

Sémantique axiomatique

Frédéric Boulanger

CentraleSupélec

01/2020



CentraleSupélec

Rappel : sémantique opérationnelle

- La sémantique opérationnelle à grands pas établit une relation entre des paires (instruction, état) et des états.
- $(instr, s) \rightsquigarrow s'$ signifie qu'en exécutant instr dans l'état s, on peut arriver dans l'état s' .
- Cette sémantique permet de dérouler la séquence d'états par lesquels passe la machine d'exécution lorsqu'on exécute un programme.



Sémantique dénotationnelle

Ce style de sémantique associe à chaque instruction une relation entre états.

On définit une fonction $\llbracket _ \rrbracket$ telle que :

$$\llbracket \text{instr} \rrbracket = \{(\text{pre}, \text{post}) \mid (\text{pre}, \text{instr}) \rightsquigarrow \text{post}\}$$

Cette sémantique est compositionnelle : la dénotation d'une suite d'instructions s'obtient par composition de leurs dénotations.



Sémantique axiomatique

On cherche à définir la manière dont l'exécution d'un programme affecte la validité de propositions portant sur l'état de la machine d'exécution.

Hoare : sémantique pré/post

$$\models \{\varphi\}P\{\psi\}$$

Si φ est vraie dans un état s ,
alors ψ est vraie dans tout état s' tel que $(s, P) \rightsquigarrow s'$.

Sémantique axiomatique

On cherche à définir la manière dont l'exécution d'un programme affecte la validité de propositions portant sur l'état de la machine d'exécution.

Hoare : sémantique pré/post

$$\models \{\varphi\}P\{\psi\}$$

Si φ est vraie dans un état s ,
alors ψ est vraie dans tout état s' tel que $(s, P) \rightsquigarrow s'$.

Substitution

Pour traiter l'affectation, il faut pouvoir substituer une expression à une variable :

$$P[\text{with } \textit{expr} \text{ for } x]$$

est la proposition P dans laquelle on a substitué $\textit{expr}$ à x .

Règles sémantiques

On indique pour chaque instruction comment elle affecte les propositions :

$\text{Nop} \vdash \{P\} \text{Nop} \{P\}$

$\text{VDecl} \vdash \{P[\text{with } (N\ 0) \text{ for } v]\} \text{var } v \{P\}$

...

Dans cette dernière règle, on anticipe l'effet de la déclaration en substituant 0 à v dans P .

Validité et dérivabilité

- $\models \{\varphi\}P\{\psi\}$: le triplet est valide
- $\vdash \{\varphi\}P\{\psi\}$: le triplet est dérivable selon les règles sémantiques

Sémantique axiomatique de Niklaus



Exemple : calcul du PGCD par soustractions successives

```
while (x <> y) {  
  if x < y {  
    y := y - x  
  } else {  
    x := x - y  
  }  
}
```

Preuve de l'algorithme



Correction et complétude

Correction (*Soundness*)

La sémantique axiomatique est correcte si toute dérivation est valide :

$$\vdash \{P\} \text{ instr } \{Q\} \implies \models \{P\} \text{ instr } \{Q\}$$

Complétude (*Completeness*)

Elle est complète si tout triplet valide peut être dérivé :

$$\models \{P\} \text{ instr } \{Q\} \implies \vdash \{P\} \text{ instr } \{Q\}$$

Preuve de la correction et de la complétude

