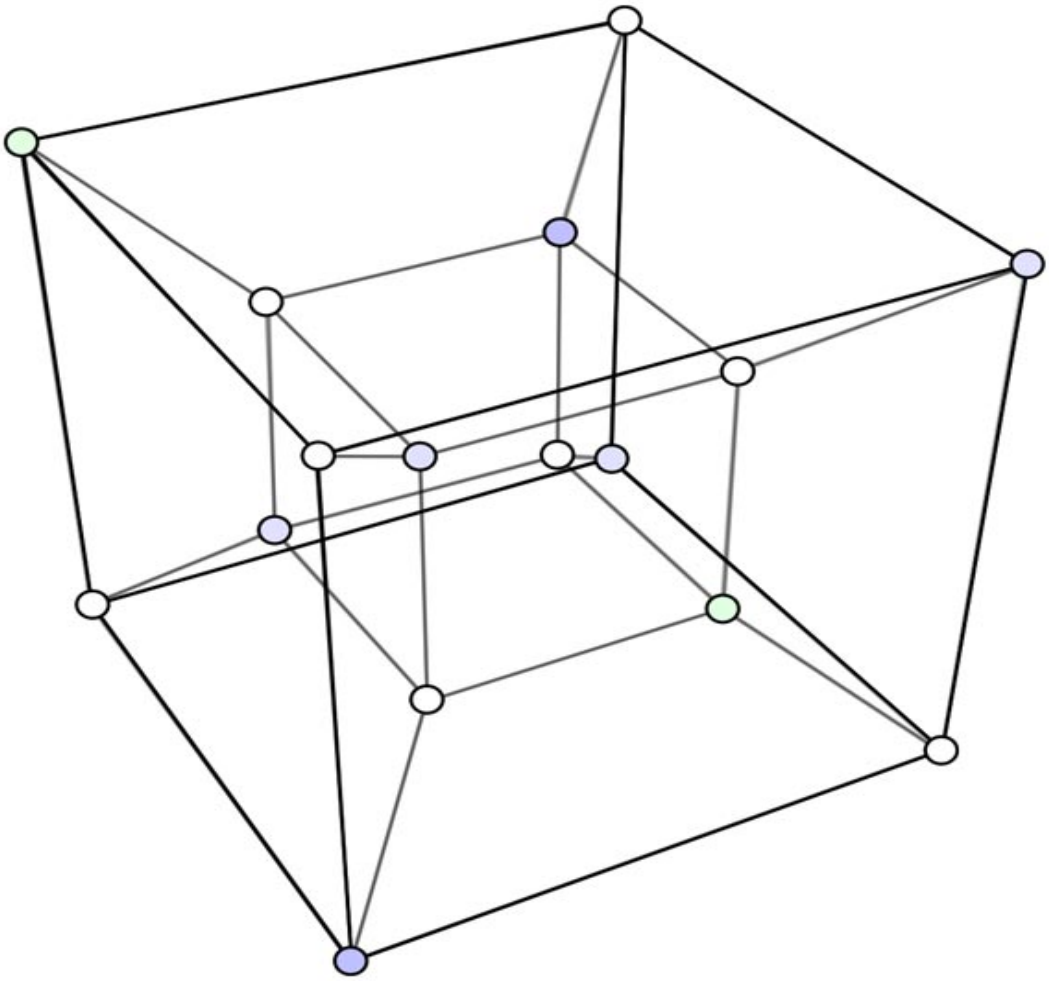
MT42

Fondements théoriques de l'informatique

Frédéric Holweck



Avant-propos

Ces notes reprennent les concepts et les résultats qui seront présentés en MT42 (sauf la partie logique qui sera dans un autre document). Les démonstrations ou encore les solutions des exercices seront détaillées en cours. Ce document est à vous et nous vous incitons à l'utiliser comme un outil de travail en cours, en TD et pendant les sessions d'examens. Vous pouvez l'anoter et même le compléter (ou le corriger). Nous vous recommandons fortement de ne pas attendre la veille des examens pour parcourir ces notes mais plutôt de les utiliser de manière active tout au long du semestre.

L'UV MT42 a longtemps été sous la responsabilité de J.-N. Martin. L'ancien polycopié, encore disponible sous Moodle, a servi de base à la rédaction de cette nouvelle version. La forme de la nouvelle version doit beaucoup (tout) aux très beaux packages LaTeX d'Alexis Flesch.

Ce semestre les enseignants de l'UV sont :

- Frédéric Holweck (CM, TD, resp. UV)
- Fabrice Lauri (CM, TD pour le chapitre logique)

Pour nous joindre, contact : prenom.nom@utbm.fr.

Table des matières

Ce premier chapitre a pour objet d'introduire le concept d'induction (ou induction structurelle) qui est une généralisation multidimensionnelle du principe de récurrence.

I Principe de récurrence

Le principe de récurrence repose sur les propriété de l'ensemble des entiers naturels $\mathbb{N}$, en particulier sur la propriété fondametale suivante, *toute partie non vide de* $\mathbb{N}$ *admet un plus petit élément*. Connaissant $\mathbb{N}$ on peut démontrer la validité du principe de récurrence,

Théorème 1.1

Soit $\mathscr{P}(n)$ une propriété dépendant de $n \in \mathbb{N}$. Si les conditions suivantes sont vérifiées :
- $\mathscr{P}(0)$ est vraie ;
- $\forall n \in \mathbb{N}, \mathscr{P}(n) \Rightarrow \mathscr{P}(n+1)$;

alors la propriété $\mathscr{P}(n)$ est vraie pour tout $n \in \mathbb{N}$.

Remarque 1. $\mathscr{P}(0)$ est parfois appelée la base, et $\forall n \in \mathbb{N}, (\mathscr{P}(n) \Rightarrow \mathscr{P}(n+1))$ est appelé le pas de récurrence.

Remarque 2. La base peut être relative à un autre entier que 0.

1 Soit n de $\mathbb{N}^*$. On pose $S_n = \sum_{i=1}^{n} i$. Démontrer par récurrence que $S_n = \frac{n(n+1)}{2}$.

Remarque 3. Le principe de récurrence permet d'établir la validité d'une propriété $\mathscr{P}(n)$ qu'il faut déjà avoir identifiée ou conjecturée.

2 Existence et unicité de l'écriture d'un entier naturel en base b où b est élément de $\mathbb{N} - \{0, 1\}$.

Il existe une version forte du principe de récurrence qui permet de supposer dans le pas d'induction la validité de $\mathscr{P}$ à tous les rangs inférieurs. Ce principe s'énonce ainsi,

Théorème 1.2

Soit $\mathscr{P}(n)$ une propriété dépendant de $n \in \mathbb{N}$. Si les conditions suivantes sont vérifiées :
- $\mathscr{P}(0)$ est vraie ;
- $\forall n \in \mathbb{N}, (\mathscr{P}(0), \ldots, \mathscr{P}(n)) \Rightarrow \mathscr{P}(n+1)$;

alors la propriété $\mathscr{P}(n)$ est vraie pour tout $n \in \mathbb{N}$.

3 Montrer que tout entier supérieur ou égal à 2 possède un diviseur premier.

II Utilisation informatique de la récurrence

L'utilisation du raisonnement par récurrence peut s'illustrer dans l'analyse des algorithmes. Les algorithmes récursifs ou encore la présence de boucle de type *tant que* et *pour* sont particulièrement propices à des raisonnement par récurrence. Plus précisémment, la récurrence peut être utile pour :

- établir une preuve de validité (le programme calcule bien à chaque itération ce qu'il doit calculer)
- établir une preuve de convergence (l'algorithme se termine, c'est donc bien un algorithme...)
- évaluer la complexité temporelle d'un algorithme donné (ce point sera développé au chapitre 6).

II.1 Étude de l'algorithme d'Euclide

Considérons l'algorithme d'Euclide (≈ 300 av JC) qui étant donné deux entiers $a, b \in \mathbb{N}$, $a > b$, calcule le plus grand commun diviseur.

Algorithme 1

Algorithme d'Euclide
  fonction $pgcd(a,b)$
  while $b \neq 0$ **do**
    $t := b$
    $b := a \bmod b$
    $a := t$
  end while
  return a

4

a. Montrer que si q et r sont les uniques entiers tels que $a = bq + r$ (division euclidienne) alors $pgcd(a,b) = pgcd(b,r)$

b. Montrer que si $b = 0$ alors $pgcd(a,b) = a$.

c. En déduire que l'algorithme calcule bien le pgcd de a et b.

d. Donner un argument qui justifie la convergence de l'algorithme.

Remarque 4. Lorsqu'on démontre par "récurrence" la validité d'un algorithme on identifie une propriété appelée *invariant de boucle* $\mathscr{P}(n)$ dont on prouvera la véracité par récurrence et telle que, à la sortie de la boucle, la propriété implique le résultat voulu.

L'intérêt de la récurrence pour prouver la validité et la convergence d'un algorithme récursif est encore plus directe. Voici la version récursive de l'algorithme d'Euclide.

Algorithme 2

Algorithme d'Euclide récursif

fonction $pgcdR(a,b)$
if $b = 0$ **then**
    return a
else return $pgcdR(b, a \bmod b)$
end if

III L'induction

La preuve par récurrence s'appuie sur la propriété fondamentale de $\mathbb{N}$ qui assure que chaque ensemble non vide possède un plus petit élément. La notion d'induction va nous permettre de construire d'autres ensembles et d'établir des preuves par induction sur ces ensembles.

III.1 Définition d'un ensemble par induction

Définition 1.3. Soit E un ensemble. Une définition inductive d'une partie X de E revient à se donner :

a. Une base $\mathscr{B}$, c'est-à-dire un sous ensemble non vide de E.

b. Un ensemble de règles $\mathscr{R} : (R_1, \ldots, R_n)$ assurant la construction de X à partir d'éléments déjà construits.

On dit alors que X est défini par le shéma inductif $\langle \mathscr{B}, \mathscr{R} \rangle$.

Exemple 1. On considère le sous-ensemble M de $\mathbb{R}$ défini par la donnée d'un schéma $M = \langle \mathscr{B}, \mathscr{R} \rangle$ avec

a. $\mathscr{B} : 0 \in M$

b. $\mathscr{R}$:
- $R_1 : x \in M \Rightarrow x + 1 \in M$
- $R_2 : x \in M \Rightarrow x - 1 \in M$

Remarque 5. Le problème posé à ce stade est le suivant : l'ensemble issu de l'application du schéma $\langle \mathscr{B}, \mathscr{R} \rangle$ est-il le même que l'ensemble $\mathbb{Z}$ connu ordinairement par les mathématiciens ? Deux questions émergent alors : la complétude et la consistance du schéma $\langle \mathscr{B}, \mathscr{R} \rangle$.

Définition 1.4. Soit E un ensemble et $E' = \langle \mathscr{R}, \mathscr{B} \rangle$, un ensemble défini par induction. Alors
- on dit que le schéma d'induction $\langle \mathscr{B}, \mathscr{R} \rangle$ est consistant ssi $E' = \langle \mathscr{B}, \mathscr{R} \rangle \subset E$;
- on dit que le schéma d'induction $\langle \mathscr{B}, \mathscr{R} \rangle$ est complet ssi $E' = \langle \mathscr{B}, \mathscr{R} \rangle \supset E$.

Remarque 6. Si E' est consistant et complet par rapport à E, la définition 1.4 implique que $E' = E$.

5 Montrer que $\mathbb{Z}$ est bien l'ensemble construit par le schéma inductif de l'exemple 1.

Définition 1.5. Soit $x \in X = \langle \mathscr{B}, \mathscr{R} \rangle$, on appelle arbre de dérivation de x la donnée d'un élément de la base $x_0 \in \mathscr{B}$ et d'une suite $(R_{i_1}, \ldots, R_{i_m})$ de règles de $\mathscr{R}$ permettant d'obtenir x à partir de x_0 en appliquant $(R_{i_1}, \ldots, R_{i_m})$.

Remarque 7. Attention il n'y a pas unicité de l'arbre de dérivation ! Lorsque plusieurs arbres de dérivation permettent d'obtenir le même élément, on dit que le schéma est *ambiguë*.

III.2 Preuve par induction

Le résultat suivant généralise le principe de récurrence.

Théorème 1.6

Preuve par induction. Soit $X = \langle \mathscr{B}, \mathscr{R} \rangle$ un sous-ensemble de E défini par induction et $\mathscr{P}$ une propriété dépendant de $x \in E$. Si les conditions suivantes sont vérifiées :
- $\mathscr{P}(x)$ est vraie pour tout $x \in \mathscr{B}$;
- $\mathscr{P}$ est stable pour les règles R_i, c'est-à-dire $\mathscr{P}(x)$ vraie implique $\mathscr{P}(R_i(x))$ est encore vraie ;

Alors $\mathscr{P}(x)$ est vraie pour tout $x \in X$.

Exemple 2. On considère une "propriété", notée $\mathscr{P}(z)$, dépendant de l'entier relatif z. On a alors l'équivalence suivante :

$$(\forall z \in \mathbb{Z} \quad \mathscr{P}(z)) \Leftrightarrow ([\mathscr{P}(0)] \quad \text{et } [\forall z \in \mathbb{Z} \quad (\mathscr{P}(z) \Rightarrow \mathscr{P}(z+1) \text{ et } \mathscr{P}(z-1))])$$

Remarque 8.
- $\mathscr{P}(0)$ est appelée la base ;
- $\forall z \in \mathbb{Z} \quad (\mathscr{P}(z) \Rightarrow \mathscr{P}(z+1) \text{ et } \mathscr{P}(z-1))$ est appelé le pas d'induction ou la règle.

6 Démontrer par induction sur $\mathbb{Z}$ que $\forall (m,n) \in \mathbb{Z}^2 \quad a^{m+n} = a^m * a^n$

III.3 Définition inductive d'une fonction

Définition 1.7. Une fonction $f : E \to F$ est définie inductivement sur un ensemble $E = \langle \mathscr{B}, \mathscr{R} \rangle$ si

a. $\mathscr{B}$: $\forall x \in \mathscr{B}$, $f(x)$ est donnée ;

b. $\mathscr{R}$: la donnée de $f(x)$ permet de connaître $f(R_i(x))$ pour toute règle R_i de $\mathscr{R}$.

7 On suppose connus le produit et le quotient des réels. Définir inductivement la fonction $f : n \longmapsto a^n$ par un schéma $\langle \mathscr{B}, \mathscr{R} \rangle$, à partir de la définition inductive de $\mathbb{Z}$ (a désigne un réel non nul et n un élément quelconque de $\mathbb{Z}$).

Remarque 9. Lorsque l'ensemble $X = \langle \mathscr{B}, \mathscr{R} \rangle$ est ambiguë et qu'on définit par induction $f : X \to E$, il faut vérifier que $f(x)$ est indépendant du choix de l'arbre de dérivation de x.

Le but de ce chapitre est de donner une présentation élémentaire (naïve) de la théorie des ensembles, vu comme la structure la plus élémentaire possible : une collection d'objets.

I Ensembles

I.1 Définitions

Définition 2.1. Un ensemble est une collection d'objets. Les objets d'un ensemble sont appelés *éléments* de l'ensemble. Si x est un élément de l'ensemble E on note $x \in E$ qui se lit x appartient à E.

Exemple 3. $A = \{1,2,3\}$
$B = \{\text{Les cercles du plan de rayon } > 1\}$
$C = \{n \in \mathbb{N}, 7 \mid n \text{ et } n^2 > 1000\}$

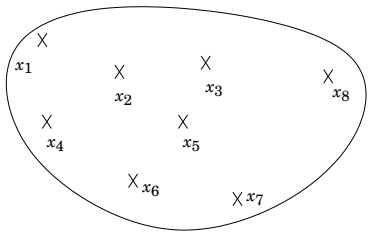


FIGURE 2.1 – Représentation schématique d'un ensemble

Définition 2.2. Un ensemble peut être défini par extension, c'est-à-dire on enumère les éléments constituant l'ensemble $E = \{x_1, x_2, \ldots\}$ ou bien par compréhension, c'est-à-dire on caractérise par une propriété les éléments constituant l'ensemble, $E = \{x | P(x)\}$.

Définition 2.3. Soient E et F deux ensembles.

a. On dit que E est inclus dans F, et on note $E \subset F$, si et seulement si $\forall x, x \in E \Rightarrow x \in F$.

b. On dit que E et F sont égaux, et on note $E = F$, si et seulement si $\forall x, x \in E \Leftrightarrow x \in F$.

c. On appelle ensemble vide, et on note $\emptyset$, l'ensemble qui ne contient aucun élément. Ce qui se traduit par $E = \emptyset \Leftrightarrow \forall x, x \notin E$.

Proposition 2.4

a. $\forall E, \emptyset \subset E$.

b. E et F sont égaux si et seulement si $E \subset F$ et $F \subset E$. (Principe de double inclusion).

Définition 2.5. Soit E un ensemble et $(A_i)_{i \in I}$ une famille de parties (i.e. de sous-ensembles) de E est une partion de E lorsque les conditions suivantes sont vérifiées :

- $\forall i \in I, A_i \neq \emptyset$
- $\forall (i,j) \in I^2, (i \neq j \Rightarrow A_i \cap A_j = \emptyset)$
- $\bigcup_{i \in I} A_i = E$.

I.2 Opérations sur les ensembles

Définition 2.6. Soient E et F deux ensembles,

a. La réunion de E et F, notée $E \cup F$ est l'ensemble des éléments de E et de F. C'est à dire $(x \in E \cup F) \Leftrightarrow (x \in E)$ ou $(x \in F)$.

b. L'intersection de E et F, notée $E \cap F$ est l'ensemble des éléments communs à E et F. C'est à dire $(x \in E \cap F) \Leftrightarrow (x \in E)$ et $(x \in F)$

c. E privé de F, noté $E \setminus F$ est l'ensemble des éléments de E qui ne sont pas dans F. C'est à dire $(x \in E \setminus F) \Leftrightarrow (x \in E)$ et $(x \notin F)$.

d. Si $F \subset E$, le *complémentaire de F dans E*, noté $C_E F$, ou encore F^c ou $\overline{F}$ si il n'y a pas d'ambiguïté sur E, désigne $E \setminus F$.

e. Le *produit cartésien* $E \times F$ est l'ensemble constitué des couples (e, f) avec $e \in E$ et $f \in F$.

Remarque 10. Le "ou" est toujours inclusif.

Exemple 4. **a.** $\{1,3,5\} \cup \{2,3,4\} = \{1,2,3,4,5\}$

b. $\{1,3,5\} \cap \{2,3,4\} = \{3\}$

c. $\{1,3,5\} \setminus \{2,3,4\} = \{1,5\}$

d. $\{1,2\} \subset \{1,2,3,4\}$, $\overline{\{1,2\}} = \{3,4\}$.

e. $\{1,3,5\} \times \{2,3,4\} = \{(1,2);(1,3);(1,4);(3,2);(3,3);(3,4);(5,2);(5,3),(5,4)\}$

Remarque 11. Attention en général $E \times F \neq F \times E$!

Remarque 12. Les quatre premières définitions de la définition 2.6 s'expliquent très bien avec un dessin (diagramme de Venn) !

Définition 2.7. Si E est fini, on appelle cardinal de E, et on note $Card(E)$ ou $\sharp E$ ou $|E|$ le nombre d'éléments de E.

1 Si E et F sont finis, montrer que $|E \times F| = |E| \times |F|$.

2 Soient E et F deux ensembles, avec $F \subset E$. Montrer que $\overline{\overline{F}} = F$.

I.3 Ensemble des parties d'un ensemble

Définition 2.8. Une *partie* (ou sous ensemble) F de E est un ensemble F tel que $F \subset E$. On note $\mathscr{P}(E)$ l'ensemble constitué des parties de E.

Remarque 13. On notera que $\mathscr{P}(E)$ peut être considéré comme un ensemble (c'est l'ensemble des parties de E). Ses éléments sont des ensembles ! En particulier $\mathscr{P}(E)$ contient deux éléments remarquables : $\emptyset$ et E.

3 Déterminer $\mathscr{P}(E)$ pour $E = \{\diamond, \triangle\}$.

Proposition 2.9

Si $|E| = n$, $|\mathscr{P}(E)| = 2^n$.

II Applications

II.1 Définitions

Définition 2.10. On appelle application f de E sur F, la donnée d'un triplet (E,F,Γ) où $\Gamma \subset E \times F$ tel que

$$\forall x \in E, \exists! y \in F, (x,y) \in \Gamma.$$

Lorsque $(x,y) \in \Gamma$ on note $y = f(x)$ et on dit que y est *l'image* de x par f, et que x est un *antécédent* de y par f. Les ensembles E et F sont respectivement appelés ensemble de départ et ensemble d'arrivée. Le sous-ensemble Γ de $E \times F$ est appelé *graphe* de f.

Remarque 14. On retiendra qu'une application associe à tout élément de l'ensemble de départ un unique élément dans l'ensemble d'arrivée et qu'elle est caractérisée par la donnée d'un graphe. De plus on prendra soin de bien distinguer les notions d'image et d'antécédent.

Remarque 15. On confondra très souvent les termes fonctions et applications. [1]

Remarque 16. On notera F^E l'ensemble des applications de E dans F. Par exemple $\mathbb{R}^{\mathbb{N}}$ représente l'ensemble des applications de $\mathbb{N}$ dans $\mathbb{R}$, c'est à dire l'ensemble des suites numériques.

Définition 2.11. Soit $f : E \to F$ une application et $G \subset E$ un sous-ensemble de E. On peut définir naturellement une application $g : G \to F$ telle que $\forall x \in G$, $g(x) = f(x)$. La fonction g est appelée restriction de f à G. De même on qualifiera f de prolongement de g.

Définition 2.12. Soient $f : E \to F$ et $g : F \to G$ deux applications, on appelle application composée de f et de g l'application notée $g \circ f : E \to G$ telle que $x \in E$ a pour image $g(f(x))$.

Remarque 17. Attention en général $f \circ g \neq g \circ f$.

4 Proposer un (ou des) contre-exemple à la remarque précédente.

II.2 Injectivité, surjectivité, bijectivité

Définition 2.13. Soit f une application de E dans F. On dit que f est
- *injective* si tout élément de F admet au plus un antécédent.
- *surjective* si tout élément de F admet au moins un antécédent.
- *bijective* tout élément de F admet exactement un antécédent.

Cette définition se traduit de la façon suivante à l'aide de quantificateurs :
- f est injective $\Leftrightarrow \forall x \in E, \forall y \in E, f(x) = f(y) \Rightarrow x = y$.
- f est surjective $\Leftrightarrow \forall y \in F, \exists x \in E, f(x) = y$.
- f est bijective $\Leftrightarrow \forall y \in F, \exists! x \in E, f(x) = y$.

Remarque 18. On notera qu'une application f est bijective si et seulement si f est injective et surjective.

5 Soit $\mathbb{R}[X]$ l'ensemble des polynômes à coefficients réels. Soit l'application $\phi : \left\{ \begin{array}{ccc} \mathbb{R}[X] & \to & \mathbb{R}[X] \\ P & \mapsto & P' \end{array} \right.$.
Montrer que ϕ est surjective mais n'est pas injective.

1. On appelle fonction f de E sur F, la donnée d'un triplet (E,F,Γ) et d'un ensemble $D \subset E$ où $\Gamma \subset D \times F$ tel que $\forall x \in D, \exists! y \in F, (x,y) \in \Gamma$. Le sous-ensemble D est appelé domaine de définition de la fonction.

II.3 Images directes, images réciproques

Définition 2.14. Soit f une application de E dans F.
- Si $A \subset E$, l'image *directe* de A par f, notée $f(A)$, est l'ensemble

$$f(A) = \{y \in F, \exists x \in A, y = f(x)\}.$$

- Si $B \subset F$, l'image *réciproque* de B par f, notée $f^{-1}(B)$ est l'ensemble

$$f^{-1}(B) = \{x \in E, f(x) \in B\}.$$

Remarque 19. En général f^{-1} ne désigne pas une application. Par contre lorsque $f : E \to F$ est bijective on peut définir une application réciproque que l'on note f^{-1}.

Proposition 2.15

Soit $f : E \to F$ une application. On a alors,
f bijective $\Leftrightarrow \exists\, g : F \to E$ telle que $f \circ g = Id_F$ et $g \circ f = Id_E$. Dans ce cas on a $g = f^{-1}$.

III Cardinaux

Le cardinal d'un ensemble fini est le nombre d'éléments de l'ensemble. Dans ce paragraphe on donne les premiers éléments pour étudier les cardinaux d'ensembles infinis.

Définition 2.16. On note $\aleph_0$ ("aleph zero") le cardinal de $\mathbb{N}$.

Une façon naturelle de comparer deux ensembles est de mettre en relation 1-1 les éléments des deux ensembles respectifs.

Définition 2.17. Deux ensembles E et F sont dits équipotents (ou de même cardinal) s'il existe une bijection $\phi : E \to F$, on note alors $|E| = |F|$.

Tous les ensembles infinis ne sont pas équipotents (il existe des infinis de tailles différentes !).

Théorème 2.18 (de Cantor)

Pour tout ensemble E, $|\mathscr{P}(E)| > |E|$.

Définition 2.19. Un ensemble infini est dit dénombrable s'il est de cardinal $\aleph_0$.

6 Montrer que $|\mathbb{N}^*| = \aleph_0$.

Théorème 2.20

a. $\mathbb{Q}$ est dénombrable ($|\mathbb{Q}| = \aleph_0$)

b. $\mathbb{R}$ est non dénombrable (en particulier $|\mathbb{R}| = |\mathscr{P}(\mathbb{N})| = 2^{\aleph_0}$)

I Relations

I.1 Généralités

Définition 3.1. Une relation (binaire) sur les ensembles E et F (de E vers F) est la donnée d'un triplet (E,F,Γ) où $\Gamma \subset E \times F$. E est appelé ensemble de départ et F ensemble d'arrivée, la partie $\Gamma \subset E \times F$ est appelée graphe de la relation. Si $(x,y) \in \Gamma$ on dit que x est en relation avec y et on note $x\mathscr{R}y$.

1 Quelles sont les différences entre les définitions 2.10 et 3.1 ?

Exemple 5. Toute application $f : E \to F$ définit une relation de E vers F.

Définition 3.2. Lorsqu'on étudie une relation de E vers E (c'est à dire lorsque $F = E$), on parlera de relation sur E ou dans E. Une telle relation est dite interne.

Exemple 6. Sur $\mathbb{N}^*$ on définit la relation suivante : $n\mathscr{R}m \Leftrightarrow n|m$.

Exemple 7. Dans $\mathscr{V}$ l'ensemble des vecteurs du plan, on définit la relation suivante $\vec{v}\mathscr{R}\vec{w} \Leftrightarrow \vec{v}.\vec{w} = 0$.

Remarque 20. On peut également définir des relations n-aire sur des ensembles $E_1,\dots,E_n$. Dans ce cas la relation est donnée par une sous-partie $\Gamma \subset E_1 \times \cdots \times E_n$.

Définition 3.3. On condisidère deux relations $\mathscr{R}$ de E dans F et $\mathscr{S}$ de F dans G. Alors la composée $\mathscr{S} \circ \mathscr{R}$ est la relation de E dans G telle que $\forall (x,z) \in E \times G$,

$$x(\mathscr{S} \circ \mathscr{R})z \Leftrightarrow \exists y \in F, x\mathscr{R}y \text{ et } y\mathscr{S}z \tag{3.1}$$

Définition 3.4. Soit $\mathscr{R}$ une relation de E dans F. On définit la relation inverse $\mathscr{R}^{-1}$ de F dans E de la façon suivante, $\forall (y,x) \in F \times E$, $y\mathscr{R}^{-1}x \Leftrightarrow x\mathscr{R}y$.

I.2 Représentation des relations binaires

Lorsque les ensembles finis $E = \{e_1,\dots e_n\}$ et $F = \{f_1,\dots,f_m\}$, on peut représenter des relations binaires en :

a. Représentant $\Gamma \subset E \times F$ le graphe de la relation ;

b. Donnant un graphe orienté : les éléments de E et de F sont les sommets du graphe liés par des flèches de sorte que si $e_i\mathscr{R}f_j$ alors une flèche doit lier e_i à f_j.

c. Construisant la matrice booléenne de la relation.

Définition 3.5. Soit $\mathscr{R}$ une relation de $E = \{e_1,\dots,e_m\}$ dans $F = \{f_1,\dots,f_n\}$. La matrice booléenne de $\mathscr{R}$, notée $M_{\mathscr{R}}$ est la matrice de taille $m \times n$ telle que $m_{ij} = 1 \Leftrightarrow e_j\mathscr{R}f_i$ et $m_{ij} = 0$ sinon.

Remarque 21. On peut trouver une définition alternative dans les ouvrages ($m_{ij} = 1 \Leftrightarrow e_i\mathscr{R}f_j$) qui peut sembler plus naturelle mais celle-ci inverse l'ordre usuel dans la détermination matricielle d'une composée.

2 Écrire la matrice boolénne de la relation $n|m$ sur $E = \{1,2,3,4\}$.

Théorème 3.6

Soient E, F et G des ensembles finis, $\mathscr{R}$ une relation de E dans F et $\mathscr{S}$ une relation de F dans G de matrices boolénnes associées $M_{\mathscr{R}}$ et $M_{\mathscr{S}}$. Alors la matrice de la relation $\mathscr{S} \circ \mathscr{R}$ est donnée par

$$M_{\mathscr{S}\circ\mathscr{R}} = M_{\mathscr{S}} M_{\mathscr{R}} \tag{3.2}$$

(dans le calcul du produit matriciel, les sommes et produits sont booléens).

Théorème 3.7

Soit E et F deux ensembles finis et $\mathscr{R}$ une relation de E dans F de matrice boolénne associée $M_{\mathscr{R}}$. Alors la matice booléenne de la relation $\mathscr{R}^{-1}$ est

$$M_{\mathscr{R}^{-1}} = {}^{t}M_{\mathscr{R}} \tag{3.3}$$

II Relations d'équivalence

Définition 3.8. Une relation d'équivalence $\mathscr{R}$ sur E est une relation sur E telle que :
- $\mathscr{R}$ est réflexive, i.e. $\forall x \in E, x\mathscr{R}x$.
- $\mathscr{R}$ est symétrique, i.e. $\forall (x,y) \in E^2, x\mathscr{R}y \Rightarrow y\mathscr{R}x$.
- $\mathscr{R}$ est transitive, i.e. $(x,y,z) \in E^3, (x\mathscr{R}y)$ et $(y\mathscr{R}z) \Rightarrow x\mathscr{R}z$.

Exemple 8. **a**. La relation égalité sur $\mathbb{R}$ est une relation d'équivalence.

b. La relation de parallélisme est une relation d'équivalence sur l'ensemble des droites du plan.

c. La relation «modulo 2π» pour les mesures d'angles est une relation d'équivalence.

d. Sur $\mathbb{Z}$ on considère la relation suivante $a\mathscr{R}b \Leftrightarrow 5|(a-b)$. C'est une relation d'équivalence.

e. Sur $\mathbb{N}^2$, on considère la relation $(n,m)\mathscr{R}(n',m') \Leftrightarrow n+m' = n'+m$ est une relation d'équivalence.

3 Les relations des exemples 6 et 7 sont-elles des relations d'équivalence ?

4 Soit E un ensemble fini et $\mathscr{R}$ une relation d'équivalence sur E. Montrer que $M_{\mathscr{R}}$ est symétrique et de trace égale à $|E|$.

Définition 3.9. Soit E un ensemble muni d'une relation d'équivalence $\mathscr{R}$. On appelle classe d'équivalence de x le sous-ensemble de E suivant :

$$\overline{x} = \{y \in E | x\mathscr{R}y\}$$

Exemple 9. Pour la relation parallélisme dans l'ensemble des droites du plan muni d'un repère $(O, \vec{i}, \vec{j})$, si on note Δ la droite correspondant à (Ox) alors

$\overline{\Delta}$ = {l'ensemble des droites d'équations $y = k, k \in \mathbb{R}$}.

Théorème 3.10

Soit $\mathscr{R}$ une relation d'équivalence sur E. Soit x et y deux éléments de E et $\overline{x}$ et $\overline{y}$ les classes d'équivalence de x et y. Alors,

a. $\overline{x} \neq \emptyset$.

b. $\overline{x} \cap \overline{y} \neq \emptyset \Rightarrow \overline{x} = \overline{y}$

c. Les classes d'équivalence de E forment une partition de E,

$$E = \sqcup_{x \in E} \overline{x} \text{ (union disjointe)}$$

C'est à dire l'ensemble des classes d'équivalence constitue un ensemble de parties non vides deux à deux disjointes dont la réunion est l'ensemble tout entier.

Définition 3.11. L'ensemble quotient de E par la relation d'équivalence $\mathscr{R}$ est l'ensemble noté $E/\mathscr{R}$ des classes d'équivalence de E pour la relation $\mathscr{R}$:

$$E/\mathscr{R} = \{\overline{x} | x \in E\}.$$

Exemple 10. Notons $[2\pi]$ la relation d'équivalence modulo 2π sur $\mathbb{R}$, alors $\mathbb{R}/[2\pi] = [0, 2\pi[$.

Exemple 11. Soit $\mathscr{R}$ la relation d'équivalence sur $\mathbb{Z}$ définie par $x \mathscr{R} y \Leftrightarrow 5|(x-y)$, alors $\mathbb{Z}/\mathscr{R} = \{\overline{0}, \overline{1}, \overline{2}, \overline{3}, \overline{4}\}$

5 Calculer les ensembles quotients pour les relations d'équivalence vues dans l'exemple 8.

6 Soit $n \in \mathbb{N}$ on considère sur $\mathbb{Z}$ la relation $\mathscr{R}$ définie par $x \mathscr{R} y \Leftrightarrow n|(x-y)$

a. Montrer que $\mathscr{R}$ est une relation d'équivalence.

b. Pour tout $x \in \mathbb{Z}$ déterminer la classe $\overline{x}$ de x pour cette relation et vérifier que $\overline{x} \cap \overline{y} = \emptyset \Leftrightarrow \overline{x} \neq \overline{y}$

c. On note $\mathbb{Z}/n\mathbb{Z}$ l'ensemble quotient pour cette relation. Quel est son cardinal ?

d. On note $\overline{+}$ l'opération dans $\mathbb{Z}/n\mathbb{Z}$, $\overline{x} \overline{+} \overline{y} = \overline{x+y}$. Vérifier que $\overline{+}$ est bien définie.

e. Comment peut-on définir $\overline{\times}$?

Remarque 22. La notion de relation d'équivalence permet de penser un ensemble initial E sous forme de classes (les éléments de l'ensembles quotients) qui partitionnent l'ensemble E. En fait la donnée d'une partition de E ou d'une relation d'équivalence sur E est équivalente comme le montre le résultat suivant.

Théorème 3.12

Soit E un ensemble. Notons $E(\mathscr{R})$ l'ensemble de toutes les relations d'équivalence sur E et $E(\mathscr{P})$ l'ensemble de toutes les partitions possibles de E. On considère $F : E(\mathscr{R}) \to E(\mathscr{P})$ l'application qui, à une relation $\mathscr{S}$, associe la partition $\pi_{\mathscr{S}}$ de E définie par les classes $\pi_{\mathscr{S}} = (\overline{x})_{x \in E}$, i.e. $F(\mathscr{S}) = \pi_{\mathscr{S}}$. Alors F est une bijection.

III Relations d'ordre

Définition 3.13. Une relation d'ordre sur E est une relation $\mathscr{R}$ telle que :
- $\mathscr{R}$ est réflexive, i.e. $\forall x \in E, x\mathscr{R}x$
- $\mathscr{R}$ est antisymétrique, i.e. $\forall (x,y) \in E^2, (x\mathscr{R}y)$ et $(y\mathscr{R}x) \Rightarrow x = y$.
- $\mathscr{R}$ est transitive, i.e. $(x,y,z) \in E^3, (x\mathscr{R}y)$ et $(y\mathscr{R}z) \Rightarrow x\mathscr{R}z$.

Un ensemble muni d'une relation d'ordre est un ensemble ordonné.

Exemple 12. $\geq$ définit une relation d'ordre sur $\mathbb{R}$,
$\subset$ définit une relation d'ordre sur $\mathscr{P}(\mathbb{N})$ (et plus généralement sur $\mathscr{P}(E)$).
La relation divise, définie par $a\mathscr{R}b \Leftrightarrow a|b$ est une relation d'ordre sur $\mathbb{N}^*$.

Définition 3.14. On dit qu'une relation d'ordre $\mathscr{R}$ est une relation d'ordre total sur E lorsque deux éléments quelconques de E peuvent être comparés, c'est à dire que

$$\forall (x,y) \in E^2 \Leftrightarrow x\mathscr{R}y \text{ ou } y\mathscr{R}x.$$

On dit alors que E est totalement ordonné.
Si la relation d'ordre n'est pas d'ordre total on parle de relation d'ordre partiel et d'ensemble partiellement ordonné.

Exemple 13. $(\mathbb{R}, \geq)$ est un ensemble totalement ordonné.
$(\mathscr{P}(E), \subset)$ est un ensemble partiellement ordonné.

7 Trouver une relation d'ordre sur $\mathbb{R}^2$. Cette relation est-elle d'ordre total ?

8 Soit $\mathscr{R}$ la relation sur $\mathbb{N}^*$ définie par $n\mathscr{R}m \Leftrightarrow n \mid m$. Montrer que $(\mathbb{N}^*, \mathscr{R})$ est un ensemble partiellement ordonné.

Dans ce chapitre nous définissons la structure de Treillis sur un ensemble. Il s'agit d'une structure d'ordre possédant la propriété suivante : tout sous-ensemble fini possède un plus petit majorant (borne sup) et un plus grand minorant (borne inf). C'est une structure fondamentale en informatique qui se situe à l'intersection de l'algèbre, de la logique ou encore de la combinatoire. Plusieurs domaines actifs de l'informatique font appel à cette notion : traitement des bases de données et représentation des connaissances, apprentissage machine (classification conceptuelle), systèmes concurrents.

I Ensemble ordonné

Rappelons la définition d'une relation d'ordre et d'ensemble ordonné vue au chapitre précédent.

Définition 4.1. Une relation d'ordre sur E est une relation $\mathscr{R}$ telle que :
- $\mathscr{R}$ est réflexive, i.e. $\forall x \in E, x\mathscr{R}x$
- $\mathscr{R}$ est antisymétrique, i.e. $\forall (x,y) \in E^2, (x\mathscr{R}y)$ et $(y\mathscr{R}x) \Rightarrow x = y$.
- $\mathscr{R}$ est transitive, i.e. $(x,y,z) \in E^3, (x\mathscr{R}y)$ et $(y\mathscr{R}z) \Rightarrow x\mathscr{R}z$.

Un ensemble muni d'une relation d'ordre est un ensemble ordonné.

Définition 4.2. On dit qu'une relation d'ordre $\mathscr{R}$ est une relation d'ordre total sur E lorsque deux éléments quelconques de E peuvent être comparés, c'est à dire que

$$\forall (x,y) \in E^2 \Leftrightarrow x\mathscr{R}y \text{ ou } y\mathscr{R}x.$$

On dit alors que E est totalement ordonné.

Si la relation d'ordre n'est pas d'ordre total on parle de relation d'ordre partiel et d'ensemble partiellement ordonné.

Exemple 14. Ensembles ordonnés : $(\mathbb{N}, \leq)$, $(\mathscr{P}(E), \subset)$, $(\mathbb{N}^*, |)$.

Exemple 15. On considère un alphabet $A = \{a_1, \ldots, a_n\}$ muni d'une relation d'ordre (total) définie par $a_i \leq a_{i+1}$. Alors on définit l'ordre lexicographique sur l'ensemble des mots (non vides) engendrés par A de la manière suivante.
- Si $\alpha_1 \leq \beta_1$ alors $\alpha_1 \ldots \alpha_n \preceq \beta_1 \ldots \beta_m$.
- Si $\alpha_i = \beta_i$ pour $1 \leq i \leq k$ et $\alpha_{k+1} \leq \beta_{k+1}$ alors $\alpha_1 \ldots \alpha_n \preceq \beta_1 \ldots \beta_m$.
- Si $\alpha_1 \ldots \alpha_n = \beta_1 \ldots \beta_n$ alors $\alpha_1 \ldots \alpha_n \preceq \beta_1 \ldots \beta_m$ ($\alpha_1 \ldots \alpha_n$ est un préfixe de $\beta_1 \ldots \beta_m$).

La donnée d'une relation d'ordre permet de parler de majorant et minorant d'une partie non-vide.

Définition 4.3. Soit $(E, \preceq)$ un ensemble ordonné et $A \subset E$ une partie non vide de E. Alors
- $M \in E$ est un majorant de A si et seulement si $\forall x \in A, x \preceq M$.
- $m \in E$ est un minorant de A si et seulement si $\forall x \in A, x \preceq m$.

1 Donner un majorant et un minorant de toute partie $A \subset \mathscr{P}(E)$.

Définition 4.4. Soit A une partie non vide de $(E, \preceq)$.
- On appelle borne sup de A, l'élément Φ de E (s'il existe !) qui soit le plus petit des majorants de A.

- On appelle borne inf de A, l'élément ϕ de E (s'il existe !) qui soit le plus grand des minorants de A.

Proposition 4.5

Soit A un ensemble non vide de E. La borne sup (resp. inf) de A est unique.

Remarque 23. La borne sup (resp. la borne inf) n'est pas nécessairement le plus grand (resp. le plus petit) élément de A.

2 On considère $A = \{x \in \mathbb{Q}_+, x^2 < 2\}$. Montrer que A n'a pas de plus grand élément. Quelle est sa borne sup ?

Définition 4.6. Un ensemble est dit bien ordonné si et seulement si toute partie non vide admet un plus petit élément.

Exemple 16. $(\mathbb{N}, \leq)$ est bien ordonné.

Définition 4.7. Un ensemble ordonné fini se représente à l'aide d'un diagramme de Hasse.

3 Donner le diagramme de Hasse de $(\mathscr{P}(E), \subset)$ pour $E = \{1,2,3\}$.

Proposition 4.8 (Ordre sur le produit cartésien)

Soient $(E_1, \leq_1)$ et $(E_2, \leq_2)$ deux ensembles ordonnées.

a. On définit sur $E_1 \times E_2$ l'ordre produit $\preceq$ de la façon suivante :

$$(x_1, x_2) \preceq (y_1, y_2) \Leftrightarrow x_1 \leq_1 y_1 \text{ et } x_2 \leq_2 y_2 \tag{4.1}$$

b. On définit sur $E_1 \times E_2$ l'ordre lexicographique $\preceq_{lex}$ de la façon suivante :

$$(x_1, x_2) \preceq_{lex} (y_1, y_2) \Leftrightarrow (x_1 \neq y_1 \text{ et } x_1 \leq_1 y_1) \text{ ou } (x_1 = y_1 \text{ et } x_2 \leq_2 y_2) \tag{4.2}$$

II Treillis (définition par relation d'ordre)

Définition 4.9. Un ensemble ordonné $(E, \leq)$ est un treillis si et seulement si toute paire d'éléments de E, $A = \{a, b\}$ possède une borne supérieur et une borne inférieure.

4 Montrer qu'un ensemble totalement ordonné est un treillis.

5 Proposer un ensemble fini partiellement ordonné qui n'est pas un treillis.

Notation 1. La borne supérieure (resp. inférieure) d'une paire d'éléments $A = \{a, b\}$ sera, notée si elle existe, $a \vee b$ (resp. $a \wedge b$).

Théorème 4.10

Dans un treillis toute partie finie non vide possède une borne supérieure et une borne inférieure.

Exemple 17. Exemples de treillis classiques.

a. $(\mathscr{P}(E), \subset)$, de plus pour toute paire $\{A, B\}$ on a $A \vee B = A \cup B$ et $A \wedge B = A \cap B$.

b. $(\mathbb{N}^*, |)$, de plus pour toute paire $\{a, b\}$ on a $a \vee b = ppcm(a, b)$ et $a \wedge b = pgcd(a, b)$.

c. Si $(E, \leq)$ est un treillis alors $(E^n, \leq)$ est un treillis pour l'ordre produit.

d. Cas particulier : $E = \mathscr{B} = \{0, 1\}$ muni de la relation d'ordre $\leq$. Alors $(\mathscr{B}, \leq)$ est un treillis et $(\mathscr{B}^n, \leq)$ également pour la relation d'ordre produit.

6 Dans le cas de $(\mathscr{B}^n, \leq)$, comment définit-on $a \wedge b$ et $a \vee b$?

Définition 4.11. Un treillis $(E, \leq)$ est dit complet si et seulement si tout sous-ensemble (fini ou non) non vide de E admet une borne supérieure et une borne inférieure.

7 Tout treillis fini est complet.

Proposition 4.12

Soit $(E, \leq)$ un treillis complet. Alors E possède un plus petit élément, noté 0, et un plus grand élément, noté 1.

Définition 4.13. Dans un treills $(E, \leq)$ possédant un plus petit élément 0 et un plus grand élément 1, on appelle le complément de $x \in E$, tout élément $x' \in E$ tel que

$$x \wedge x' = 0 \text{ et } x \vee x' = 1 \tag{4.3}$$

On appelle treillis complémenté, un treillis possédant un plus petit et un plus grand élément tel que tout élément admet au moins un complément.

Exemple 18. $(\mathscr{P}(E), \subset)$ est un treillis complet et complémenté.

Définition 4.14. Soit $(E, \leq)$ un treillis et F une partie de E, alors F est dit sous-treillis de E si et seulement si $\forall x, y \in F$, $x \vee y \in F$ et $x \wedge y \in F$.

III Treillis (définition algébrique)

Nous venons de définir la notion de treillis à l'aide de la notion d'ensemble ordonné. On peut également définir la même structure par une définition purement algébrique.

Définition 4.15. Un treillis algébrique est un ensemble E muni de deux lois internes (c'est-à-dire deux applications $E \times E \to E$) notées $\vee$ et $\wedge$ telles que ces lois soient

a. associatives $\forall x, y, z \in E$, $x \vee (y \vee z) = x \vee (y \vee z)$ et $(x \wedge y) \wedge z = x \wedge (y \wedge z)$.

b. commutative $\forall x, y \in E$, $x \vee y = y \vee x$ et $x \wedge y = y \wedge x$.

c. idempotentes $\forall x \in E,\ x \vee x = x$ et $x \wedge x = x$.

d. absorbantes $\forall x, y \in E,\ x \vee (x \wedge y) = x \wedge (x \vee y) = x$.

Les deux notions sont équivalentes.

Proposition 4.16

Un treillis algébrique est un treillis pour la relation d'ordre $\preceq$ définie par $a \preceq b$ si et seulement si $a \wedge b = a$ (ou de manière équivalente $a \vee b = b$). Réciproquement un treillis est un treills algébrique pour les opérations $a \vee b = \sup(a, b)$ at $a \wedge b = \inf(a, b)$.

Remarque 24 (Principe de dualité)**.** Dans un treills toute propriété $\mathscr{P}$ conséquence logique des axiomes et du choix de la relation d'ordre ($x \preceq y \Leftrightarrow x \vee y = y$ ou $x \leq y \Leftrightarrow x \wedge y = x$) reste vraie lorsqu'on permute les symboles $\wedge$ et $\vee$ d'une part et $\preceq$ et $\succeq$, d'autre part. La nouvelle propriétée obtenue s'appelle ainsi la duale de $\mathscr{P}$.

8 Montrer que $(\mathscr{P}(E), \subset)$ est un treillis algébrique *distributif* (distributivité de $\cap$ par rapport à $\cup$ et réciproquement).

Proposition 4.17

Dans un treillis, si une loi est distributive par rapport à l'autre alors la seconde est également distributive par rapport à la première.

Remarque 25. Tous les treillis ne sont pas distributifs, mais on peut montrer qu'on a toujours $(x \wedge y) \vee (x \wedge z) \preceq x \vee (y \wedge z)$.

L'algèbre de Boole [1] est une algèbre construite sur des ensembles binaires (0, 1) particulièrement efficace pour modéliser les circuits logique (VRAI/FAUX) dont les applications vont de l'informatique à la conception de circuit électroniques en passant par les circuits de commutation téléphoniques. On peut présenter cette algèbre sous forme axiomatique avec ses règles et ses axiomes ou bien faire émerger cette structure à partir de celle de treillis.

Une fois la notion d'algèbre de Boole introduite, on s'intéresse à la notion de fonctions booléennes qui a l'état d'entrée d'un circuit logique associe une valeur de vérité en sortie. Ces fonctions sont aussi largement utilisée en cryptographie.

I Définitions - Propriétés premières

I.1 Définitions

Définition 5.1. On appelle algèbre de Boole tout treillis distributif et complémenté. Le complément de x sera noté x'.

Notation 2. Vues en termes algébriques, et non plus seulement sous l'aspect d'ensemble ordonné, les lois internes associées sont en général notées $\wedge$ et $\vee$. Pour la suite nous les noterons respectivement . et +.

Une définition directe (algébrique) équivalente est la suivante :

Définition 5.2. On appelle algèbre de Boole $(B,+,.)$ tout ensemble B muni de deux lois internes, notées + et ., vérifiant les propriétés suivantes :

a. $\forall x \in B, \exists x' \in B,\ x+x' = 1$ et $x.x' = 0$, où 1 et 0 sont des éléments distincts de B vérifiant : $\forall x \in B, 0.x = 0$ et $0+x = x; 1.x = x$ et $1+x = 1$

b. *Commutativité* : $\forall (x,y) \in B^2,\ x+y = y+x$ et $x.y = y.x$

c. *Associativité* : $\forall (x,y,z) \in B^3,\ x+(y+z) = (x+y)+z$ et $x.(y.z) = (x.y).z$

d. *Distributivité* [2] : $\forall (x,y,z) \in B^3,\ x.(y+z) = x.y + x.z$ et $x+(y.z) = (x+y).(x+z)$

e. *Idempotence* : $\forall x \in B,\ x+x = x$ et $x.x = x$

f. *Absorption* : $\forall (x,y) \in B^2,\ x.(x+y) = x$ et $x+x.y = x$

Remarque 26. Le jeu d'axiomes proposé n'est pas minimal.

Exemple 19.

a. $(\mathscr{P}(E), \cup, \cap)$, où E est un ensemble, est une algèbre de Boole ; historiquement, cet exemple fut le modèle menant à la structure abstraite.

b. $(\mathscr{B}, +, \times)$ constitue une algèbre de Boole,
sachant que $\mathscr{B} = \{0,1\}$, avec + et $\times$ désignant le *ou*, le *et* ordinaires.

c. $(\mathscr{B}^n, +, \times)$,avec n dans $\mathbb{N}^*$, muni des lois produits ordinaires (composantes à composantes) constitue une algèbre de Boole très importante car elle joue le rôle de modèle pour toutes les algèbres de Boole finies qui lui sont isomorphes, grâce à un bon choix de n (voir théorème de structure).

1. Georges Boole (1815-1864), logicien, mathématicien et philosophe anglais
2. L'une est ordinaire, l'autre moins. On pourra penser à l'opération équivalente sur les ensembles.

I.2 Propriétés premières

Proposition 5.3

Dans une algèbre de Boole $(B,+,.)$, pour tout (x,y) de B^2 on a :

a. $(x')' = x$;

b. $(x+y)' = x'.y'$;

c. $(x.y)' = x'+y'$

Remarque 27. Les deux dernières propriétés sont fréquemment connues sous le nom de lois de Morgan.

Proposition 5.4

Une algèbre de Boole $(B,+,.)$ est un ensemble ordonné par :

$$x \leq y \Leftrightarrow x.y = x \quad \text{ou} \quad x \leq y \Leftrightarrow x+y = y$$

qui constituent deux définitions équivalentes d'une même relation d'ordre.

Proposition 5.5

Sous les notations antérieures, l'ordre est compatible avec l'addition et le produit (à droite et à gauche), c'est-à-dire :

a. $\forall (x,y,z) \in B^3 (x \leq y \Rightarrow x+z \leq y+z \text{ et } z+x \leq z+y)$;

b. $\forall (x,y,z) \in B^3 (x \leq y \Rightarrow x.z \leq y.z \text{ et } z.x \leq z.y)$

Proposition 5.6

Dans une algèbre de Boole $(B,+,.)$, pour tout (x,y) de B^2 on a :

a. $x \leq y \Leftrightarrow x' \geq y'$;

b. $x \leq y \Leftrightarrow x.y' = 0$;

c. $x \leq y \Leftrightarrow x'+y = 1$.

I.3 Caractérisation des algèbres de Boole finies

Le résultat qui suit est qualitativement très important. Il établit que, structurellement, toutes les algèbres de Boole sont les "mêmes" à isomorphisme près, qu'un modèle unique.

Théorème 5.7

Toute algèbre de Boole finie est isomorphe, en tant que treillis, au treillis des parties d'un certain ensemble fini.

L'ensemble "générateur" de l'algèbre de Boole finie est celui de ses atomes, qui est à l'algèbre finie ce que les lettres d'un alphabet sont aux mots.

II Fonctions booléennes

Définition 5.8. On appelle fonction booléenne de n variables toute application f de $\mathscr{B}^n$ dans $\mathscr{B}$.

Remarque 28. Une fonction boolénne peut être donnée explicitement par une table de vérité.

1 Montrer qu'il existe 2^{2^n} fonctions booléennes de n variables. Déterminer toutes les fonctions booléennes de deux variables, à titre d'exercice.

Remarque 29. Une fonction booléenne $f : \mathscr{B}^n \to \mathscr{B}$ s'interprète géométriquent en termes de k-cube ($k \leq n$) "couvert" par f, i.e. la préimage $f^{-1}(1)$.

II.1 Forme canonique des fonctions booléennes

Théorème 5.9

Soit f une fonction de $\mathscr{B}^n$ dans $\mathscr{B}$. Alors f peut s'écrire sous forme canonique

a. Disjonctive, c'est-à-dire :

$$\forall (x_1,...,x_n) \in \mathscr{B}^n, f(x_1,...,x_n) = \sum f(a_1,...,a_n).x_1^{a_1}...x_n^{a_n},$$

avec $x^a = x$ si $a = 1$ et $x^a = x'$ si $a = 0$.

b. Conjonctive c'est-à-dire :

$$\forall (x_1,...,x_n) \in \mathscr{B}^n, f(x_1,...,x_n) = \prod (f(a_1,...,a_n) + x_1^{a_1} + ... + x_n^{a_n}),$$

avec $x^a = x$ si $a = 0$ et $x^a = x'$ si $a = 1$.

2 Écrire sous forme canonique des fonctions booléennes données.

Le thèorème de structure assure qu'une fonction boolénne est nécessairement un polynôme des variables booléennes directes ou accentuées.

Définition 5.10. Un monôme (ou une fonction booléenne monôme) de degré n est un produit de n variables booléennes distinctes, les unes sous forme directe, les autres sous forme accentuée.

Définition 5.11. Un polynôme (ou une fonction booléenne polynôme) est une somme de monômes.

Définition 5.12. Un monôme de degré n est dit canonique.

3 Montrer qu'il existe 2^n monômes canoniques sur $\mathscr{B}^n$.

4 Interpréter les monômes de degré p ($p \leq n$) en termes de $(n-p)$-faces.

5 On pose $n = 3$

a. Écrire sous forme disjonctive la fonction f de trois variables booléennes qui prend la valeur 1 sur $\{001, 011, 100, 101\}$.

b. Écrire sous forme conjonctive la fonction g de trois variables booléennes qui prend la valeur 0 sur $\{000, 010, 110, 111\}$.

c. Remarques ? Quelles sont les $(3-p)$-faces couvertes par f et g ?

III Simplification des fonctions booléennes

Définition 5.13. On appelle forme polynômiale d'une fonctions booléenne sa forme canonique disjonctive.

Dans ce qui suit on cherchera à simplifier une fonction booléenne sous forme polynomiale.

Lemme 1. Soit f_1 et f_2 deux fonctions boléennes, alors

$$(f_1 \leq f_2) \Leftrightarrow \forall x \in \mathscr{B}^n \quad (f_1(x) = 1 \Rightarrow f_2(x) = 1)$$

Lemme 2. Soit m_1 et m_2 deux monômes, alors

$$(m_1 \leq m_2) \Leftrightarrow \left\{ \begin{array}{c} \text{Toute variable présente dans } m_2 \\ \text{figure dans } m_1 \text{ de façon identique} \\ \text{en termes d'accentuation ou non.} \end{array} \right\}$$

Remarque 30. On considère la fonction nulle comme une fonction monôme.

Proposition 5.14

L'ensemble des monômes inférieurs à une fonction f donnée est fini et non vide.

Définition 5.15. On appelle monôme premier d'une fonction booléenne, un monôme inférieur à cette fonction tel qu'aucun de ses facteurs ne soit inférieur à cette fonction.

Remarque 31. Un monôme premier de f représente une face maximale de l'hypercube associé, couverte par la fonction f. On retrouve en particulier des phénomènes observés lors de l'utilisation des tables de Karnaugh (voir TD et annexe) !

Théorème 5.16

Toute fonction booléenne de n variables est somme de ses monômes premiers.

Remarque 32. La somme considérée n'est pas nécessairement minimale ; pourtant, simplifier une fonction booléenne passe par la détermination de ses monômes premiers.

L'étude de la complexité des algorithmes comporte deux aspects, l'un spatial, l'autre temporel. Dans ce chapitre on abordera seulement l'aspect temporel. Commençons par un exemple classique :

1 On considère le problème classique du voyageur de commerce : un voyageur doit visiter n villes : $v_1, \ldots, v_n$. Les distances sont connues entre chaque ville et on demande au voyageur de réaliser son parcours en moins de D kilomètres.

a. Proposer un algorithme simple qui résolve le problème, i.e. qui étant donné une liste de ville et de distance (par exemple donné sous forme d'une matrice symétrique), renvoie après l'avoir calculé une séquence $(i_1, \ldots, i_n)$ d'ordre de passage des villes qui satisfait la condition sur la distance parcourue.

b. On suppose qu'il faut 1 nano seconde ($10^{-9}s$) à la machine pour calculer la distance correspondant à une séquence donnée. Combien de temps nécessite la résolution du problème avec votre algorithme pour $n = 10$, $n = 20$ et $n = 32$ villes ?

I Mesure de complexité

Soit $\mathscr{A}$ un algorithme résovant un problème ($\mathscr{P}$) sur des données de taille n ($n \in \mathbb{N}^*$).

Définition 6.1. On appelle fonction complexité temporelle de $\mathscr{A}$, la fonction $f : \mathbb{N} \to \mathbb{R}$ telle que, pour des données de tailles n, $\mathscr{A}$ résout dans le pire des cas, le problème ($\mathscr{P}$) en réalisant $f(n)$ opérations (étapes) élémentaires

Remarque 33. On peut également évaluer la complexité au meilleur des cas, c'est-à-dire trouver la fonction $f : \mathbb{N} \to \mathbb{R}$ qui parmi les entrées de taille n évaluera le nombre d'opérations élémentaires pour l'entrée qui minimise le nombre d'opérations effectuées. Mais cette information n'est pas forcément celle qu'on cherche. En effet on veut se préserver du pire et donc c'est ce cas qu'on cherche à évaluer.

Remarque 34. Dans le cas où "le pire des cas" correspond à des données d'entrées "rares" on peut construire une fonction qui évaluera en "moyenne" le nombre d'opérations effectuées. Pour cela on construit plusieurs fonctions complexités (au pire des cas, au meilleur, au cas générique, etc...) et on pondère chaque fonction par la probabilité d'obtenir une entrée du cas correspondant.

Dans un premier temps pour évaluer la complexité temporelle d'un algorithme donné, on pourra attribuer des valeurs arbitraires $\alpha, \beta, \gamma, \ldots$ aux opérations élémentaires effectuées par l'algorithme et calculer la fonction complexité temporelle en fonction de n et de ces constantes.

2 Calculer la fonction complexité temporelle de l'algorithme qui résout le problème du voyageur de commerce.

3 On rappelle l'algorithme suivant utiliser dans l'antiquité pour calculer le produit de deux entiers.

Algorithme 1

```
Function mult(x, y)
r := 0
while x ≠ 0 do
    if x est impair then
        r := r + y
        x := x − 1
    end if
    x := x/2
    y := y × 2
end while
Return r
```

a. Montrer que pour tout $x \in \mathbb{N}$, il existe $n \in \mathbb{N}$ tel que $2^n \leq x < 2^{n+1}$.

b. Montrer qu'après le k-ème passage en boucle la valeur actualisée de x est $[\frac{x}{2^k}]$.

c. Que vaut $[\frac{x}{2^n}]$? En déduire, en fonction de x, le nombre de passage en boucle de l'algorithme.

d. Déterminer la fonction complexité de l'algorithme.

Remarque 35. Les constantes α, β, γ, etc, utilisées dans l'expression de la fonction complexité temporelle ne sont pas significatives. En effet elles dépendent des performances machine pour réaliser une tache élémentaire (opération arithmétique, assignation, lecture...). Elles ne traduisent pas une propriété intrinsèque de l'algorithme. De même pour les grandes valeurs de n il n'est pas nécessaire de connaître précisemment la fonction de complexité. Ce qui importe dans un premier temps c'est d'évaluer l'ordre de grandeur.

II Notations asymptotiques

Pour comparer asymptotiquement l'efficacité temporelle d'algorithmes, on compare leurs fonctions de complexité.

II.1 Notations $\mathcal{O}$, Ω et Θ

Définition 6.2. Soit $\mathcal{F} = \{f : \mathbb{N} \to \mathbb{R}^+\}$. Étant donné f de $\mathcal{F}$ on définit

$$\mathcal{O}(f) = \{g \in \mathcal{F} / \exists c \in \mathbb{R}^{+*}, \exists n_0 \in \mathbb{N}, \forall n \geq n_0, g(n) \leq cf(n)\}$$

L'ensemble $\mathcal{O}(f)$ est l'ensemble des fonctions asymptotiquement dominées par f.

4 On établira les résultats suivants :

a. $2^{n+1} \in \mathcal{O}(2^n)$

b. $(n+1)! \notin \mathcal{O}(n!)$

c. Si $f(n) \in \mathcal{O}(n)$ alors $2^{f(n)} \in \mathcal{O}(2^n)$ n'est pas vraie en général.

Définition 6.3. Étant donné f de $\mathscr{F}$ on définit

$$\Omega(f) = \{g \in \mathscr{F} / \exists c \in \mathbb{R}^{+*}, \exists n_0 \in \mathbb{N}, \forall n \geq n_0, g(n) \geq cf(n)\}$$

L'ensemble $\Omega(f)$ est l'ensemble des fonctions dominant asymptotiquement f.

Définition 6.4. Étant donné f de $\mathscr{F}$ on définit

$$\Theta(f) = \left\{g \in \mathscr{F} / \exists (c,c') \in \left(\mathbb{R}^{+*}\right)^2, \exists n_0 \in \mathbb{N}, \forall n \geq n_0, cf(n) \leq g(n) \leq c'f(n)\right\}$$

On dit alors que f et g sont dites asymptotiquement équivalentes ou ont des croissances asymptotiques équivalentes.

Proposition 6.5

On a la propriété suivante :

$$f(n) \in \Theta(g(n)) \Leftrightarrow f(n) \in \mathscr{O}(g(n)) \text{et } g(n) \in \mathscr{O}(f(n))$$

Proposition 6.6

Sous les notations antérieures on a :

a. Si $\lim\limits_{n\to+\infty} \frac{f(n)}{g(n)} = l \in \mathbb{R}^{+*}$ alors $f(n) \in \Theta(g(n))$ et inversement.

b. Si $\lim\limits_{n\to+\infty} \frac{f(n)}{g(n)} = l \in \mathbb{R}^{+}$ alors $f(n) \in \mathscr{O}(g(n))$, mais on n'a pas en général $f(n) \in \Theta(g(n))$

c. Si $\lim\limits_{n\to+\infty} \frac{f(n)}{g(n)} = +\infty$ alors $f(n) \in \Omega(g(n))$, mais on n'a pas en général $f(n) \in \Theta(g(n))$

Proposition 6.7

On considère sur $\mathscr{F}$ la relation $f \sim g \Leftrightarrow f \in \Theta(g)$. Alors $\sim$ est une relation d'équivalence sur $\mathscr{F}$.

Remarque 36. La notation Θ est propre à l'analyse des algorithmes. En mathématiques, il existe une notion d'équivalence asymptotique sur les fonctions définie de la manière suivante

$$f \mathscr{R} g \Leftrightarrow \lim_{n\to\infty} \frac{f(n)}{g(n)} = 1 \tag{6.1}$$

Cette définition est plus restrictive que la notion de classe d'équivalence au sens de Θ.

Déterminer la complexité asymptotique d'un algorithme revient à déterminer une fonction «simple» g telle que la fonction complexité f de $\mathscr{A}$ vérifie $f \in \Theta(g)$. Évaluer l'efficacité (asymptotique) d'un algorithme $\mathscr{A}_1$ par rapport à un algorithme $\mathscr{A}_2$ revient à comparer les fonctions complexités f_1 et f_2 ou plus simplement comparer leur classe de complexité.

Remarque 37. En pratique l'étude asymptotique doit s'accompagner de tests expérimentaux pour déterminer l'influence des constantes. En effet un algorithme $\mathscr{A}_1$ peut être asymptotiquement meilleur que $\mathscr{A}_2$ mais cette efficacité ne peut être réelle qu'à partir d'un seuil déjà trop grand pour les applications de l'algorithme.

II.2 Ordre de grandeurs (classe de complexité)

On retiendra l'ordre de comparaison suivant sur les classes de complexité

Classe	$\Theta(C)$	Commentaire
constant	1	très rapide
logarithmique	$\log(n)$	très rapide
linéaire	n	très rapide
quasilinéaire	$n\log(n)$	rapide
polynômiale	n^α $(\alpha > 1)$	faisable sur un ordinateur
exponentielle	$e^{\alpha n} = k^n$ $(\alpha > 0, k > 1)$	infaisable
factorielle	$n!$	infaisable

Définition 6.8. Soit f et g deux fonctions de $\mathscr{F}$. On dit que f est négligeable relativement à g au voisinage de $+\infty$, et on note alors $f = o(g)$ (prononcé f égal petit o de g)ssi on a :

$$\forall \varepsilon \in \mathbb{R}^{+*}, \exists n_0 \in \mathbb{N}, \forall n \geq n_0, f(n) \leq \varepsilon g(n)$$

Proposition 6.9

Soit $f, g \in \mathscr{F}$, telle que $g(n) \neq 0$, $\forall n \in \mathbb{N}$, alors $f = o(g)$ si et seulement si $\lim_{n\to+\infty} \dfrac{f(n)}{g(n)} = 0$

Sous les notations antérieures, le tableau précédent fournit une l'échelle de comparaison, chaque fonction (et donc classe) étant négligeable par rapport à la suivante :

$$\mathscr{E} = \{1, \ln(n), n^\varepsilon, n, n\ln(n), n^c, k^n, n!\}$$

Remarque 38. Les classes de complexité polynomiale et exponentielle sont au cœur de la fameuse conjecture [1] $\mathbf{P} \neq \mathbf{NP}$ en informatique théorique. La classe **P** représente l'ensemble des problèmes résolubles par un algorithme de complexité polynomiale. La classe **NP** représente l'ensemble des problèmes pour lesquels on ne connait pas d'algorithme qui résout le problème en temps polynomial mais pour lesquels on peut vérifier en temps polynomial la validité d'une solution. Il est clair que $\mathbf{P} \subset \mathbf{NP}$ mais l'inlcusion stricte ou l'inclusion réciproque sont toujours des questions ouvertes.

5 Montrer que le problème du voyageur de commerce est **NP**.

1. La fondation Clay propose un prix de 1000000$ pour sa résolution

Dans ce chapite on propose quelques outils standard utiles pour les problèmes de dénombrement présent en informatique lors de l'évaluation des performances d'un algorithme (complexité), l'étude d'un graphe ou d'une structure d'arbre.

Les ensembles considérés dans ce chapitre sont de cardinaux finis.

I Outils classiques

On rappelle un résultat déjà présent au chapitre 2

Proposition 7.1

Etant donné les deux ensembles A et B, alors

$$|A \times B| = |A| \times |B| \tag{7.1}$$

Une traduction utile de cette proposition est la règle dite "règle du produit"

Proposition 7.2

Si une procédure peut se décomposer en deux étapes notées E_1 et E_2 et si le nombre de résultats possibles pour l'étape E_1 est $m = |E_1|$ et le nombre de résultats possibles pour E_2 est $n = |E_2|$, alors le nombre de possibilités pour réaliser la procédure en suivant les étapes E_1 et E_2 est mn.

1 Soit $A = \{a, b, c\}$ un alphabet à trois lettres. Déterminer le nombre de mots de 4 lettres qu'on peut écrire sur cet alphabet.

Une autre définition du chapitre 2.

Définition 7.3. Etant donné les deux ensembles A et B, on appelle injection de A dans B toute fonction vérifiant

$$\forall (a, a') \in A^2 \quad [f(a) = f(a') \Rightarrow a = a'] . \tag{7.2}$$

Proposition 7.4

Si on note $|A| = p$ et $|B| = n$, alors le nombre d'injections de A dans B, noté A_n^p, est appelé le nombre d'arrangements de p éléments parmi n. Il vérifie :

$$A_n^p = \frac{n!}{(n-p)!} = n(n-1)...(n-p+1). \tag{7.3}$$

Si $p = n$, l'injection considérée est alors une bijection (appelée encore, permutation de $\{1, ..., n\}$). Le nombre de permutations de $\{1, ..., n\}$ est donné par

$$A_n^n = n! \tag{7.4}$$

2 Donner le nombre de façon de placer 7 invités autour d'une table ronde.

Définition 7.5. Soit E un ensemble de cardinal n ; soit p de $\mathbb{N}$ ($p \leq n$) On appelle combinaison de p éléments parmi n le nombre de parties de E contenant p éléments.

Proposition 7.6

Sous les notations précédentes, on a

$$\binom{n}{p} = \frac{n!}{(n-p)!p!} \tag{7.5}$$

Proposition 7.7

Sous les notations précédentes on a,

$$\forall (n,p) \in \mathbb{N}^* \times \mathbb{N}^*, \binom{n}{p} = \binom{n-1}{p} + \binom{n-1}{p-1} \tag{7.6}$$

3 Étant donné un quadrillage du plan, combien de chemins, définis par une succession de mouvement D (à droite) et H (en haut) permettent de joinde le point de coordonnées $(0,0)$ au point (m,n).

II Principe d'inclusion-exclusion

Ce principe est en fait un théorème qui s'énnonce ainsi,

Théorème 7.8

Soit E un ensemble et $A_1, ..., A_n$ des parties de E alors :

$$\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{i=1}^{n} |A_i| - \sum_{1\leq r\langle s\leq n}^{n} |A_r \cap A_s| + \sum_{1\leq r\langle s\langle l\leq n}^{n} |A_r \cap A_s \cap A_l| + ... + (-1)^{n-1} \left|\bigcap_{i=1}^{n} A_i\right| \tag{7.7}$$

C'est la généralisation de la formule bien connue sur les ensembles $|A \cup B| = |A| + |B| - |A \cap B|$.

On reprend dans cette annexe quelques algorithmes conduisant à la simplification des fonctions booléenes.

On considère ici une fonction booléenne f de n variables.

I Tables de Karnaugh

Cette méthode est bien connue des étudiants pour $n \leq 3$; on en fera une relecture à travers le concept de monômes premiers.

II Méthode de Quine-Mc Cluskey

Nous décrivons ici seulement le principe du logiciel, lourd car universel.

Principe

a. Dans une première étape, on écrit f sous forme disjonctive. Les variables x_i absentes dans l'écriture d'un bloc $\boxed{b}$ sont "forcées" en les faisant intervenir en tant que $\boxed{b}.1 = \boxed{b}.(x_i + x_i')$.
Cette première étape peut donner lieu à un algorithme particulier, dévolu à cette seule tâche.

b. Dans une seconde étape on regroupe progressivement
les faces de degré 0 [autrement dit les monômes de degré n] pour en faire
des faces de degré 1 [autrement dit des monômes de degré $(n-1)$]
et ainsi de suite.
Cette étape s'achève dès qu'il n'y a plus de regroupements possibles.

Mise en oeuvre Elle s'effectue via la construction d'un tableau.

a. Dans la première colonne, on groupe les monômes canoniques en classes, selon le nombre de variables accentuées qu'ils contiennent ; à l'intérieur des classes, ils sont écrits suivant l'ordre lexicographique de n-uplets non accentués, des variables intervenantes.

b. Les colonnes suivantes sont construites en regroupant, dans la colonne précédente, les monômes appartenant à deux classes voisines.
Par exemple, pour

$$f(x,y,z,t) = xyzt + x'yzt + xy'zt + xyzt' + x'y'zt + x'yz't + x'yzt' + xy'z't +$$
$$x'y'z't + x'y'zt' + x'y'z't'$$

On passe de la première colonne à la seconde comme suit :

numéro	monôme			via	
			yzt		0.1+1.1
			xzt		0.1+1.2
			xyz		0.1+1.3
0.1	$xyzt$		$x'zt$		1.1+2.1
1.1	$x'yzt$		$x'yt$		1.1+2.2
1.2	$xy'zt$		$x'yz$		1.1+2.3
1.3	$xyzt'$		$y'zt$		1.2+2.1
2.1	$x'y'zt$	à	$xy't$	via	1.2+2.4
2.2	$x'yz't$		yzt'		1.3+2.3
2.3	$x'yzt'$		$x'y't$		2.1+3.1
2.4	$xy'z't$		$x'y'z$		2.1+3.2
3.1	$x'y'z't$		$x'z't$		2.2+3.1
3.2	$x'y'zt'$		$x'zt'$		2.3+3.2
4.1	$x'y'z't'$		$y'z't$		2.4+3.1
			$x'y'z'$		3.1+4.1
			$x'y't'$		3.2+4.1

Le passage de la seconde à la troisième s'opère de même et on obtient :

$$f(x,y,z,t) = \underbrace{zt+yz}+\underbrace{x't+x'z+y't}+\underbrace{x'y'}$$

où les accolades rappellent la génération des sous-classes, lors de la deuxième passe...

Compléments Il resterait à démontrer

a. la convergence de l'algorithme proposé,

b. et à en faire l'étude de complexité... [1]

III Méthode du consensus

Cette méthode s'applique à une fonction donnée sous forme polynômiale quelconque.

Définitions

Définition 8.1. On dit que deux monômes m_1 et m_2 forment une paire productive lorsqu'une seule des variables apparaissant dans m_1 et m_2 est biforme, c'est-à-dire "directe" dans l'un et accentuée dans l'autre.

Sous cette condition, on appellera consensus de m_1 et m_2, le monôme noté $cons(m_1,m_2)$, défini comme le produit de tous les monômes "monoformes" de m_1 et m_2.

Exemple 20. On donne $m_1 = xyz'$; $m_2 = xy'zt$; $m_3 = x'yz$; $m_4 = xt'$

Les paires productives sont au nombre de deux :

- $\{m_2, m_4\}$ de consensus $cons(m_2,m_4) = xy'z$,
- $\{m_3, m_4\}$ de consensus $cons(m_3,m_4) = yzt'$.

Remarque 39. *Interprétation géométrique*

Se présentent deux cas :

1. ce qui n'est pas abordé ici...

- $m_1 = xp$ et $m_2 = x'p$ alors $cons(m_1, m_2) = p$ est un nouveau monôme supérieur à m_1 et m_2 qui vont ainsi s'éliminer.
- $m_1 = xp$ et $m_2 = x'q$ alors m_1 et m_2 "mettent en commun leurs sommets" pour constituer une nouvelle face couverte par $cons(m_1, m_2)$. La situation conduit à une simplification lorsque m_1 ou m_2 est majoré par $cons(m_1, m_2)$.

Algorithme du consensus

a. *Initialisation*
Eliminer de la liste les monômes couverts par d'autres.

b. *Deuxième phase : productrice de la liste des monômes premiers de f* [2]

- On répète
  Pour le 1°, puis le 2° etc..., de la liste en cours d'évolution
  - **i**. Former les consensus avec ceux qui le suivent.
  - **ii**. ajouter à la liste les consensus qui ne sont pas couverts par un monôme existant (phase de nettoyage).
- jusqu'à ce qu'il n'y ait plus de nouveaux monômes créés par consensus [3]

c. *Troisième phase : suppression des redondances éventuelles*
Voir le paragraphe suivant.

Suppression des redondances On sait que la somme des monômes premiers d'une fonction booléenne est égale à cette fonction ; toutefois la somme précitée n'est pas nécessairement minimale.

Définition 8.2. *Bases de fonctions booléennes*

a. On appelle base d'une fonction booléenne f tout ensemble de monômes premiers dont la somme est égale à cette fonction.

b. Une telle base est dite irredondante lorsqu'elle est minimale au sens de l'inclusion.

Proposition 8.3

Méthode d'obtention d'une base irredondante
L'idée est de couvrir de façon minimale les monômes canoniques couverts par f grâce aux monômes premiers $m_1, ..., m_p$ de f.

a. A chaque sommet P_i de l'hypercube associé à f ($i \in \{1, ..., k\}$), donc à chaque monôme canonique de f , on associe la somme s_i des indices des monômes premiers qui le couvrent.

b. Pour couvrir tous les P_i , il convient de choisir un indice dans chaque somme s_i ce qui ramène à transformer le produit de sommes $s_1 ... s_k$ en une somme de produits, en ne conservant que les termes multiples d'aucun autre,

c. ce qui revient à extraire les familles d'indices de monômes premiers intervenant pour la couverture de l'ensemble des P_i .

2. dont on doit savoir qu'elle n'est pas nécessairement irredondante...
3. on voit la convergence de l'algorithme proposé via l'interprétation géométrique des monômes premiers d'une fonction f.

Etudes d'exemples des différentes étapes

Exemple 21. Soit f donnée par :

$$f(x,y,z,t) = x'z' + xyt + x'yz + xy'z' + y'zt' + x'y'z'$$

a. La liste initiale simplifiée s'écrit :

$$L^{(0)} = \left[\underbrace{x'z'}_{1}, \underbrace{xyt}_{2}, \underbrace{x'yz}_{3}, \underbrace{xy'z'}_{4}, \underbrace{y'zt}_{5}\right].$$

b. Lors de la première passe

$$L^{(1)}_{ns} = \left[\underbrace{x'z'}_{1}, \underbrace{xyt}_{2}, \underbrace{x'yz}_{3}, \underbrace{xy'z'}_{4}, \underbrace{y'zt'}_{5}, \underbrace{yz't}_{cons(1,2)}, \underbrace{x'y}_{cons(1,3)}, \underbrace{y'z'}_{cons(1,4)}, \underbrace{x'y't'}_{cons(1,5)}\right],$$

c. qui se simplifie en

$$L^{(1)} = \left[\underbrace{x'z'}_{1}, \underbrace{xyt}_{2}, \underbrace{y'zt'}_{3}, \underbrace{yz't}_{4}, \underbrace{x'y}_{5}, \underbrace{y'z'}_{6}, \underbrace{x'y't'}_{7}\right];$$

d. et ainsi de suite, pour finir en

$$L^{(r)} = \left[\underbrace{x'z'}_{1}, \underbrace{x'y}_{2}, \underbrace{y'z'}_{3}, \underbrace{yt}_{4}, \underbrace{y't'}_{5}, \underbrace{z't}_{6}, \underbrace{x't'}_{7}\right].$$

e. Etude de la recherche de base irredondante (voir travaux dirigés).