Propriétés de programmes récursifs

I. Préambule mathématique

Un ingrédient important pour l'analyse de programmes récursifs est le principe d'induction appliqué à des ensembles ordonnés par un ordre bien fondé, qui généralise le principe des démonstrations par récurrence.

1. Ordre bien fondé

Commençons par quelques rappels.

- ⋆ Une *relation d'ordre* $\preceq$ définie sur un ensemble E est une relation binaire sur E qui vérifie les propriétés suivantes :

Réflexivité	$\forall x \in E,$	$x \preceq x$
Anti-symétrie	$\forall x, y \in E,$	$(x \preceq y \ \textbf{et} \ y \preceq x) \implies x = y$
Transivitivé	$\forall x, y, z \in E,$	$(x \preceq y \ \textbf{et} \ y \preceq z) \implies x \preceq z$

  On dit alors que $(E, \preceq)$ est un ensemble bien ordonné, et on notera $x \prec y$ pour $(x \preceq y \ \textbf{et} \ x \neq y)$.
- ⋆ Un élément x d'un ensemble ordonné $(E, \preceq)$ est
  - *minimal* si $\{y \in E \mid y \prec x\}$ est vide ou, de façon équivalente,
$$\forall y \in E, y \preceq x \implies y = x\,;$$
  - *le plus petit élément* de E si $\forall y \in E, x \preceq y$. Bien sûr, un ensemble ordonné admet au plus un plus petit élément.

  On définit de façon duale les notions d'élément maximal et de plus grand élément.
- ⋆ Une relation d'ordre $\preceq$ sur E est *totale* si de plus :
$$\forall x, y \in E, (x \preceq y \ \textbf{ou} \ y \preceq x)$$

Définition (Ordre bien fondé)

- ⋆ Une relation d'ordre $\preceq$ sur un ensemble E est **bien fondée** s'il n'existe pas de suite infinie $(x_n) \in E^{\mathbf{N}}$ strictement décroissante. De façon équivalente, tout partie non vide de E admet un élément minimal.
- ⋆ Une relation d'ordre bien fondée et totale est appelée un **bon ordre**.

La notion d'ordre bien fondé assure que l'on aura jamais de *descente infinie*. Informatiquement, la présence d'un ordre bien fondé nous permettra de montrer que sous de bonnes conditions, une fonction récursive termine.

Exemple

- ⋆ La relation d'ordre usuel sur $\mathbf{N}$ est un ordre bien fondé, et même un bon ordre, puisqu'il est total.
- ⋆ Les relations d'ordre usuel sur $\mathbf{Z}$, $\mathbf{Q}^+$ et sur $[0, 1]$ ne sont pas bien fondées.

Quelques constructions Voici quelques ordres bien fondés sur $\mathbf{N}$ (que l'on pourra généraliser).

- ⋆ L'ordre produit, défini sur $\mathbf{N}^2$ par :
$$(a, b) \preceq (c, d) \iff (a \leq c \ \textbf{et} \ b \leq d)$$
  est un ordre bien fondé, mais ce n'est pas un bon ordre.
- ⋆ L'ordre lexicographique, défini sur $\mathbf{N}^2$ par :
$$(a, b) \preceq (c, d) \iff (a < c \ \textbf{ou} \ (a = c \ \textbf{et} \ b \leq d))$$
  est un bon ordre.

Remarques

- ⋆ Pour tout ensemble ordonné $(E, \preceq)$ et tout $x \in E$, si l'ensemble $\{y \in E \mid y \preceq x\}$ est fini, alors l'ordre est bien fondé.
- ⋆ Cette condition de finitude n'est pas nécessaire. Pour l'ordre lexicographique, par exemple, l'ensemble $\{(a, b) \in \mathbf{N}^2 \mid (a, b) \preceq (2, 10)\}$ est infini.

Exercice 1 Prouver que l'ordre lexicographique sur $\mathbf{N}^2$ est bien un bon ordre.

Exercice 2 Prouver que l'ordre lexicographique gradué, défini sur $\mathbf{N}^2$ par :

$$\forall a, b, c, d \in \mathbf{N}, (a, b) \preceq (c, d) \iff (a + b < c + d \ \textbf{ou} \ (a + b = c + d \ \textbf{et} \ a \leq c))$$

est un bon ordre.

Exercice 3 Montrer qu'une relation d'ordre $\preceq$ sur un ensemble E est un bon ordre si et seulement si toute partie non vide de E admet un plus petit élément.

2. Principe d'induction

On définit l'ensemble des valeurs de vérité comme l'ensemble $\mathbf{B} = \{\mathrm{V}, \mathrm{F}\}$ où l'on interprète V comme *vrai* et F comme *faux*.

Définition Un **prédicat** sur un ensemble E est une fonction $\varphi : E \to \mathbf{B}$.

Théorème 1 - Principe d'induction

Étant donnés un ensemble muni d'un ordre bien fondé $(E, \preceq)$ et un prédicat φ sur E, si l'on a :

$$\forall x \in E, \left(\forall y \in E, y \prec x \implies \varphi(y)\right) \implies \varphi(x), \tag{H}$$

alors on en déduit que :

$$\forall x \in E, \varphi(x).$$

Preuve

Supposons par l'absurde que l'ensemble

$$A = \left\{x \in E \;\middle|\; \text{non}(\varphi(x))\right\}$$

est non vide. Puisque $(E, \preceq)$ est bien fondé, A admet un élément minimal que nous noterons x_0. Par minimalité de x_0 dans A, pour tout $y \in E$, si $y \prec x_0$, alors $y \notin A$ et donc $\varphi(y)$ est vrai. Cela implique $\varphi(x_0)$par application de (H), ce qui est absurde. Ainsi, $A = \emptyset$.

3. Lien avec la récurrence

On peut voir le principe d'induction comme une généralisation de le principe de récurrence forte. En effet, pour les entiers naturels, l'hypothèse d'induction (H) peut se réécrire :

$$\forall x \in \mathbf{N}, \left(\forall y \in \mathbf{N}, y < x \implies \varphi(y)\right) \implies \varphi(x)$$

que l'on peut éventuelle décomposer de façon équivalente en :

$$\varphi(0) \quad \textbf{et} \quad \forall x \in \mathbf{N}, \left(\forall y \in \mathbf{N}, y \leq x \implies \varphi(y)\right) \implies \varphi(x+1)$$

Plus généralement, il n'est pas nécessaire de traiter les éléments minimaux de E de façon spécifique. En effet, si x est minimal, alors $\left\{y \in E \;\middle|\; y \prec x\right\} = \emptyset$ et donc la proposition « $\forall y \in E, y \prec x \implies \varphi(y)$ » est vrai (on peut s'en convaincre en remarquant que sa négation :

$$\exists y \in E : \left(y \prec x \textbf{ et non } \varphi(y)\right)$$

est fausse). Ainsi, d'après (H), tout élément minimal de E vérifie φ.

Ainsi, le principe de récurrence forte n'est rien d'autre que l'application du principe d'induction appliqué spécifiquement à l'ensemble bien ordonné $\mathbf{N}$.

Pour la récurrence simple, on ne regarde, pour un $n \in \mathbf{N}$ donné, que son *prédécesseur*, autrement dit le plus grand élément de $\left\{k \in \mathbf{N} \;\middle|\; k < n\right\}$. Il s'agit d'un cas particulier d'induction, puisque pour un $n \in \mathbf{N}^\star$ donné, si $\varphi(k)$ est vrai pour tout $k < n$ alors en particulier $\varphi(n-1)$ est vrai.

II. Terminaison

Commençons par utiliser le principe d'induction pour la preuve de terminaison d'une fonction. Ici, la notion d'ordre bien fondé généralise celle de variant.

Proposition 2

Pour toute fonction f définie sur un ensemble E, s'il existe une fonction t dite *d'ordre* de E dans un ordre bien fondé $(F, \preceq)$ tel que pour tout $x \in E$, l'exécution de $f(x)$

- ⋆ termine (sans tenir compte des appels récursifs),
- ⋆ et ne fait qu'un nombre fini d'appels récursifs à f avec des arguments y tels que $t(y) \prec t(x)$,

alors l'exécution de f termine toujours.

Preuve

En notant φ le prédicat qui associe à $z \in F$ la propriété « l'exécution des $f(x)$ tels que $t(x) = z$ termine », alors d'après nos hypothèses, si pour tout $x \in E$, on a :

1. l'exécution de $f(x)$ hors appels récursifs termine et
2. pour tout $z \prec t(x)$, $\varphi(z)$ est vrai donc par hypothèse tous les appels récurrents sont en nombre fini et terminent.

On en déduit par principe d'induction que pour tout $z \in F$, $\varphi(z)$ est vrai, autrement dit la fonction termine.

Remarques

- ⋆ Comme fonction de terminaison, on peut souvent utiliser l'identité, la projection sur une ou plusieurs composantes. Pour des cas plus complexes, on peut utiliser la taille d'un tableau, la longueur d'une chaîne de caractères, etc. C'est la généralisation de la notion de variant pour les boucles.
- ⋆ La fonction d'ordre permet de traîter le cas où l'ensemble E n'est pas lui-même muni d'un ordre bien fondé.

Exemple

★ La fonction récursive qui calcule la factorielle termine, la fonction de terminaison étant simplement l'identité.

```
let rec fact n =
    if n = 0 then 1 else n * fact (n - 1)
;;
```

★ La fonction suivante, qui calcule les coefficients du binôme, termine. On utilise encore l'identité comme fonction de terminaison, pour l'ordre produit sur $\mathbf{N}^2$.

```
let rec binom n p =
    match n, p with
    | _, 0 -> 1
    | 0, _ -> 0
    | _    -> binom (n - 1) (p - 1) + binom (n - 1) p
;;
```

Exercice 4 — Algorithme d'Euclide
On considère la fonction suivante :

```
    let rec pgcd a b =
        if b = 0
        then a
        else pgcd b (a mod b)
    ;;
```

1. En prenant la projection selon la seconde composante comme fonction d'ordre et l'ordre usuel sur $\mathbf{N}$, montrer que la fonction termine.
2. Si l'on prends l'identité comme fonction d'ordre, les ordres bien fondés vus sur $\mathbf{N}^2$ vus précédemment permettent-ils de prouver la terminaison ?

Exercice 5 — Fonction d'Ackermann
On considère la fonction d'Ackermann, définie sur $\mathbf{N}^2$ par le programme suivant termine.

```
let rec ack m n =
    match (m, n) with
    | 0, _ -> n + 1
    | _, 0 -> ack (m - 1) 1
    | _    -> ack (m - 1) (ack m (n - 1))
;;
```

1. Prouver qu'elle termine.

Ce n'est pas parce que la fonction termine qu'elle est calculable en pratique. Par exemple, $\mathrm{Ack}(4,2) = 2^{65536} - 3$, ce qui est assez long à obtenir en faisant simplement des appels récursifs et des ajouts de 1.

2. Prouver que pour tout $n \in \mathbf{N}$, $\mathrm{Ack}(1,n) = 2 + (n+3) - 3$.
3. Prouver que pour tout $n \in \mathbf{N}$, $\mathrm{Ack}(2,n) = 2 \times (n+3) - 3$.
4. Prouver que pour tout $n \in \mathbf{N}$, $\mathrm{Ack}(3,n) = 2^{n+3} - 3$.

III. Correction

Nous allons utiliser, comme pour la terminaison, le principe d'induction pour prouver la correction d'une fonction, c'est-à-dire pour prouver que le résultat de la fonction vérifie les propriétés que l'on veut.

On utilise, comme précédemment, une variante du principe d'induction basée sur une fonction d'ordre. Ici, la fonction f pour laquelle on veut prouver une propriété est directement utilisée dans l'énoncé du prédicat et dans la preuve de l'induction.

Proposition 3

Pour toute fonction f définie sur un ensemble E et tout prédicat P sur E, s'il existe une fonction t dite *d'ordre* de E dans un ordre bien fondé $(F, \preceq)$ tel que pour tout,

$$\forall x \in E, \big(\forall y \in E, t(y) \prec t(x) \implies P(y)\big) \implies P(x)$$

alors

$$\forall x \in E, P(x)$$

En pratique, on a donc besoin d'une façon d'associer (par le biais de la fonction d'ordre) à chaque élément de l'ensemble de départ un élément d'un ensemble avec un ordre bien fondé et tel que les appels récursifs se font avec des arguments strictement plus petits pour cet ordre. Il peut s'agir, bien sûr, du même ordre que

pour la terminaison.

Exemple Prouvons que la fonction suivante calcule bien le PGCD comme on peut l'espérer :

```
let rec pgcd a b =
    if b = 0
    then a
    else pgcd b (a mod b)
;;
```

Nous allons utiliser pour cela l'ordre usuel sur $\mathbf{N}$ et la projection selon la deuxième composante comme fonction d'ordre, ainsi que le prédicat :

$$P(a,b) = \text{« } \texttt{pgcd a b} \text{ renvoie } a \wedge b \text{ »}$$

Prouvons l'étape d'induction. Soit donc $(a,b) \in \mathbf{N}^2$ tel que $P(c,d)$ est vrai pour tous (c,d) tels que $d < b$. D'après le test, deux cas sont à considérer :

1. si $b = 0$, alors `pgcd a b` renvoie a qui est bien la valeur de a $\wedge$ b. Dans ce cas, $P(a,b)$ est donc bien vérifiée.

2. sinon, `pgcd a b` renvoie `pgcd b (a mod b)` qui, par hypothèse d'induction, est égal à $b \wedge (a \bmod b)$ puisque $a \bmod b < b$. Finalement, on a :

   $$\texttt{pgcd a b} = \texttt{pgcd b (a mod b)} = b \wedge (a \bmod b) = a \wedge b$$

   ce qui prouve que $P(a,b)$ est donc aussi vérifiée.