), alors $x = \bigvee_{i \geq 0} x_i$, avec les x_i approximations finies de x .

La relation de continuité

$$G\left(\bigvee_{i \geq 0} x_i\right) = \bigvee_{i \geq 0} G(x_i)$$

s'interprète selon le principe suivant :

On peut approcher le résultat de G sur une entrée infinie x en calculant le résultat de G sur les approximations finies de x .



Un exemple

Comparer les fonctions G et H suivantes :

$$Gf = \text{fun } n \rightarrow f(n + 1)$$

$$Hf = \begin{cases} \text{fun } n \rightarrow 0 & \text{si } f(n) = 0 \text{ pour tout } n, \\ \text{fun } n \rightarrow 0 & \text{sinon.} \end{cases}$$

La fonction G n'est pas continue.

On ne peut pas définir la fonction G en OCaml.



Quelques résultats

Proposition

Soient P_0, P_1 deux CPOs, alors l'espace $P_1^{P_0}$ des fonctions continues de P_0 vers P_1 est lui-même un CPO.

De façon que :

$$\| t1 \rightarrow t2 \| = \| t2 \|^{t1}$$

Proposition

Si P est un CPO, alors $\bigvee_{n \geq 0} f^n(\perp)$ est le plus petit point fixe de f .

De façon que :

$$\| \text{let rec expr}(x) \| = \text{fix}(\| \text{expr}(x) \|).$$

