MÉTHODES DE PREUVES

MICHEL BOYER

RÉSUMÉ. Essentiellement Johnsonbaugh sections 1.5 “Preuves” et 1.7 “Induction mathématique”. La différence essentielle est que l’on fait ressortir le lien entre les preuves par induction mathématiques et des fonctions définies récursivement.

4. PREUVES

4.1. Système mathématique.

Axiome: Enoncé admis comme vrai. Exemples:

(1) si x et y sont réels, $xy = yx$

(2) $\forall x \exists x' (x + x' = 0)$.

Définition: Introduit un nouveau concept construit à partir de concepts existants. Exemple:

(1) Un entier n est dit *pair* s’il existe un entier k tel que $n = 2k$.

(2) Un entier n est *impair* s’il existe un entier k tel que $n = 2k + 1$.

Théorème: Enoncé qui est conséquence logique des axiomes. On réserve souvent le nom de théorèmes aux conséquences particulièrement utiles ou importantes. Exemple:

(1) Tout entier positifs s’écrit de façon unique (à ordre des facteurs prés) comme produit de nombre premiers.

(2) Pour tout entier n et tout entier $d > 0$ il existe un unique entier q (appelé *quotient*) et un unique entier r (appelé *reste*) tels que $n = dq + r$ et $0 \leq r < b$.

Lemme: Théorème mineur ou facile qui sert souvent d’étape intermédiaire pour la démonstration d’un théorème en “plein titre”. Exemple: tout entier est soit pair, soit impair.

Corollaire: Conséquence facile d’un théorème

Preuve: Argumentation qui utilise des *règles d’inférence* pour déduire des conséquences ‘a partir de résultats déjà démontrés et d’axiomes (pour

démontrer un théorème) et souvent aussi d’hypothèses (pour démontrer une implication).

4.2. **Preuves directes.** Nous donnons ici un exemple de preuve directe que si m est un entier impair et n un entier pair, alors $n + m$ est un entier impair:

(4.2.1)	$m = 2k_1 + 1$	m impair
(4.2.2)	$n = 2k_2$	n pair
(4.2.3)	$n + m = 2k_2 + (2k_1 + 1)$	de (4.2.1) et (4.2.2)
	$\forall x \forall y \forall z((x + y) + z = x + (y + z))$	axiome d’associativité
(4.2.4)	$(2k_2 + 2k_1) + 1 = 2k_2 + (2k_1 + 1)$	
(4.2.5)	$n + m = (2k_2 + 2k_1) + 1$	de (4.2.3 et (4.2.4)
	$\forall x \forall y \forall z(x(y + z) = xy + xz)$	axiome de distributivité
(4.2.6)	$2(k_2 + k_1) = 2k_2 + 2k_1$	
(4.2.7)	$n + m = 2(k_2 + k_1) + 1$	de (4.2.5) et (4.2.6)

En général, on applique l’associativite et les autres règles connues sans les mettre explicitement dans la preuve mais un système informatisée de déduction automatique devrait tout de même les utiliser pour construire sa preuve.

4.3. **Preuves par contradiction.** Elles sont basées sur le fait que

$$p \to q \equiv (p \wedge \neg q) \to \mathrm{F}$$

où F est un énoncé toujours faux, (par exemple $r \wedge \neg r$).

p	q	$\neg q$	$p \to q$	$p \wedge \neg q$	F	$(p \wedge \neg q) \to \mathrm{F}$
V	V	F	V	F	F	V
V	F	V	F	V	F	F
F	V	F	V	F	F	V
F	F	V	V	F	F	V

Il s’agit donc de supposer p et la négation de q et d’en arriver à un énoncé faux.

Proposition 1. *Si $x + y \geq 2$ avec, x et y réels, alors $x \geq 1$ ou $y \geq 1$ (i.e. $(x + y) \geq 2 \to ((x \geq 1) \vee (y \geq 1))$).*

Preuve. On suppose que $x+y \geq 2$ et $\neg((x \geq 1) \vee (y \geq 1))$

(4.3.1)	$x+y \geq 2$	hypothèse
(4.3.2)	$\neg((x \geq 1) \vee (y \geq 1))$	hypothèse
(4.3.3)	$\neg(x \geq 1) \wedge \neg(y \geq 1)$	de Morgan et (4.3.2)
(4.3.4)	$(x < 1) \wedge (y < 1)$	
(4.3.5)	$x+y < 2$	de (4.3.4)
(4.3.6)	F	de (4.3.5) et (4.3.1)

□

4.4. **Preuve par contrapositive.** C'est une preuve qui se base sur le fait que $p \rightarrow q$ est équivalent à sa contrapositive $\neg q \rightarrow \neg p$. En enlevant la première et la dernière ligne de la preuve précédente, on obtient (c'est un hasard) une preuve par contrapositive du résultat que nous voulions démontrer.

4.5. **Preuve par cas.** Pour prouver que $(p_1 \vee p_2 \ldots \vee p_n) \rightarrow q$ on prouve pour chaque cas que $(p_1 \rightarrow q)$, $p_2 \rightarrow q \ldots p_n \rightarrow q$. On utilise ici le fait que $(p_1 \vee \ldots p_n) \rightarrow q$ est équivalent à $(p_1 \rightarrow q) \wedge \ldots \wedge (p_n \rightarrow q)$.

4.6. **Preuve d'existence.** Pour prouver $\exists x \mathrm{P}(x)$ suffit d'exiber une valeur de x qui satisfait $\mathrm{P}(x)$. Pour prover que $\forall x \mathrm{P}(x)$ il faut prouver qu'il n'existe pas de x pour lequel P soit faux i.e. $\neg\exists x \neg \mathrm{P}(x)$.

4.7. **Démonstration.** On appelle démonstration ("argument" dans Johnsonbaugh) une suite de proposition $p_1, p_2, \ldots, p_n / \therefore q$ où le symbole $\therefore$ se lit "donc" ou "par conséquent" et qui signifie que si $p_1, \ldots p_n$ sont vraies, alors q doit l'être; l'argumentation démontre donc q sous les hypothèses $p_1, \ldots, p_n$ qui peuvent être des hypothèse posées par celui qui veut prouver ou même des axiomes. Une preuve s'appuie en général sur des arguments types appelés règles d'inférence

(et en limitant les règles admises, on peut changer les règles de la logique, ce que font les logiciens et le philosophes):

Rêgle d'inférence	Nom
$p \rightarrow q$ p $\therefore q$	Modus ponens
$p \rightarrow q$ $\neg q$ $\therefore \neg p$	Modus tollens
$p \rightarrow q$ $q \rightarrow r$ $\therefore p \rightarrow r$	Syllogisme hypothétique
$p \vee q$ $\neg p$ $\therefore q$	Syllogisme disjonctif
p q $\therefore p \wedge q$	Conjonction
p $\therefore p \vee q$	Addition
$p \wedge q$ $\therefore p$	Simplification

Chacune des règles d'inférence ci-dessus correspond à une tautologie, i.e. à une proposition toujours vraie.

Tautologie	Règle d'inférence
$((p \rightarrow q) \wedge p) \rightarrow q$	Modus ponens
$((p \rightarrow q) \wedge \neg q) \rightarrow \neg p$	Modus tollens
$((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow r)$	Syllogisme hypothétique
$((p \vee q) \wedge \neg p) \rightarrow q$	Syllogisme disjonctif
$p \wedge q \rightarrow p \wedge q$	Conjonction
$p \rightarrow p \vee q$	Addition
$p \wedge q \rightarrow p$	Simplification

Pour le quantificateur universel $\forall x \mathrm{P}(x)$ permet d'en déduire $\mathrm{P}(d)$ pour n'importe quel élément d du domaine. Par contre $\exists x \mathrm{P}(x)$ permet simplement de se donner un d_0 tel que $\mathrm{P}(d_0)$ et ce d_0 doit être différent de toutes les constantes qu'on a déjà utilisé dans la preuve. D'autre part, si on a déjà prouvé P(A) alors on peut en déduire $\exists x \mathrm{P}(x)$.

Enfin (et même si c'est sous-entendu, ça n'est pas du tout banal) si on a $P(e)$ pour une certaine expression e est si $e = f$ (où f est une autre expressions, par exemple e est $2k_2 + 2k_1$ et f est $2(k_2 + k_1)$, alors on peut déduire $P(f)$ de $P(e)$: on peut toujours rempacler une valeur par une autre qui lui est égale.

5. L'INDUCTION MATHÉMATIQUE

C'est une méthode de preuve pour des prédicats portant sur les entiers.

Si $P(n)$ est tel que

(1) $P(n_0)$ est vrai
(2) pour tour $n \geq n_0$, $P(n) \rightarrow P(n+1)$

alors $P(n)$ est vrai pour tout $n \geq n_0$.

L'entier n_0 est appelé la base (et aussi la preuve que $P(n_0)$ est vrai); le plus souvent $n_0 = 0$ ou $n_0 = 1$.

Quand fait-on des preuves par induction mathématique? Quelquefois on a simplement découvert l'énoncé $P(n)$ plus ou moins expérimentalement et on n'arrive pas à faire une preuve directe et on s'essaie par induction. On est aussi très souvent obligé de faire des preuves par induction pour démontrer des propriétés de fonctions définies récursivement.

5.1. **Exemple de la factorielle.** Par exemple pour définir $k! = k \times (k-1) \times \ldots \times 2 \times 1$ sans petits points, on pose:

$$k! = \begin{cases} 1 & \text{si } k = 0 \\ k \times (k-1)! & \text{si } k \geq 1 \end{cases}$$

Donc $4! = 4 \times 3! = 4 \times 3 \times 2! = 4 \times 3 \times 2 \times 1! = 4 \times 3 \times 2 \times 1 \times 0! = 24$.

On veut maintenant prouver que $n! \geq 2^{n-1}$ pour $n \geq 1$. Donc la base ici est $n_0 = 1$. Bien sûr le résultat est tout à fait évident mais faisons quand même l'exercice. On a que $P(k)$ est $n! \geq 2^{k-1}$ donc $P(n)$ est $n! \geq 2^{k-1}$ et $P(n+1)$ est $(n+1)! \geq 2^n$

base: On démontre $P(1)$ i.e. que $1! \geq 2^0$: suffit de calculer; $1! = 1 \times 0! = 1 = 1$; $2^0 = 1$.

pas: On démontre que $(n \geq 1) \wedge P(n) \rightarrow P(n+1)$

(5.1.1)	$n \geq 1$	$n \geq 1$
(5.1.2)	$n! \geq 2^{n-1}$	$P(n)$
(5.1.3)	$n + 1 \geq 2$	
(5.1.4)	$(n+1)! = (n+1)n!$	car $n + 1 \geq 1$
(5.1.5)	$(n+1)! \geq 2 \times 2^{n-1}$	par (5.1.2) et (5.1.3)
(5.1.6)	$(n+1)! \geq 2^{(n+1)-1}$	$P(n+1)$

De façon générale on y va à l'envers en posant $P(k)$ comme étant $k! \geq 2^{k-1}$ puis en écrivant $P(n)$ qui est $n! \geq 2^{n-1}$ et $P(n+1)$ qui est $(n+1)! \geq 2^n$.

5.2. **Exemple de la progression géométrique.** On veut montrer que

$$a + ar^1 + \ldots + ar^n = \frac{a(r^{n+1} - 1)}{r - 1} \tag{5.2.1}$$

pour tout $n \geq 0$. Notons $G(k)$ la somme de k termes et définissons correctement $G(k)$ récursivement:

$$G(k) = \begin{cases} a & \text{si } k = 0 \\ ar^k + G(k-1) & \text{si } k \geq 1 \end{cases}$$

Une fois cette définition clairement posée, il devient naturel de vouloir utiliser l'induction mathématiqaue pour démontrer l'égalité (5.2.1). Il faut montrer $P(n)$ pour $n \geq 0$ où $P(k)$ est par définition $G(k) = \frac{a(r^{k+1} - 1)}{r - 1}$.

base: Il faut prouver $P(0)$ i.e. $G(0) \geq \frac{a(r^{0+1} - 1)}{r - 1} = a$. Mais $G(0)$ est justement a ce qui termine la preuve.

pas d'induction: On pose $n > 0$ et $P(n)$ et il faut en déduire $P(n+1)$:

(5.2.2)	$n \geq 0$	$n \geq 0$
(5.2.3)	$G(n) \geq \frac{a(r^{n+1} - 1}{r - 1}$	$P(n)$
(5.2.4)	$G(n+1) = ar^{n+1} + G(n)$	car $k = n + 1 \geq 1$ par (5.2.2)
(5.2.5)	$G(n+1) \geq ar^{n+1} + \frac{a(r^{n+1} - 1)}{r - 1}$	par (5.2.4) et (5.2.3)
(5.2.6)	$G(n+1) \geq \frac{a(r^{n+2} - 1)}{r - 1}$	$P(n+1)$ (après simplification)

La conclusion obtenue en dernière ligne n'est autre que $P(n+1)$ à savoir[1] $G(k) \geq \frac{a(r^{k+1}-1)}{r-1}$ pour $k = n+1$.

5.3. **Invariant de boucle (pas fait en classe).** Soit le pseudo code

```
i = 1
fact = 1
while (i < n){
  i = i+1
  fact = fact * i
}
```

On veut montrer qu'en fin d'exécution (pour $n \geq 1$) on on a bien $fact = n!$; pour cela, on démontre par induction qu'à chaque tour de boucle, après l'affectation `fact=fact * i` on a $fact = i!$. Quand on sort de la boucle, c'est que $i = n$ et on a alors $fact = n!$. On dit que $fact = i!$ est un invariant de boucle.

Notons k la valeur de `i` avant d'entrer dans la boucle.

base: Si $k = 1$ c'est qu'on n'est pas encore rentré dans la boucle et alors on a bien $fact = i!$ car $fact = 1$ et $i = 1$.

pas: Juste avant le tour de boucle on a par hypothèse que `i` contient k et `fact` contient $k!$. L'affectation `i = i+1` incrémenté i de 1 et i vaut dont maintenant $k+1$. Ensuite l'affectation `fact = fact * i` fait que `fact` devient égal à $k! \times (k+1) = (k+1)!$. Donc on a encore $fact = i!$ à la fin du tour de boucle. Cette égalité est laissée invariante par le corps de la boucle et va donc rester vraie jusqu'à ce que l'on arrête de tourner.

Quand on sort de la boucle, comme déjà dir, c'est que la condition $i < n$ n'est plus satisfaite et celà arrive quand $i = n$; on a alors $fact = i! = n!$ et c'est gagné. On vient de faire une preuve qu'un programme fait ce qu'on attendait de lui.

1. En effet $ar^{n+1} + \frac{a(r^{n+1}-1)}{r-1} = \frac{ar^{n+2}-ar^{n+1}}{r-1} + \frac{ar^{n+1}-a}{r-1} = \frac{ar^{n+2}-a}{r-1} = \frac{a(r^{n+2}-1)}{r-1}$.