Différence récurrence forte/récurrence faible

Théorème 1 (Récurrence faible sur $\mathbb{N}$): Soit P une propriété sur $\mathbb{N}$. On suppose que $P(0)$ et $\forall n, P(n) \Rightarrow P(n+1)$. Alors pour tout $n \in \mathbb{N}$, $P(n)$ est vérifiée.

Autrement dit, on peut montrer $\forall n \in \mathbb{N}, P(n)$ en montrant uniquement (initialisation) $P(0)$ et (hérédité) $\forall n, P(n) \Rightarrow P(n+1)$.

Théorème 2 (Récurrence forte sur $\mathbb{N}$): Soit P une propriété sur $\mathbb{N}$. On suppose que $(\forall i, i < n \Rightarrow P(i)) \Rightarrow P(n)$. Alors pour tout $n \in \mathbb{N}$, $P(n)$ est vérifiée.

Autrement dit, on peut montrer $\forall n \in \mathbb{N}, P(n)$ en montrant uniquement $(\forall i < n, P(i)) \Rightarrow P(n)$.[1]

Pour illustrer ça, quelques exemples:

- La somme $\sum_{i=0}^{n} i$ des n premiers entiers est égale à $\frac{n(n+1)}{2}$. Initialisation: $\sum_{i=0}^{0} = 0 = \frac{0\times 1}{2}$. Hérédité:

$$\sum_{i=0}^{n+1} i = \left(\sum_{i=0}^{n} i\right) + n + 1 = \frac{n(n+1)}{2} + n + 1 = \frac{n^2 + n + 2n + 2}{2} = \frac{(n+2)(n+1)}{2}$$

Donc pour tout n, $\sum_{i=0}^{n} i = \frac{n(n+1)}{2}$.

- Tout entier $n \geq 2$ peut être décomposé en produit de nombres premiers. Soit n, on suppose que $P(k)$ est vrai pour tout $2 \leq k < n$, et on montre $P(n)$. Si n est premier, alors c'est bon. Sinon, n est composé avec $n = pq$ avec $2 \leq p, q < n$. Puisque $2 \leq p, q < n$, $P(p)$ et $P(q)$ sont vérifiés. On peut donc combiner les deux décompositions en facteurs premiers de p et q pour en déduire une décomposition en facteurs premiers de n. Par récurrence forte, il existe une décomposition en facteurs premiers pour tout n.

- Un exemple plus abstrait: on peut passer (sur $\mathbb{N}$) du théorème de récurrence forte au théorème de récurrence faible, et vice versa. Je te laisse montrer que si une propriété satisfait les hypothèses du théorème de récurrence faible ($P(0)$ et $\forall n, P(n) \Rightarrow P(n+1)$), alors on peut montrer $\forall n, P(n)$ en utilisant le théorème de récurrence forte. Indiciation: il faut considérer $Q(n) := \forall k < n, P(k)$ pour utiliser le théorème de récurrence forte.

Un dernier commentaire: le principe d'induction forte fait beaucoup penser aux ordres bien fondés, comme je l'ai fait remarquer lors des TD.

Sur d'autres objets que $\mathbb{N}$

Il est aussi possible de faire des induction structurelles sur d'autres objets que les entiers. L'exemple que vous êtes peut-être en train de voir sont les arbres. Pour ça, on a l'induction structurelle qui correspond à l'induction faible. Je passe en ocaml pour te décrire quelques exemples

```
type nat = 0 | S of nat
```

[1]Petite note ici: il peut arriver d'écrire $(\forall i < n, P(i)) \Rightarrow P(n)$ pour $(\forall i, i < n, P(i)) \Rightarrow P(n)$. Ça ne change pas grand chose.

Le type `nat` correspond aux entiers naturels. Le constructeur `O:nat` correspond à 0 et le constructeur `S:nat -> nat` correspond à $n + 1$. Le principe d'induction structurelle correspondant à ce type est grosso modo le même que le théorème de récurrence faible:

Théorème 3 (Induction structurelle sur `nat`): Soit P une propriété sur `nat`. On suppose que $P(O)$ et $\forall n, P(n) \Rightarrow P(\texttt{S}\ n)$. Alors pour tout $n \in$ `nat`, $P(n)$ est vérifiée.

Un autre exemple plus compliqué est celui des arbres

```
type tree = L | N of tree * tree
```

Ici, le constructeur `L:tree` correspond à une feuille (Leaf), et le constructeur `N: tree * tree -> tree` correspond à un noeud (Node). Voici le principe d'induction structurelle sur les arbres tel que nous l'avons vu en math discrètes.

Théorème 4 (Récurrence faible sur `tree`/Induction structurelle sur `tree`): Soit P une propriété sur `tree`. On suppose que $P(L)$ et $\forall t_1 t_2, P(t_1) \wedge P(t_2) \Rightarrow P(\texttt{N}\ (t_1, t_2))$. Alors pour tout $t \in$ `tree`, $P(t)$ est vérifiée.

Ça correspond à une récurrence faible. Pour aller plus loin, on peut définir une relation d'ordre induite par l'ensemble des arbres. Pour cela, on commence par définir une relation "$\triangleleft$" par $t_1 \triangleleft N(t_1, t_2)$ et $t_2 \triangleleft N(t_1, t_2)$ pour tout t_1 et t_2. Ensuite, on prend la clôture réflexive transitive "$\trianglelefteq$" de $\triangleleft$. Cette relation correspond à la relation "est un sous arbre de". Par exemple, $L \trianglelefteq N(N(L, L), N(L, L))$. Je te laisse vérifier que c'est bien une relation d'ordre. La transitivité et reflexivité sont déjà données puisque il s'agit d'une clôture transitive réflexive. Il reste à montrer l'antisymétrie.

Le truc vraiment intéressant avec cette relation $\trianglelefteq$ est qu'elle est **bien-fondé**. Et le théorème d'induction bien fondé appliqué à cette relation est le principe de récurrence forte sur les arbres:

Théorème 5 (Récurrence forte sur `tree`/Induction bien fondée sur "$\trianglelefteq$"): Soit P une propriété sur `tree`. On suppose que pour tout t, $(\forall r, r \blacktriangleleft t \Rightarrow P(r)) \Rightarrow P(t)$. Alors pour tout $t \in$ `tree`, $P(t)$ est vérifiée.

Je te mets la preuve du théorème plus loin, car elle est un peu compliquée.

En fait, on peut faire la même chose pour tout type ocaml recursif, quelque soit son arité, et le nombre de constructeurs. Si on fait la même chose pour le type `nat`, on remarque que la relation "$\trianglelefteq$" obtenue est exactement la relation $\leq$ usuelle!

En pratique, on n'utilise que rarement le théorème de récurrence forte pour un type inductif (à part $\mathbb{N}$). À la place, on utilise une mesure (par exemple la taille d'un arbre), ce qui nous permet de faire des récurrences forte sur $\mathbb{N}$. C'est beaucoup plus pratique, parce que ça nous permet de modifier sous arbres, en récupérant directement les hypothèses d'induction dans les preuves:

Théorème 6 (Récurrence forte sur `tree`/Induction sur la hauteur): Soit P une propriété sur `tree`. On suppose que pour tout t, $(\forall r, h(r) < h(t) \Rightarrow P(r)) \Rightarrow P(t)$ où h est la hauteur d'un arbre. Alors pour tout $t \in$ `tree`, $P(t)$ est vérifiée.

Théorème 7 (Récurrence forte sur `tree`/Induction sur la taille): Soit P une propriété sur `tree`. On suppose que pour tout t, $(\forall r, n(r) < n(t) \Rightarrow P(r)) \Rightarrow P(t)$ où n est le nombre de noeuds d'un arbre. Alors pour tout $t \in$ `tree`, $P(t)$ est vérifiée.

J'aimerais te donner un exemple simple, mais ça serait mieux que tu me décrives la structure de donnée sur laquelle tu as besoin d'induction forte.

Preuve du Théorème 5

Passons à la preuve du théorème de récurrence forte pour terminer. C'est un passage (théorique) obligatoire pour définir les fonction de taille ou de hauteur des arbres.

Preuve du Théorème 5: Il suffit de montrer que $\trianglelefteq$ est bien fondée. Pour cela, on va montrer par induction structurelle (Théorème 4) la propriété suivante:

$$P(t) := \text{“Il n‘existe pas de suite infinie strictement décroissante pour } \trianglelefteq \text{ commençant par } t.\text{“}$$

- (Initialisation). L est un plus petit élement, donc il n'y a pas de suite infinie strictement décroissante commençant par t.
- Hérédité. On suppose par l'absurde qu'il existe une suite infinie strictement décroissante $N(t_1, t_2) \blacktriangleright r_1 \blacktriangleright r_2 \blacktriangleright \cdots$. On distingue trois cas.
  - Si $r_1 = t_1$, alors la suite $t_1 \blacktriangleright r_2 \blacktriangleright \cdots$ est infinie décroissante, ce qui est en contradiction avec $P(t_1)$.
  - De même si $r_1 = t_2$, alors la suite $t_2 \blacktriangleright r_2 \blacktriangleright \cdots$ est infinie décroissante, ce qui est en contradiction avec $P(t_2)$.
  - Il reste le cas où $r_1 \notin \{t_1, t_2\}$. Dans ce cas, par définition de “$\trianglelefteq$”, puisque $r_1 \notin \{t_1, t_2\}$, soit $t_1 \blacktriangleright r_1$ soit $t_2 \blacktriangleright r_1$. Dans les deux cas, cela revient encore à une contradiction avec soit $P(t_1)$ soit $P(t_2)$.

On conclut par le principe d'induction structurelle sur `tree` que pour tout t, il n'existe pas de suite infinie décroissante commençant par t. Donc $\trianglelefteq$ est bien fondée. ■