

Théorème de Scott et application au théorème de Rice

Julie Parreaux

2018-2019

Référence du développement : Hindley [1, p.91]

Leçons où on présente le développement : 914 (Décidabilité, Indécidabilité) ; 929 (Lambda-calcul).

1 Introduction

Le λ -calcul est un modèle de calcul qui vient toucher la frontière de l'indécidabilité. Le théorème de Scott nous confirme que séparer deux ensemble de λ -termes (deux ensembles de fonctions possédant leurs propres caractéristiques) est très indécidable. Ce résultat nous permet notamment de déduire le théorème de Rice. Par leur proximité avec les langages de programmation fonctionnels. Nous venons de prouver que la correction d'un programme est impossible à implémenter par un programme purement fonctionnel (le résultat est vrai pour tous les langages bien évident).

Remarques sur le développement

1. Preuve du lemme : $F \llbracket gd(X) \rrbracket = X$.
2. Preuve du théorème pour les ensembles récursivement énumérables.
3. Application au théorème de Rice.

2 Preuve du théorème de Scott

On note Λ un ensemble de λ -termes non vide clos par β -équivalence. On se donne un codage des λ -termes : $gd : \Lambda \rightarrow \mathbb{N}$ injectif.

Théorème (Théorème de Scott). Soient A et B deux sous-ensembles de Λ . Alors A et B ne sont pas récursivement énumérable.

Définition. Soient A et B deux sous-ensembles de Λ . A et B sont récursivement séparable si et seulement s'il existe un ensemble récursif (son indicatrice est récursive) tel que $(A \subset C) \wedge (B \cap C = \emptyset)$.

Lemme. Pour tout $F \in \Lambda$, il existe X tel que $F \llbracket gd(X) \rrbracket = X$.

Démonstration. On définit

$$\begin{aligned} \text{Ap} \llbracket gd(M) \rrbracket (\llbracket gd(N) \rrbracket) &= \llbracket gd(MN) \rrbracket && \text{Interprété comme l'interprétation de } M \text{ appliqué en } N \\ \text{Num} \llbracket gd(n) \rrbracket &= \llbracket gd(\llbracket gd(n) \rrbracket) \rrbracket && \text{Interprété comme la traduction d'un entier} \end{aligned}$$

On pose $W = \lambda x.F(\text{Ap}x(\text{Num}x))$ et $X = W \llbracket gd(W) \rrbracket$. Alors,

$$\begin{aligned} X &=_{\beta} F(\text{Ap} \llbracket gd(W) \rrbracket (\text{Num} \llbracket gd(W) \rrbracket)) \\ &= F(\text{Ap} \llbracket gd(W) \rrbracket (\llbracket gd(\llbracket gd(W) \rrbracket) \rrbracket)) && \text{(En remplaçant Num)} \\ &= F(\llbracket gd(W \llbracket gd(W) \rrbracket) \rrbracket) && \text{(En remplaçant Ap)} \\ &= F \llbracket X \rrbracket && \text{(Définition de X)} \end{aligned}$$

□

Démonstration du théorème. Soit $M_0 \in A$ et $M_1 \in B$. Raisonnons par l'absurde et supposons que C un ensemble récursif séparant A et B . La fonction caractéristique de $\text{gd}(C)$ est récursive et définie par F comme :

$$\begin{aligned} M \in C &\Rightarrow F[\text{gd}(M)] = \llbracket 0 \rrbracket \\ M \notin C &\Rightarrow F[\text{gd}(M)] = \llbracket 1 \rrbracket \end{aligned}$$

On pose $G = \lambda x(\text{Zero}(Fx))M_1M_0$ (définie comme la conditionnelle sur la fonction caractéristique et être dans C donne le résultat inverse). Alors, si $M \in C$, on obtient

$$G[\text{gd}(M)] =_{\beta} \text{Zero}(F[\text{gd}(M)])M_1M_0 =_{\beta} \text{Zero}(\llbracket 0 \rrbracket)M_1M_0 =_{\beta} M_1$$

En raisonnant de même, si $M \in C$, on obtient $G[\text{gd}(M)] =_{\beta} M_0$.

Par le lemme, il existe $X \in \Lambda$ tel que $G[\text{gd}(X)] =_{\beta} X$. On en déduit que si

$$\begin{aligned} X \in C &\Rightarrow X = G[\text{gd}(X)] = M_1 \quad \text{et} \quad X \notin C \quad \text{car } B \cap C = \emptyset \\ X \notin C &\Rightarrow X = G[\text{gd}(X)] = M_0 \quad \text{et} \quad X \in C \quad \text{car } A \subset C \end{aligned}$$

Contradiction. □

Définition. A un sous-ensemble de Λ est non trivial si $A \neq \emptyset$ et $A \neq \Lambda$.

Corollaire (Théorème de Rice). *Soit A un sous-ensemble de Λ non trivial. Alors, A n'est pas récursif.*

Démonstration. Soit A un sous-ensemble de Λ non trivial. On applique le premier point du théorème à A et à A^c ; Donc pour tout ensemble récursif C tel que $A \subseteq C$, $C \cap A^c \neq \emptyset$. En particulier $C \neq A$ et A n'est pas récursif. □

Références

[1] J. P. Seldin J. R. Hindley. *Lambda-Calculus and Combinators : An Introduction*. Cambridge University Press, 2 edition, 2008.