Brève introduction à la théorie des ensembles

$$\begin{aligned}\mathcal{P}(\{a,b,c,d,e\}) \;=\; \Big\{ & \emptyset, \{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \\ & \{a,b\}, \{a,c\}, \{a,d\}, \{a,e\}, \{b,c\}, \\ & \{b,d\}, \{b,e\}, \{c,d\}, \{c,e\}, \{d,e\}, \\ & \{a,b,c\}, \{a,b,d\}, \{a,b,e\}, \{a,c,d\}, \\ & \{a,c,e\}, \{a,d,e\}, \{b,c,d\}, \{b,c,e\}, \\ & \{b,d,e\}, \{c,d,e\}, \{a,b,c,d\}, \{a,b,c,e\}, \\ & \{a,b,d,e\}, \{a,c,d,e\}, \{b,c,d,e\}, \{a,b,c,d,e\} \Big\}\end{aligned}$$

François Bergeron, Département de mathématiques, UQAM

11 mai 2011

Table des matières

Chapitre 1

Ensembles et fonctions

1.1 Introduction

Les notions de la théorie des ensembles et des fonctions sont à la base d'une présentation moderne des mathématiques. Immanquablement, on y fait appel pour la construction d'objets plus complexes, ou pour donner une base solide aux arguments logiques. En plus d'être des notions fondamentales pour les mathématiques, elles sont aussi cruciales en informatique, par exemple pour introduire la notion de « structures de données ».

1.2 Ensembles

Georg Cantor
(1845–1918)

La théorie des ensembles a été introduite par Georg Cantor. On peut en donner une axiomatique rigoureuse qui n'est pas vraiment approfondie ici (voir cependant l'appendice B). C'est tout de même un aspect important de la question comme on va l'entrevoir à la section 1.8. La théorie suppose que les ensembles contiennent des *éléments*, et on écrit $x \in A$ pour dire que « x est un élément de A ». Deux ensembles sont égaux si et seulement s'ils ont les mêmes éléments. Autrement dit, pour « connaître » un ensemble il faut savoir dire quels en sont les éléments. Ainsi, on a les présentations équivalentes

$$\{a, b, c\} = \{c, a, b\} = \{a, b, a, b, c, a, b, a\},$$

d'un même ensemble qui contient les trois éléments : a, b et c.

Typiquement, sans les définir très rigoureusement ici, on commence par considérer des ensembles « simples » comme

- L'ensemble des *entiers naturels*,

$$\mathbb{N} := \{0, 1, 2, 3, \ \dots\};$$

- L'ensemble des *entiers*,

$$\mathbb{Z} := \{\ \dots\ , -3, -2, -1, 0, 1, 2, 3, \ \dots\};$$

- L'ensemble des *nombres rationnels*,

$$\mathbb{Q} := \left\{ \frac{a}{b} \mid a \in \mathbb{Z}, \quad b \in \mathbb{N}, \quad \text{et} \quad b \neq 0 \right\};$$

l'ensemble $\mathbb{R}$ des *nombres réels*, qui inclue les nombres rationnels et tous les nombres qu'on peut construire [1] à partir de ceux-ci par « passage à la limite » ; l'ensemble des entiers naturels entre 1 et n

$$[n] := \{i \in \mathbb{N} \mid 1 \leq i \leq n\};$$

l'ensembles de lettres (minuscules) de l'alphabet

$$\mathcal{A} := \{a, b, c, d, \ \dots\ , z\};$$

ou encore des ensembles d'objets divers comme

$$\{\bullet, \bullet, \bullet\}, \qquad \text{où} \qquad \{\clubsuit, \ \diamondsuit, \ \heartsuit, \ \spadesuit\}.$$

À partir de tels ensembles de « base » on construit des ensembles plus complexes au moyen d'opérations entre ensembles qui seront introduites dans les sections suivantes. Une axiomatique correcte de la théorie des ensembles explique comment procéder à des descriptions admissibles d'ensembles de bases, et comment construire ensuite des ensembles plus complexes. Ainsi, on peut décrire l'ensemble

$$A = \{x \in S \mid P(x)\}, \tag{1.1}$$

des éléments de S qui satisfont une certaine propriété $P(x)$, formulée sous forme d'énoncé logique (voir AppendiceA). Dans ce cas, la notion d'égalité $A = B$, avec $B = \{x \in S \mid Q(x)\}$, correspond au fait que les propriétés $P(x)$ et $Q(x)$ sont logiquement équivalentes. On dénote $\emptyset$, l'*ensemble vide*, qui ne contient aucun élément. Le nombre d'éléments (ou *cardinal*) d'un ensemble fini A, est dénoté $|A|$, ou parfois aussi $\#A$.

1. À voir dans un cours d'analyse.

1.3 Sous-ensembles

Si tous les éléments de B sont aussi des éléments de A, on dit que B est un *sous-ensemble* de A, et on écrit

$$B \subseteq A.$$

On dit aussi que B est une *partie* de A. Se donner un sous-ensemble B, de k éléments d'un ensemble A de cardinal n, correspond donc à

$$\ll \textit{ choisir } k \textit{ éléments parmi } n \gg.$$

L'inclusion d'ensembles possède les propriétés suivantes. Pour tout A, B et C, on a

$$\begin{array}{l} \text{a) } \emptyset \subseteq A, \\ \text{b) } A \subseteq A, \\ \text{c) si } A \subseteq B \quad \text{et} \quad B \subseteq A, \quad \text{alors} \quad A = B, \\ \text{d) si } A \subseteq B \quad \text{et} \quad B \subseteq C, \quad \text{alors} \quad A \subseteq C. \end{array} \tag{1.2}$$

Lorsque A et B sont décrit comme en (1.1), on a $B \subseteq A$ si et seulement si l'énoncé logique $\ll Q(x) \Longrightarrow P(x) \gg$ est vrai.

On dénote $\mathcal{P}[S]$ l'ensemble de tous les sous-ensembles de S :

$$\mathcal{P}[S] := \{A \mid A \subseteq S\}. \tag{1.3}$$

On dit aussi que $\mathcal{P}[S]$ est l'*ensemble des parties* de S. Par exemple,

$$\mathcal{P}[\{a,b,c\}] = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a,b\}, \{a,c\}, \{b,c\}, \{a,b,c\}\}.$$

Pour chaque $0 \leq k$, on considère aussi l'ensemble des *parties à k-éléments* de S :

$$\mathcal{P}_k[S] := \{A \mid A \subseteq S, \quad |A| = k\}.$$

C'est donc

$$\ll \text{l'ensemble des possibilités de choix de } k \text{ éléments parmi } n \gg.$$

Ainsi, on a $\mathcal{P}_0[S] = \{\emptyset\}$, puis

$$\mathcal{P}_1[S] = \{\{x\} \mid x \in S\},$$

et encore

$$\mathcal{P}_2[S] = \{\{x,y\} \mid x, y \in S, \quad \text{et} \quad x \neq y\},$$

et ainsi de suite jusqu'à $\mathcal{P}_n[S] = \{S\}$, pour n égal au cardinal de S. Ainsi, si $S = \{a, b, c\}$, alors on a

$$\mathcal{P}_0[S] = \{\emptyset\}, \quad \mathcal{P}_1[S] = \{\{a\}, \{b\}, \{c\}\}, \quad \mathcal{P}_2[S] = \{\{a, b\}, \{a, c\}, \{b, c\}\}, \quad \mathcal{P}_3[S] = \{\{a, b, c\}\}.$$

Ces ensembles contiennent donc respectivement 1, 3, 3, et 1 éléments, et leur union disjointe donne l'ensemble à 8 éléments $\mathcal{P}[S]$ vu plus haut.

Attention, les ensembles

$$\{a, b, c\} \quad \text{et} \quad \{\{a\}, \{b\}, \{c\}\},$$

sont différents, puisqu'ils n'ont pas les mêmes éléments. Autrement dit, les accolades « { » et « } » ont un ici rôle mathématique important. Ce ne sont pas que de simples séparateurs comme en français. On dit des éléments $\{x, y\}$ de $\mathcal{P}_2[A]$ que ce sont des *paires* d'éléments de A. On a alors forcément $x \neq y$.

1.4 Opérations de base sur les ensembles

Une première opération de base entre ensembles est celle d'*intersection*, $A \cap B$, entre deux ensembles A et B. C'est l'ensemble des éléments qui sont communs à ces deux ensembles. Plus précisément on a

$$A \cap B := \{x \mid x \in A, \quad \text{et} \quad x \in B\}. \tag{1.4}$$

Par exemple, on a

$$\{a, b, c, d\} \cap \{1, b, 3, d\} = \{b, d\}.$$

D'autre part, l'*union* de A et B est l'ensemble

$$A \cup B := \{x \mid x \in A, \quad \text{où} \quad x \in B\}. \tag{1.5}$$

Par exemple

$$\{a, b, c, d\} \cup \{1, b, 3, d\} = \{a, b, c, d, 1, 3\}.$$

On vérifie facilement que les égalités suivantes sont valables en général, quels que soient les ensembles A, B, et C :

$$\begin{array}{ll} \text{(i)} \ A \cap \emptyset = \emptyset, & \text{(i)}' \ A \cup \emptyset = A, \\ \text{(ii)} \ A \cap B = B \cap A, & \text{(ii)}' \ A \cup B = B \cup A, \\ \text{(iii)} \ A \cap (B \cap C) = (A \cap B) \cap C, & \text{(iii)}' \ A \cup (B \cup C) = (A \cup B) \cup C, \\ \text{(iv)} \ A \cap (B \cup C) = (A \cap B) \cup (A \cap C), & \text{(iv)}' \ A \cup (B \cap C) = (A \cup B) \cap (A \cup C). \end{array} \tag{1.6}$$

Lorsque $A \cap B = \emptyset$, on dit que $A \cup B$ est une *union disjointe*, et on écrit alors $A + B$ pour désigner cette union. Voir l'exercice **??** pour ce qui concerne les propriétés de l'union disjointe [2].

La *différence*, $A \setminus B$, de deux ensembles A et B, est l'ensemble des éléments de A qui ne sont pas dans B, i.e. :

$$A \setminus B := \{x \in A \mid x \notin B\}. \tag{1.7}$$

En supposant qu'on a un ensemble S fixé, on dénote $\overline{A}$ le *complément* de A dans S. C'est tout simplement un autre nom pour

$$\overline{A} := S \setminus A,$$

qu'il est pratique d'utiliser lorsque le « sur-ensemble » S est clair dans le contexte. Ainsi, lorsqu'on fixe $S = \{a, b, c, d, e, f\}$ et $A = \{b, c, e\}$, on a $\overline{A} = \{a, d, f\}$. Si $A = \{x \in S \mid P(x)\}$, alors on a

$$\overline{A} = \{x \in S \mid \neg P(x)\}.$$

Quels que soit A et B des sous-ensembles de S (donc des éléments de $\mathcal{P}[S]$), les identités suivantes sont valables

$$\begin{array}{lll} \text{(i)}\ \overline{\overline{A}} = A, & & \\ \text{(ii)}\ A \cap \overline{A} = \emptyset, & \text{(ii)}'\ A \cup \overline{A} = S, & \\ \text{(iii)}\ \overline{A} \cap \overline{B} = \overline{A \cup B}, & \text{(iii)}'\ \overline{A} \cup \overline{B} = \overline{A \cap B}. & \end{array} \tag{1.8}$$

Pour une famille d'ensembles [3]

$$\{A_1,\ A_2,\ A_3,\ \ldots\ , A_n\},$$

on a les unions et intersections

$$\bigcap_{i=1}^{n} A_i, \quad \text{et} \quad \bigcup_{i=1}^{n} A_i.$$

En fait, $\mathcal{P}[S]$ est toujours l'union (disjointe, voir Exercice **??** pour la notation) des ensembles $\mathcal{P}_k[S]$, i.e. :

$$\mathcal{P}[S] = \sum_{k=0}^{\infty} \mathcal{P}_k[S]. \tag{1.9}$$

Comme $\mathcal{P}_k[S] = \emptyset$, si k est plus grand que le cardinal de S, cette sommation est en fait finie. On a donc ici deux descriptions du même ensemble.

2. Lorsque A et B ne sont pas disjoints ($A \cap B \neq \emptyset$), on peut tout de même considérer leur union disjointe en « forçant » A et B à être disjoints. Plus précisément, on pose

$$A + B = (\{0\} \times A) \cup (\{1\} \times B).$$

On donne ainsi des « couleurs » distinctes aux éléments de A et de B.

3. On exploite ici l'associativité de l'union et de l'intersection.

On peut « calculer » récursivement l'ensemble des parties à k éléments d'un ensemble $S = T + \{x\}$, avec $x \notin T$, en posant

$$A \in \mathcal{P}_k[S] \quad \text{ssi} \quad \begin{cases} 1)\ A = S \text{ et } k = |S|, \\ 2)\ A \in \mathcal{P}_k[T], \quad \text{où} \\ 3)\ A = B + \{x\}, \quad \text{et} \quad B \in \mathcal{P}_{k-1}[T]. \end{cases} \tag{1.10}$$

1.5 Produit cartésien

Avant d'introduire la prochaine construction, rappelons que deux couples (a, b) et (c, d) sont égaux, si et seulement si on a les deux égalités $a = c$ et $b = d$. Le *produit cartésien* de A et B est l'ensemble de tous les couples (x, y), avec x élément de A et y élément de B. Autrement formulé, on a

$$A \times B = \{(x, y) \mid x \in A \quad \text{et} \quad y \in B\}.$$

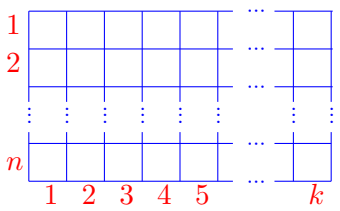
Si l'un des ensemble A ou B est vide, alors le produit cartésien $A \times B$ est vide, i.e. :

$$A \times \emptyset = \emptyset \times B = \emptyset. \tag{1.11}$$

Une illustration du produit cartésien est donnée par

$$\begin{aligned} \{a, b, c, d\} \times \{1, 2, 3, 4, 5\} = \ & \{(a,1), \ (a,2), \ (a,3), \ (a,4), \ (a,5), \\ & (b,1), \ (b,2), \ (b,3), \ (b,4), \ (b,5), \\ & (c,1), \ (c,2), \ (c,3), \ (c,4), \ (c,5), \\ & (d,1), \ (d,2), \ (d,3), \ (d,4), \ (d,5)\}. \end{aligned}$$

Dans le cas des ensembles finis $[n] = \{1, 2, \ldots, n\}$ et $[k] = \{1, 2, \ldots, k\}$, on constate que les éléments du produit cartésien $[n] \times [k]$ s'identifient aux *cases* d'un *tableau* (ou d'une *matrice*) ayant n lignes et k colonnes :



Remarquons qu'on utilise ici des *coordonnées matricielles*, indexant les lignes du haut vers le bas, plutôt que des *coordonnées cartésiennes* pour lesquelles on indexerait les lignes du bas vers le haut. À strictement parler, le produit cartésien n'est pas associatif. Ainsi, les éléments de $(A \times B) \times C$ sont de la forme $((x, y), z)$, avec $x \in A$, $y \in B$ et $z \in C$; tandis que ceux de $A \times (B \times C)$ sont

de la forme $(x,(y,z))$. On considère cependant souvent une construction intermédiaire, dénotée $A \times B \times C$, dont les éléments sont les triplets (x,y,z). Lorsque A, B et C sont des ensembles finis, les éléments de $A \times B \times C$ peuvent se représenter sous forme de tableau tridimensionnel (un peu comme dans la Figure ci-contre).

Plus généralement, on a le produit cartésien multiple

$$A_1 \times A_2 \times \cdots \times A_n = \{(x_1, x_2, \ldots, x_n) \mid x_i \in A_i,\ 1 \leq i \leq n\}.$$

Lorsque $A_i = B$, pour tous les i, on obtient la *puissance cartésienne* n-ième,

$$B^n := \underbrace{B \times B \times \cdot \times B}_{n \text{ copies}},$$

de l'ensemble B, avec $B^0 := \{*\}$. Les éléments de B^n sont les n-tuplets $(x_1, x_2, \ldots, x_n)$, d'éléments de $x_i \in B$. Il arrive parfois qu'on veuille écrire plus simplement $x_1\, x_2\, \ldots\, x_n$, pour un tel élément. On dit alors qu'on l'a écrit sous *forme de mot*.

1.6 Relation

Une *relation* R, entre les ensembles A et B, est simplement un sous-ensemble du produit cartésien $A \times B$, i.e. : $R \subseteq A \times B$. Par exemple, pour $A = \{a,b,c,d\}$ et $B = \{1,2,3\}$, on a la relation

$$R = \{(a,1),(a,2),(b,3),(c,2),(d,1)\}.$$

Un exemple typique est la relation (entre êtres humains) :

$$R := \ \ll \text{est un ancêtre de} \gg.$$

Si $\mathcal{H}$ désigne l'ensemble des êtres humains (morts ou vivants), on définit récursivement $R \subseteq \mathcal{H} \times \mathcal{H}$ en posant :

$$(a,b) \in R \quad \text{ssi} \quad \begin{cases} 1)\ a \text{ est le père ou la mère de } b, \text{ ou} \\ 2)\ (a,c) \in R \text{ et } (c,b) \in R. \end{cases} \tag{1.12}$$

La première clause amorce le processus, et la seconde affirme que

$$\ll \textit{les ancêtres de mes ancêtres sont mes ancêtres} \gg.$$

Une relation R sur A est dite

(a) *réflexive* si pour chaque x dans A, on a $(x,x) \in R$,

(b) *symétrique* si $(x, y) \in R$ entraine $(y, x) \in R$,

(c) *antisymétrique* si $(x, y) \in R$ et $(y, x) \in R$ entraine $x = y$,

(d) *transitive* si $(x, y) \in R$ et $(y, z) \in R$ entraine $(x, z) \in R$.

La plus simple des relations réflexives, sur un ensemble A, est la *relation d'égalité* entre éléments de A. Le sous-ensemble correspondant de $A \times A$ est clairement $\{(x, x) \mid x \in A\}$.

Une *relation d'équivalence* R sur A, est une relation qui est à la fois réflexive, symétrique et transitive. Plutôt que d'écrire $(x, y) \in R$, on écrit souvent $x \sim y$ (ou $x \equiv y$), et on dit que x est équivalent à y. Ainsi, $\ll \sim \gg$ est une relation d'équivalence si et seulement si, pour tout x, y et z dans A, on a

(a) $x \sim x$,

(b) si $x \sim y$, alors $y \sim x$,

(c) si $x \sim y$ et $y \sim z$, alors $x \sim z$.

1.7 Fonctions

Gottfried Leibniz
(1646–1716)

Apparemment, le terme « fonction » a été introduit par Leibniz. Pendant longtemps la définition de cette notion n'a pas été très claire. Dans l'encyclopédie de d'Alembert, on dit à peu près qu'une fonction est donnée par une formule impliquant une variable. Rappelons qu'aujourd'hui on s'accorde plutôt sur le fait de donner une approche abstraite à la notion de fonction (voir Section **??**), en donnant seulement un critère qui permet simplement de dire quand on a affaire à une fonction. Entre autres, cela rend possible la définition de l'ensemble des fonctions de A vers B. Ainsi, on considère l'ensemble $\mathsf{Fonct}[A, B]$ dont les éléments sont les relations fonctionnelles de A vers B, c'est-à-dire que

$$\begin{aligned} f \subseteq A \times B, \quad \text{où} \quad & \text{(i) } \forall x\ \exists y\ (x \in A,\ y \in B, \quad \text{et} \quad (x, y) \in f), \\ & \text{(ii) } \forall x\ \forall y_1\ \forall y_2\ ((x, y_1) \in f \quad \text{et} \quad (x, y_2) \in f \implies y_1 = y_2). \end{aligned} \tag{1.13}$$

On a alors, pour chaque élément f de $\mathsf{Fonct}[A, B]$, une fonction

$$f : A \to B$$

de *source* A et de *but* B. Il nous arrivera souvent de parler de la fonction f, si la source et le but associés sont clair dans le contexte. Pour $A = \{a, b, c\}$ et $B = \{0, 1\}$, on a

$$\begin{aligned} \mathsf{Fonct}[A, B] = \{ \quad & \{(a,0),(b,0),(c,0)\}, \quad \{(a,0),(b,0),(c,1)\}, \\ & \{(a,0),(b,1),(c,0)\}, \quad \{(a,0),(b,1),(c,1)\}, \\ & \{(a,1),(b,0),(c,0)\}, \quad \{(a,1),(b,0),(c,1)\}, \\ & \{(a,1),(b,1),(c,0)\}, \quad \{(a,1),(b,1),(c,1)\} \ \}. \end{aligned}$$

Il y a donc 8 fonctions de A vers B. Observons que l'ensemble $\mathsf{Fonct}[\emptyset, B]$ contient exactement un élément, quel que soit l'ensemble B. C'est la relation vide (qui est fonctionnelle par défaut), et on a explicitement

$$\mathsf{Fonct}[\emptyset, B] = \{\emptyset\}. \tag{1.14}$$

En utilisant ceci comme condition initiale, on peut calculer récursivement l'ensemble des fonctions de A vers B lorsque A et B sont des ensembles finis. On a

$$f \in \mathsf{Fonct}[A, B] \quad \text{ssi} \quad \begin{cases} 1)\ f = \emptyset \text{ et } A = \emptyset, \\ 2)\ f = g + \{(x, y)\},\ \text{avec } x \in A,\ y \in B,\ \text{et} \\ \qquad g \in \mathsf{Fonct}[A \setminus \{x\}, B]. \end{cases} \tag{1.15}$$

Pour $f : A \to B$, et C un sous-ensemble de A, on a la *restriction* $f\big|_C$, de f à C, définie en posant

$$f\big|_C := \{(x, f(x)) \mid x \in C\}. \tag{1.16}$$

Il en résulte donc une fonction $f\big|_C : C \to B$.

Quel que soit B, sous-ensemble de A, on peut définir la *fonction caractéristique*, $B : A \to \{0, 1\}$, de B dans A, en posant

$$B(x) := \begin{cases} 1 & \text{si, } x \in B, \\ 0 & \text{sinon.} \end{cases} \tag{1.17}$$

Cette fonction caractérise le sous-ensemble B par le fait que

$$B = \{x \in A \mid B(x) = 1\}, \quad \text{et} \quad \overline{B} = \{x \in A \mid B(x) = 0\}. \tag{1.18}$$

Dans le cas où R est une relation de $[n]$ vers $[k]$, la fonction caractéristique $R : [n] \times [k] \to \{0, 1\}$ correspondante

$$R(i, j) := \begin{cases} 1 & \text{si, } (i, j) \in R, \\ 0 & \text{si, } (i, j) \notin R, \end{cases} \tag{1.19}$$

est *une matrice* $n \times k$ dont les coefficients sont des 1 ou des 0. On dit que R est la *matrice d'incidence* de la relation R. Par exemple, avec $n = 3$ et $k = 4$, la relation

$$R = \{(1, 1), (1, 3), (1, 4), (2, 2), (2, 3), (3, 1), (3, 4)\}$$

correspond ainsi à la matrice

$$R = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

Dans le cas d'une relation fonctionnelle f, la matrice obtenue contient exactement un seul 1 sur chacune de ses lignes.

Cette dernière construction est un cas spécial de fonction de la forme $M := [n] \times [k] \to A$, à valeur dans un ensemble A quelconque. Une telle fonction s'identifie naturellement à une matrice $n \times k$ dont les coefficients sont choisies dans l'ensemble A, simplement en prenant $M(i,j)$ comme valeur en position (i,j) dans la matrice. Ainsi, pour $A = \{a,b,c,d,e\}$, on a la matrice 2×3

$$\begin{pmatrix} a & b & c \\ c & a & e \end{pmatrix}$$

qui correspond à la fonction sur $[2] \times [3]$, prenant les valeurs

$$\begin{array}{lll} M(1,1) = a, & M(1,2) = b, & M(1,3) = c, \\ M(2,1) = c, & M(2,2) = 1, & M(2,3) = e. \end{array}$$

1.7.1 Bijections

Dans une première introduction à la théorie des ensembles, la correspondance établie par une bijection $f : A \to B$, entre les éléments de A et ceux de B, est souvent introduite par une représentation naïve comme celle de la Figure 1.1. De façon plus précise, on a la définition suivante. Une fonction f de A vers B, est une *bijection*, si on a une fonction *inverse* $f^{-1} : B \to A$, pour la composition, i.e. :

$$f^{-1} \circ f = \mathrm{Id}_A, \quad \text{et} \quad f \circ f^{-1} = \mathrm{Id}_B. \tag{1.20}$$

Il est très facile de vérifier qu'il ne peut y avoir qu'un inverse pour la composition, i.e. :

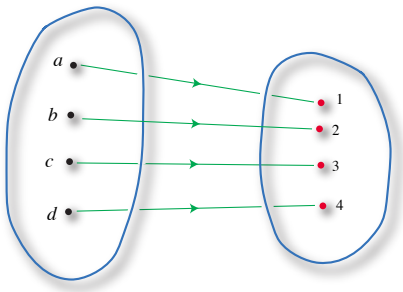


FIGURE 1.1 – Représentation naïve d'une bijection.

Proposition 1.1. *Pour toute fonction $f : A \to B$, si $g : B \to A$ est telle que*

$$g \circ f = \mathrm{Id}_A, \quad \text{et} \quad f \circ g = \mathrm{Id}_B, \tag{1.21}$$

alors $g = f^{-1}$.

Pour montrer que $f : A \to B$ est une bijection, il faut donc montrer qu'on peut construire une fonction qui satisfait (1.21).

On désigne par $\mathsf{Bij}[A,B]$ l'ensemble (fini) des relations bijectives entre A et B, i.e. :

$$\mathsf{Bij}[A,B] := \{f \in \mathsf{Fonct}[A,B] \mid f : A \xrightarrow{\sim} B\}. \tag{1.22}$$

Il découle directement du principe ci-haut qu'on a $\mathsf{Bij}[A,B]=\emptyset$, lorsque $|A|\neq|B|$. Observons que $\mathsf{Bij}[\emptyset,\emptyset]=\{\emptyset\}$ est de cardinal 1. La formule suivante permet de calculer récursivement l'ensemble des bijections entre deux ensemble A et B de même cardinal fini :

$$f\in\mathsf{Bij}[A,B] \quad \text{ssi} \quad \begin{cases} 1)\ f=\emptyset \text{ et } A=B=\emptyset, \\ 2)\ f=g+\{(x,y)\},\ \text{avec } x\in A,\ y\in B,\ \text{et} \\ \qquad\qquad g\in\mathsf{Bij}[A\setminus\{x\},B\setminus\{y\}]. \end{cases} \tag{1.23}$$

Ainsi, pour $A=\{a,b,c\}$ et $B=\{1,2,3\}$, on trouve

$$\begin{aligned}\mathsf{Bij}[A,B]=\{\ & \{(a,1),(b,2),(c,3)\},\quad \{(a,1),(b,3),(c,2)\},\\ & \{(a,2),(b,1),(c,3)\},\quad \{(a,2),(b,3),(c,1)\},\\ & \{(a,3),(b,1),(c,2)\},\quad \{(a,3),(b,2),(c,1)\}\ \}.\end{aligned}$$

Il y a donc exactement 6 bijections entre les deux ensembles à trois éléments A et B.

Le composé de fonctions bijectives est une fonction bijective, et l'inverse d'une fonction bijective est une fonction bijective. Pour tout ensemble fini A, l'ensemble $\mathsf{Bij}[A,A]$, des bijections de A vers A, forme un groupe pour la composition de fonctions, avec Id_A comme identité. On dit habituellement d'une bijection de A vers A que c'est une *permutation* de A, et on désigne souvent par $\mathbb{S}_A$ l'ensemble des permutations de A. Lorsque $A=[n]$, on écrit simplement $\mathbb{S}_n$ plutôt que $\mathbb{S}_{[n]}$. Les permutations sont souvent dénotées par des lettres grecques minuscules : σ, τ, θ, etc.

On code souvent une permutation σ, de $[n]$, sous forme d'une matrice carrée $n\times n$ de 0 et de 1, ayant un 1 dans chaque ligne, et un 1 dans chaque colonne. On dit que c'est une *matrice de permutation*. C'est en fait la fonction caractéristique de la relation fonctionnelle sous-jacente.

1.7.2 Injections

Parmi les propriétés particulières des fonctions, l'injectivité et la surjectivité sont très certainement des notions importantes. Une fonction $f : A \to B$ est dite *injective* si et seulement si

« *Pour chaque élément y de B, il existe au plus un élément x de A tel que $f(x)=y$.* »

Autrement dit, la fonction f et un processus qui

« *choisit des éléments $f(x)$ de B, un pour chaque x dans A, tous distincts,* »

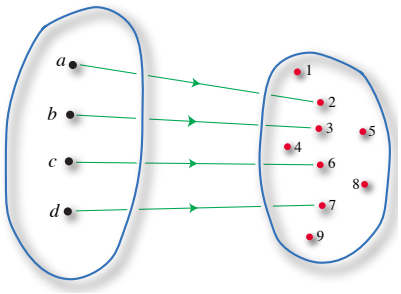


FIGURE 1.2 – Représentation naïve d'une injection.

c'est-à-dire qu'un élément ne peut-être choisi qu'une seule fois. Une formulation un peu plus technique (mais plus facile à manipuler) de cette définition prend la forme suivante. Une fonction $f : A \to B$ est injective si et seulement si, pour tout x_1 et tout x_2 dans A

$$x_1 \neq x_2 \quad \text{implique} \quad f(x_1) \neq f(x_2), \tag{1.24}$$

ce qui équivaut (logiquement) à dire aussi que

$$f(x_1) = f(x_2) \quad \text{entraine forcément} \quad x_1 = x_2. \tag{1.25}$$

Il en découle (voir Exercice **??**) que f est injective si et seulement si f admet un *inverse à gauche*, i.e. : il existe $g : B \to A$ tel que $g \circ f = \mathrm{Id}_A$. En général, un tel inverse à gauche n'est pas unique, et il n'est pas un inverse à droite.

On dénote souvent par le symbole « $\hookrightarrow$ » le fait qu'une fonction soit injective. On écrit donc $f : A \hookrightarrow B$, pour dire que f est une injective. On désigne par $\mathsf{Inj}[A, B]$ l'ensemble des relations fonctionnelles f telles que $f : A \hookrightarrow B$ soit une fonction injective de A vers B, i.e. :

$$\mathsf{Inj}[A, B] := \{f \in \mathsf{Fonct}[A, B] \mid f : A \hookrightarrow B\}. \tag{1.26}$$

On calcule récursivement l'ensemble des fonctions injectives de A vers B lorsque A et B sont des ensembles finis. On a

$$f \in \mathsf{Inj}[A, B] \quad \text{ssi} \quad \begin{cases} (1)\ f = \emptyset \text{ et } A = \emptyset, \\ (2)\ f = g + \{(x, y)\}, \text{ avec } x \in A,\ y \in B, \text{ et} \\ \qquad g \in \mathsf{Inj}[A \setminus \{x\}, B \setminus \{y\}]. \end{cases} \tag{1.27}$$

Observons que la seule différence entre cette description et celle en (1.23) est dans la partie (1). La ressemblance n'est pas fortuite. Elle s'ensuit du fait qu'une injection entre deux ensemble de même cardinal est forcément une bijection (voir Proposition 1.2). Pour $A = \{a, b\}$ et $B = \{1, 2, 3\}$, on a

$$\begin{aligned} \mathsf{Inj}[A, B] = \{ \quad & \{(a, 1), (b, 2)\}, \quad \{(a, 2), (b, 1)\}, \\ & \{(a, 1), (b, 3)\}, \quad \{(a, 3), (b, 1)\}, \\ & \{(a, 2), (b, 3)\}, \quad \{(a, 3), (b, 2)\} \quad \}. \end{aligned}$$

Pour qu'il existe un injection $f : A \hookrightarrow B$, le nombre d'éléments de A doit nécessairement être plus petit ou égal à celui de B, i.e. : $|A| \leq |B|$. On a donc $\mathsf{Inj}[A, B] = \emptyset$, lorsque $|A| > |B|$.

Le composé de deux fonctions injectives est toujours une fonction injective. De plus, si on a deux fonctions telles que le composé $g \circ f$ soit une fonction injective, alors f est forcément injective (mais pas nécessairement g).

1.7.3 Surjections

Une fonction $f : A \to B$ est dite *surjective* si et seulement si

« *Pour chaque élément y de B, il existe au moins un élément x de A tel que $f(x) = y$.* »

Autrement dit, f est un processus qui

« *choisit chaque élément y de B au moins une fois.* »

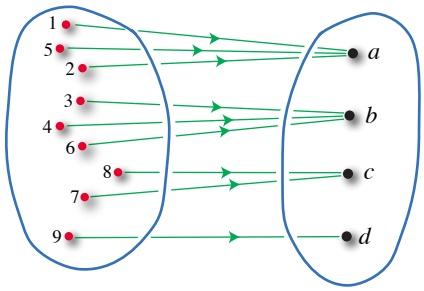


FIGURE 1.3 – Représentation naïve d'une surjection.

Une fonction $f : A \to B$ est surjective si et seulement si elle admet un *inverse à droite*, i.e. : il existe $g : B \to A$ tel que $f \circ g = \mathrm{Id}_B$. En général, un tel inverse à droite n'est pas unique, et il n'est pas un inverse à gauche si f n'est pas bijective. On dénote souvent par le symbole « $\twoheadrightarrow$ » le fait qu'une fonction est surjective. On écrit alors $f : A \twoheadrightarrow B$, pour dire que f est une surjective. On désigne par $\mathsf{Surj}[A, B]$ l'ensemble des relations fonctionnelles surjectives de A vers B, i.e. :

$$\mathsf{Surj}[A, B] := \{f \in \mathsf{Fonct}[A, B] \mid f : A \twoheadrightarrow B\}. \tag{1.28}$$

Pour qu'il existe une surjection $f : A \twoheadrightarrow B$, le nombre d'éléments de A doit nécessairement être plus grand ou égal à celui de B, i.e. : $|A| \geq |B|$. On a donc $\mathsf{Surj}[A, B] = \emptyset$, lorsque $|A| < |B|$. On peut calculer récursivement l'ensemble des fonctions surjectives entre deux ensembles fins A et B par la récurrence

$$f \in \mathsf{Surj}[A, B] \quad \text{ssi} \quad \begin{cases} (1)\ f = \emptyset \text{ et } A = B = \emptyset, \\ (2)\ f = g + h, \text{ avec } C \subseteq A,\ y \in B, \\ \qquad g \in \mathsf{Surj}[A \setminus C, B \setminus \{y\}], \text{ et} \\ \qquad h \in \mathsf{Fonct}[C, \{y\}]. \end{cases} \tag{1.29}$$

Le composé de deux fonction surjectives est toujours une fonction surjective. De plus, si on a deux fonctions telles que le composé $g \circ f$ soit une fonction surjective, alors g est forcément surjective (mais pas nécessairement f).

On montre facilement que

Proposition 1.2. *Une fonction qui est à la fois surjective et injective, est une fonction bijective. Autrement dit,*

$$\mathsf{Bij}[A, B] = \mathsf{Inj}[A, B] \cap \mathsf{Surj}[A, B]. \tag{1.30}$$

1.8 Compter les éléments d'un ensemble

Lorsqu'on cherche à compter les éléments d'un ensemble A, le problème se décompose souvent en un (ou des) problème(s) plus simple(s), selon que l'ensemble à énumérer peut se décrire en terme des constructions de base sur les ensembles.

On démarre les choses en montrant (par récurrence) que

Théorème 1.3. *Si A et B sont des ensembles finis, respectivement de cardinal n et k, alors on a les égalités suivantes :*

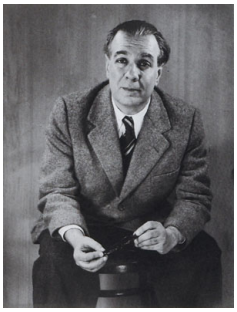
$$\begin{aligned} &\text{(i) } |A+B| = n+k, && \text{(ii) } |A\times B| = n\,k,\\ &\text{(iii) } |\mathcal{P}[A]| = 2^n, && \text{(iv) } |\mathcal{P}_k[A]| = \binom{n}{k},\\ &\text{(v) } |B^n| = k^n. \end{aligned} \tag{1.31}$$

Le problème de compter les éléments d'un ensemble fini est parfois difficile, et même dans certains cas pas encore résolu. Il donne lieu à un domaine des mathématiques qu'on appelle la *combinatoire énumérative*. C'est l'un des domaines de recherche dans lequel des professeurs du département de mathématiques de l'UQAM se sont spécialisés. Il sont à ce titre très reconnus sur la scène internationale. Pour en savoir plus à ce sujet, on peut consulter la page web du centre de recherche institutionnel « *Lacim* » :

http://www.lacim.uqam.ca/

La bibliothèque de Borges (*)

L'écrivain argentin José Luis Borges propose une vision amusante de la *Bibliothèque de Babel* (*La Biblioteca de Babel*) qui contient tous les livres (il y en a un nombre fini) de 410 pages qu'il est possible d'écrire avec les 25 lettres d'un certain alphabet. Cette nouvelle, traduite en plusieurs langues, dont le français, décrit l'existence des habitants de cette bibliothèque, qu'ils ne quittent jamais. Les livres y sont disposés au hasard, et chacun des habitants est à la recherche d'un livre particulièrement important pour lui : *le catalogue des catalogues*, *la description de toute sa vie, passée et future*, *la description de l'origine de la bibliothèque*, etc.



José Luis Borges
(1906–1978)

On estime qu'il y a de l'ordre de 10^{80} atomes dans la partie « observable » de notre univers. Cependant, si on suppose qu'il y a 1000 caractères par page d'un livre de la bibliothèque de Borges, alors le nombre de livres est

$$25^{410{,}000},$$

ce qui est quelque peu plus grand que le nombre d'atomes dont il est question. Il est donc un peu difficile de trouver assez de place (et de matériel) pour ériger la bibliothèque de Borges dans notre univers.

Le paradoxe de Russell

La théorie des ensembles telle que formulée par Cantor n'était pas assez précise. Bertrand Russell l'a mis en évidence en soulignant qu'elle donnait lieu au paradoxe suivant. On considère l'ensemble « de tous les ensembles qui ne se contiennent pas eux-même ». En formule, c'est

$$A = \{x \mid x \notin x\}. \tag{1.32}$$

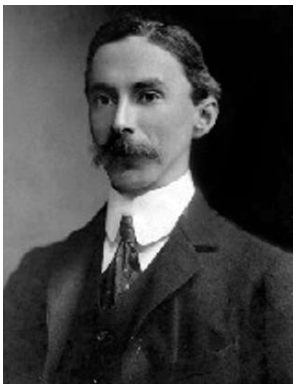
La question qui se pose est de savoir si

$$A \in A \qquad \text{où} \qquad A \notin A.$$

Hors on constate (avec Russel) que

$$A \in A \qquad \text{implique} \qquad A \notin A,$$

et réciproquement ! C'est là le paradoxe. Attention, on considère ici la notion d'ensemble à la Cantor. Cette constuction n'est pas possible dans les versions modernes de la théorie des ensembles. En effet, pour remédier au paradoxe de Russel, on a échafaudé plusieurs axiomatiques précises pour la théorie des ensemble. L'une des plus connue est celle dite de Zermelo-Fraenkel présentée schématiquement à l'appendice B. C'est dans de tels contextes que les mathématiciens travaillent maintenant.



Bertrand Russell
(1872-1970)

Annexe A

Un soupçon de logique

Pour développer des preuves, il est nécessaire de connaître les manipulations logiques de base. Nous allons ici en présenter quelques principes. Cependant, notre présentation est très sommaire et un peu trop formelle pour une première exposition à la logique mathématique. Le lecteur est encouragé à consulter un livre d'introduction plus accessible.

Informellement, un *énoncé* (mathématiques) est une phrase qui affirme un certain fait (mathématique). L'important est de pouvoir déterminer si l'énoncé est « vrai » ou « faux ». Une preuve est constituée d'un enchaînement de déductions logiques, à partir de faits connus (ou d'axiomes), avec comme conclusion le fait que l'énoncé (qu'on cherchait à prouver) est vrai. La forme d'une preuve dépend fortement de la forme de l'énoncé à prouver. On décrit ces formes possibles ci-dessous.

À partir d'énoncés connus A et B, on peut former de nouveaux énoncés au moyen d'*opérations logiques*. On a les énoncés :

1. (A et B), qui est vrai si et seulement si A et B le sont tous les deux,
2. (A ou B), qui est vrai si et seulement si A est vrai ou B est vrai,
3. ($\neg A$), qui est vrai si et seulement si A est faux,
4. ($A \Rightarrow B$), qui n'est faux que lorsque A est faux et B est vrai,
5. ($A \Leftrightarrow B$), qui est vrai exactement quand A et B sont tous les deux vrai, ou tous les deux faux.

Dans la description de chacune des opérations on décrit quel est la façon de procéder pour prouver l'énoncé composé, à partir de ses composantes. L'opération d'*équivalence* « $\Leftrightarrow$ » permet de comparer la véracité d'énoncés logique. Ainsi, l'énoncé ($A \Leftrightarrow B$) se formule aussi

$$A \text{ si et seulement si } B,$$

ou même (A ssi B). Ce sont diverse façons d'exprimer le fait que A soit vrai est équivalent au fait que B le soit. L'opération d'*implication* « $\Rightarrow$ » correspond à la déduction logique. Ainsi, l'énoncé

$(A \Rightarrow B)$ se formule aussi

si A alors B.

Ce sont diverse façons d'exprimer le fait que B soit vrai se déduit du fait que A l'est. Autrement dit, une preuve de $(A \Rightarrow B)$ pourra se dérouler comme suit. On suppose que A est vrai, et on montre qu'un enchaînement logique de déductions prouve que B l'est alors forcément.

Dans une série de manipulations logiques, il est agréable de savoir quand on peut remplacer une affirmation par une autre qui lui est logiquement équivalente (en espérant qu'elle soit plus facile à montrer). Les règles du *calcul des propositions* expliquent quand il est possible de remplacer un énoncé par un énoncé qui lui est logiquement équivalent. On a par exemple :

1. $\neg(A$ et $B)$ si et seulement si $(\neg A$ ou $\neg B)$,
2. $\neg(A$ ou $B)$ si et seulement si $(\neg A$ et $\neg B)$,
3. $(A \Rightarrow B)$ si et seulement si $(\neg B \Rightarrow \neg A)$,
4. $(A \Rightarrow B)$ si et seulement si $(\neg A$ et $B)$,
5. $(A \Leftrightarrow B)$ si et seulement si $(A \Rightarrow B)$ et $(B \Rightarrow A)$

Chacun de ces cas représente une stratégie potentielle de preuve, si l'énoncé est de la bonne forme. Cette liste est incomplète, mais elle comprend les principales stratégies usuelles.

Certains énoncés font intervenir une *variable*, et sont vrais pour certaines valeurs de cette variable. On écrit $P(x)$ pour ce genre d'énoncés, avec x la variable, et on dit qu'on a un *prédicat*. Informellement, c'est un phrase avec x comme sujet. Typiquement, on pense à P comment étant une propriété que x peut avoir (ou pas). Par exemple, on a

- $P(x) = (x$ est un nombre pair$)$, ou
- $P(x) = (x$ est égal à 1$)$,
- etc.

Pour chaque a, valeur possible[1] de x, on a un énoncé $P(a)$ qui est vrai si et seulement si a « possède » la propriété P. À partir de prédicats donnés, on peut former de nouveaux prédicats au moyen des opérations logiques : $(P(x)$ et $Q(x))$, $(P(x)$ ou $Q(x))$, $(\neg P(x))$, etc. Bien entendu, on a des prédicats à plusieurs variables, prenant la forme $P(x, y)$ par exemple.

Au moyen d'un prédicat, on peut former les énoncés logiques

1. $(\forall x\ P(x))$, qui est vrai si et seulement si $P(a)$ est vrai pour toutes la valeurs possibles de x,
2. $(\exists x\ P(x))$, qui est vrai si et seulement si il existe a, une valeur possible de x, pour laquelle $P(a)$ est vrai.

On a les équivalences logiques

1. $\neg(\forall x\ P(x))$ si et seulement si $(\exists x\ \neg P(x))$,
2. $\neg(\exists x\ P(x))$ si et seulement si $(\forall x\ \neg P(x))$.

1. Dans un ensemble A donné.

Annexe B

Axiomatique de la théorie des ensembles

La présentation ci-dessous ne vise qu'à donner une idée de ce à quoi peut ressembler une théorie axiomatique des ensembles. Le but visé est simplement de montrer qu'il existe une (des) axiomatique rigoureuse pour la notion d'ensemble. Dans un premier temps, le lecteur est encouragé à simplement survoler la description qui suit. Pour en savoir plus, il faudra suivre un cours sur le sujet, ou consulter un livre plus spécialisé, comme

J.-L.Krivine, *Théorie axiomatique des ensembles*, Presses Universitaires de France, 1969.

Il existe plusieurs systèmes axiomatiques formels pour la théorie des ensembles. L'un des plus connu est le système *ZFC* de Zermelo-Fraenkel (avec l'axiome du choix). L'axiomatique ZFC se décrie dans le contexte du calcul des prédicats avec relation d'égalité. Toute la theórie étant formulé en terme d'ensembles, on doit se rappeler que les éléments d'ensembles sont aussi des ensembles. Tout est, en quelque sorte, construit à partir de l'ensemble vide. Ainsi on aura les ensembles tous distincts

$$\begin{array}{cccc} \emptyset, & \{\emptyset\}, & \{\{\emptyset\}\}, & \dots \\ \{\emptyset,\{\emptyset\}\}, & \{\{\emptyset,\{\emptyset\}\}\}, & \{\{\{\emptyset,\{\emptyset\}\}\}\}, & \dots \\ \{\emptyset,\{\emptyset\},\{\{\emptyset\}\}\}, & \{\{\emptyset,\{\emptyset\},\{\{\emptyset\}\}\}\}, & \{\{\{\emptyset,\{\emptyset\},\{\{\emptyset\}\}\}\}\}, & \dots \\ \vdots & \vdots & \vdots & \ddots \end{array} \tag{B.1}$$

La relation d'appartenance $x \in A$ et la notion d'ensemble ne sont définies que par le fait qu'elle satisfont les axiomes suivants. La relation d'inclusion $A \subseteq B$ est définie par

$$(A \subseteq B) \qquad \text{ssi} \qquad \forall x\ (x \in A \Rightarrow x \in B).$$

1) Axiome d'extensionalité. Deux ensembles sont égaux, si et seulement si ils ont les mêmes éléments. En formule,

$$\forall A\ \forall B\ [\forall x\ (x \in A \Leftrightarrow x \in B) \Rightarrow (A = B)]. \tag{B.2}$$

2) Axiome de la paire. Pour tous ensembles A et B, on peut construire un ensemble C dont les seuls éléments sont A et B. Autrement dit, on permet ici de construire

$$C := \{A, B\}.$$

En formule,

$$\forall A\ \forall B\ \exists C[\forall x\ (x \in C) \Leftrightarrow (x = A \ \text{ ou } x = B)]. \tag{B.3}$$

3) Axiome de la réunion. Pour tout ensemble A, on peut construire un ensemble B dont les seuls éléments sont tous ceux qui sont éléments des éléments de A. Autrement dit, on permet ici la construction de l'ensemble

$$B := \bigcup_{x \in A} x.$$

En formule,

$$\forall A\ \forall B\ \exists C[\forall x\ (x \in C) \Leftrightarrow (x = A \ \text{ ou } x = B)]. \tag{B.4}$$

4) Axiome de l'ensemble des parties. Pour tout ensemble A, on peut construire l'ensemble B des sous-ensembles de A. Autrement dit, on permet ici la construction de l'ensemble

$$B := \{x \mid x \subseteq A\}.$$

En formule,

$$\forall A\ \exists B\ \forall x (x \in B) \Leftrightarrow (x \subseteq A). \tag{B.5}$$

5) Axiome de l'infini. Cet axiome permet de construire (au moins un) ensemble infini. C'est l'ensemble

$$\begin{aligned} A := \{ \quad & \emptyset, \\ & \{\emptyset\}, \\ & \{\emptyset, \{\emptyset\}\}, \\ & \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}, \\ & \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}\}, \ \ldots\}. \end{aligned}$$

Cet axiome permet aussi (avec les précédents) de construire l'ensemble $\mathbb{N}$ des entiers positifs. En formule,

$$\exists A\ (\emptyset \in A) \quad \text{et} \quad (x \cup \{x\} \in A). \tag{B.6}$$

6) Shéma d'axiomes de compréhension. Pour tout ensemble A, on peut construire le sous-ensemble B des éléments de A qui satisfont une propriété P (exprimée dans le langage de la théorie des ensembles). Autrement dit, on permet ici la construction de

$$B := \{x \in A \mid P(x)\}.$$

En formule,

$$\forall y_1 \ \cdots \ \forall y_n \ \forall A \ \exists B \ \forall x[(x \in B) \Leftrightarrow (x \in A \text{ et } P(x, y_1, \ldots, y_n)). \tag{B.7}$$

Les y_i sont ici simplement des paramètres auxiliaires dont on pourrait avoir besoin pour formuler plus facilement la propriété P. On dit qu'on a un « schéma » d'axiomes, parce qu'il y a un axiome pour chaque choix de P.

7) Shéma d'axiomes de remplacement. Pour tout ensemble A et toute relation fonctionnelle F, on a un ensemble

$$B := \{y \mid x \in A \text{ et } F(x, y)\}.$$

Pour exprimé ceci en formule (simplifiée [1]), rappelons d'abord que F est une relation fonctionnelle, on écrit $\mathsf{Fonct}(F)$, si et seulement si

$$\mathsf{Fonct}(F) \qquad \text{ssi} \qquad \forall x \ \forall y_1 \ \forall y_2 \ [(F(x, y_1) \text{ et } F(x, y_2)) \Rightarrow (y_1 = y_2)].$$

Alors l'axiome se présente comme

$$\mathsf{Fonct}(F) \Rightarrow \forall A \ \exists B \ \forall y \ [y \in B \Leftrightarrow \exists x \ (x \in A \text{ et } F(x, y))].$$

8) Axiomes de fondation. Pour tout ensemble A non vide, il existe un ensemble B, appartenant à A, qui n'a aucun élément en commun avec A, c'est-à-dire que

$$A \cap B = \emptyset.$$

En formule,

$$\forall A[(A \neq \emptyset) \text{ et } \exists B \ (B \in A \text{ et } A \cap B = \emptyset)]. \tag{B.8}$$

9) Axiomes du choix. Pour tout ensemble A, d'ensembles non vide, le produit cartésien des éléments de A est non vide. En formule,

$$[\forall x \in A \ (x \neq \emptyset)] \Rightarrow \prod_{x \in A} x \neq \emptyset. \tag{B.9}$$

1. La formulation plus juste fait apparaître des paramètres dans F comme dans l'axiome précédent.

Annexe C

Calcul formel

C.1 Introduction

Les systèmes de calcul formel permettent de manipuler concrètement des objets mathématiques abstraits de façon rigoureuse. Cela va des nombres entiers, rationnels, réels ou complexes (et des calculs sur ceux-ci) ; à des manipulations d'opérateurs sur des espaces de fonctions ; en passant par un vaste spectre de notions mathématiques de l'algèbre, du calcul, de l'analyse complexe, de la théorie des nombres, etc.

Une façon très efficace d'apprivoiser de nouvelles notions mathématiques est d'apprendre à les manipuler avec de tels systèmes de calcul formels. Nous encourageons donc fortement les étudiants en mathématiques à se familiariser avec ces systèmes pour les accompagner dans tout leur apprentissage.

Dans un système de calcul formel, une session de travail est habituellement un processus interactif qui consiste à donner au système une instruction de calcul (apparaissant en rouge dans ce qui suit). On obtient alors comme résultat la valeur du calcul demandé (apparaissant en bleu). Dans notre cas, le système affiche automatiquement le symbole « > » chaque fois qu'il est prêt à effectuer une prochaine instruction. Le « ; » signifie la fin de l'instruction donnée, et la touche « return » (ou « enter ») déclenche le calcul. Ainsi, on peut demander de calculer

$> \gcd(x^{36} - 1, x^{24} - 1)\,;$

$$x^{12} - 1$$

Ici, la fonction Maple « gcd » trouve que $x^{12} - 1$ est un [1] plus grand commun diviseur des polynômes $x^{36} - 1$ et $x^{24} - 1$.

1. Puisque définit à un multiple scalaire prêt.

En plus d'effectuer des calculs explicites, il est possible de donner des noms à certains objets au moyen l'assignation « := ». On peut donc poser

> $x := 100$;

$$x := 100$$

Dorénavant, x aura la valeur 100, et l'expression 2^x prendra donc la valeur :

> 2^x ;

$$1267650600228229401496703205376$$

Il faut bien distinguer cette assignation de la relation mathématique usuelle « = » d'égalité.

Pour les fins d'une utilisation vraiment efficace des système de calculs formels, la capacité qui est de loin la plus importante est la possibilité de définir de nouvelles fonctions (ou procédures) de calcul, à partir de celles déjà connues. On a ainsi un riche environnement de programmation spécialisé pour les mathématiques. C'est nouvelles fonctions peuvent se définir de nombreuses façons, mais celle qui est la plus naturelle est probablement via la récursivité. Ainsi on peut introduire la fonction :

```
> F := n->
      if n ≤ 1 then 1
           else F(n − 1) + F(n − 2)
      fi :
```

Dorénavant, « F » est la fonction d'une variable qui calcule (récursivement) les nombres de Fibonacci. On aura donc :

> $F(20)$;

$$10946$$

Les changement de lignes et les espaces supplémentaires n'ont ici aucun effet sur le calcul. Ils ne servent qu'à disposer la description de la fonction « F » de manière plus agréable. Normalement, le résultat d'une telle instruction est d'afficher le texte de la fonction ainsi définie. Le fait d'utiliser le « : », plutôt que le « ; » comme indication de fin de l'instruction, évite cet affichagee assez inutile.

Nous allons illustrer dans cette annexe comment il est facile d'utiliser de tels systèmes pour manipuler les objets combinatoires que nous avons rencontrés dans ce texte.

C.2 Théorie des ensembles et calcul formel

Tout système de calcul formel (Sage, Maple, etc.) permet, entre autres, de manipuler des ensembles, des listes, et diverses constructions les concernant (avec les adaptations nécessaires).

Ainsi, on peut donner le nom A à l'ensemble $\{a, b, c\}$ en posant

$> A := \{a, b, a, c, b, b\}$;

$$A := \{a, b, c\}$$

Observons ici (comme le veut la théorie) que la répétition d'un élément n'a aucun effet. Bien entendu, on a aussi les opérations usuelles sur les ensembles

$> \{a, b, c\}$ intersect $\{b, c, d\}$;

$$\{b, c\}$$

$> \{a, b, c\}$ union $\{b, c, d\}$;

$$\{a, b, c, d\}$$

$> \{a, b, c\}$ minus $\{b, c, d\}$;

$$\{a\}$$

En passant, rien n'empêche d'utiliser abstraitement ces opérations sur des ensembles B et C non spécifiés. Ainsi, on obtient

$> (B$ union $D)$ intersect $(C$ union $D)$;

$$(B \cup D) \cap (C \cup D)$$

$> B$ minus B ;

$$\emptyset$$

On peut tester l'égalité d'ensembles, et l'appartenance à un ensemble, en exploitant la fonction « evalb » qui calcule la valeur logique d'une expression. Ainsi, on a

$>$ evalb$(\{a, b, c\} = \{b, c, d\})$;

false

$>$ evalb$(\{a, b, c\} = \{b, c, a\})$;

true

$> b$ in $\{b, c, a\}$;

$$b \in \{a, b, c\}$$

$>$ evalb$(b$ in $\{b, c, a\})$;

true

La fonction « nops » est très générale. Elle permet de compter le nombre d'opérandes dans une expression. En particulier elle donne le nombre d'éléments d'un ensemble. Cependant, pour faciliter la compréhension, il est possible de lui donner un synonyme en posant :

```
> card := nops :
```

Par la même occasion, on se permet de mettre en place les autres synonymes :

```
> `&+` := `union` :
> `&-` := `minus` :
```

L'utilisation du caractère « & », dans « &+ », est nécessaire en Maple lorsqu'on désire considérer de nouveaux opérateurs binaires avec une notation « infixe ». C'est aussi une particularité de la syntaxe de Maple qui forcent l'utilisation des « ` » au moment de l'établissement de ces synonymes.

Pour décrire un ensemble de la forme $\{g(x) \mid x \in A\}$, on peut utiliser la fonction Maple « seq » qui permet de construire des séquences de valeurs $g(x)$ pour x variant dans A. Puisque l'ensemble A a déjà été défini (mais pas g), on obtient :

```
> {seq(g(x), x in A)} ;
```

$$\{g(a), g(b), g(c)\}$$

ou encore, avec la fonction de Fibonacci F :

```
> {seq(F(x), x in {1,2,3,4,5,6,7,8,9})} ;
```

$$\{1, 2, 3, 5, 8, 13, 21, 34, 55\}$$

Autres opérations. On peut définir d'autres opérations usuelles sur les ensembles comme ci-dessous. Si $A = \{x\} + B$, le calcul de l'ensemble $\mathcal{P}[A]$ des parties de A est basé sur la récurrence :

$$\mathcal{P}[A] = \mathcal{P}[B] \ + \ \{C + \{x\} \mid C \in \mathcal{P}[B]\}. \tag{C.1}$$

Avec une syntaxe légèrement différente de celle déjà utilisée, on exploite cette récurrence pour obtenir la nouvelle fonction

```
> P := proc(A) local x, B :
    if A = {} then {{}}
        else x := op(1, A) :                    (x est le premier élément dans A)
            B := A &- {x} :
            P(B) &+ {seq( C &+ {x}, C in P(B)}
    fi
end :
```

Après avoir considéré le cas spécial $A = \emptyset$, on choisit x comme étant le premier élément de A, et la cinquième ligne reproduit (presque fidèlement) le membre de droite de (C.1). On obtient ainsi une fonction calculant l'ensembles $P(A)$ des parties de A :

```
> P({a,b,c});
```

$$\{\emptyset, \{a\}, \{b\}, \{c\}, \{a,b\}, \{a,c\}, \{b,c\}, \{a,b,c\}\}$$

Avant de construire le produit cartésien comme opération binaire, on rappelle qu'en Maple un couple, normalement dénoté (x, y) dans des contextes mathématiques, est plutôt dénoté $[x, y]$. Ceci est du au fait que les parenthèses usuelles sont réservées à d'autres fins syntaxiques. On pose donc

```
> `&X` := proc(A,B) local x,y :
      {seq(seq([x,y] ,x in A),y in B)}
  end :
```

Alors, on obtient

```
> {a,b,c} &X {a,b};
```

$$\{[1,a],[1,b],[2,a],[2,b],[3,a],[3,b]\}$$

et aussi

```
> {a,b,c} &X {};
```

$$\emptyset$$

comme il se doit. On peut maintenant obtenir la fonction qui calcule l'ensemble des relations entre A et B, simplement en posant :

```
> Relations := (A,B)-> P(A &X B) :
```

et donc on peut ensuite calculer que

```
> Relations({a,b},{x,y});
```

$$\{\emptyset, \{[a,x]\}, \{[a,y]\}, \{[b,x]\}, \{[b,y]\}, \{[a,x],[a,y]\}, \{[a,x],[b,x]\}, \{[a,x],[b,y]\}, \{[a,y],[b,x]\},$$
$$\{[a,y],[b,y]\}, \{[b,x],[b,y]\}, \{[a,x],[a,y],[b,x]\}, \{[a,x],[a,y],[b,y]\}, \{[a,x],[b,x],[b,y]\},$$
$$\{[a,y],[b,x],[b,y]\}, \{[a,x],[a,y],[b,x],[b,y]\}\}$$

Pour calculer récursivement l'ensemble des fonctions entre A et B, il suffit de poser

```
> Fonct := (A,B)->
      if A = {} then {{}}
          else x := A[1] :
              {seq(seq(f &+ {x,y} ,y in B),f in Fonct(A &- {x},B))}
      fi :
```

et alors

```
> Fonct({a,b},{0,1});
```

$$\{\{[a,0],[b,0]\}, \{[a,0],[b,1]\}, \{[a,1],[b,0]\}, \{[a,1],[b,1]\}\}$$

Une construction similaire permet d'obtenir l'ensemble des mots de longueur k, sur un alphabet A, au moyen de la fonction Maple « cat » qui concatène deux mots.

```
> Mots := (A, k)->
      if k = 1 then A
           else {seq(seq(cat(w, x) , x in A), w in Mots(A, k - 1))}
      fi :
> Mots({a, b, c}, 3) ;
```

$$\{aaa, aab, aac, aba, abb, abc, aca, acb, acc,$$
$$baa, bab, bac, bba, bbb, bbc, bca, bcb, bcc,$$
$$caa, cab, cac, cba, cbb, cbc, cca, ccb, ccc\}$$

Annexe D

Notations

$|A|$ **:** le cardinal d'un ensemble A, Section 1.2.

$[n]$ **:** l'ensemble usuel de cardinal n, Section 1.2.

$\prod_{i\in I} A_i$ **et** $A \times B$ **:** produits cartésiens d'ensembles, Chapitre 1.

$\sum_{i\in I} A_i$ **et** $A + B$ **:** unions disjointes d'ensembles, Chapitre 1.

$\mathsf{Bij}[A, B]$ **:** l'ensemble des fonctions bijectives de A vers B, (1.22).

$f|_C$ **:** la fonction f restreinte au sous-ensemble C, (1.16).

f^k **:** k-ième itéré pour la composition de f, Section **??**.

$\mathsf{Fonct}[A, B]$ **:** l'ensemble des fonctions de A vers B, Chapitre 1.

$g \circ f$ **:** composé de fonctions g et de f, (**??**).

Id_A **:** la fonction identité sur A, Chapitre 1.

$\mathsf{Inj}[A, B]$ **:** l'ensemble des fonctions injectives de A vers B, (1.26).

$\mathcal{P}$ **:** l'ensemble des parties, (1.3).

$\mathcal{P}_k$ **:** parties à k-éléments, 1.3).

$\mathbb{S}_n$ **:** l'ensemble des permutations, (1.22).

$\mathsf{Surj}[A, B]$ **:** l'ensemble des fonctions surjectives de A vers B, (1.28).

χ_B **:** la fonction caractéristique d'un sous-ensemble B, (1.17).

Index