Cours de mathématiques

TOME X

INFORMATIQUE

Mathématiques générales

France $\sim$ 2024

Écrit et réalisé par Louis Lascaud

Table des matières

Chapitre 1

Analyse numérique

1.1 Résolution de systèmes linéaires

Théorème. ***(Caractérisation de la solvabilité par l'orthogonal)***

Soit A une matrice de taille $m \times n$. L'équation $Ax = b$ admet au moins une solution si et seulement si b est orthogonal à $\mathrm{Ker}(A^*)$.

Chapitre 2

Théorie de la complexité

2.1 Machines de Turing

2.1.1 Définition et premières propriétés

Définition. ***(Machine de Turing)***

Une *machine de Turing (déterministe)* ou *MT* est

Reformulation pratique. ***(Définition informelle des machines de Turing)***

Une machine de Turing est composée des éléments suivants :

(i) un *ruban infini* dans les deux sens, ou peut-être plusieurs rubans, divisé en *cases* ;

(ii) un alphabet $\mathcal{A}$, souvent $\{0,1\}$, de sorte que les cases sont remplies des symboles de l'alphabet, à raison d'une lettre par case ; on rappelle qu'un alphabet sur un ensemble engendre un mot vide, dit ici *symbole blanc* ;

(iii) une *tête de lecture*, qui *est* dans un état ;

(iv) un ensemble fini Σ d'états.

À chaque *étape* d'un *calcul*, la tête est devant une case, lit son contenu et décide de trois choses : le contenu de ladite case à la prochaine étape, son propre état pour la prochaine étape, et si elle se déplace à droite ou à gauche. Autrement dit, la tête, ou *fonction de transition*, est une application

$$\delta : A \times \Sigma \to A \times \Sigma \times \{G,D\}.$$

De plus, on supose qu'il existe un état de Σ dit *halte* de sorte que si la machine arrive à cet état, elle s'*arrête*. La *sortie* de la machine est le symbole devant la tête quand la machine s'arrête. L'*entrée* de la machine est son état initial, *i.e.* à la première étape.

Définition. ***(Configuration d'une machine de Turing)***

La *configuration* (d'une machine de Turing à une étape) du calcul consiste en les symboles dans les cases du ruban à cette étape, l'état de la tête à cette étape et la position de la tête à cette étape.

2.1.2 Lecture seule, lecture unique

Définition. ***(Lecture seule)***

Un ruban d'une machine de Turing est dit *en lecture seule* si la machine ne peut pas changer le contenu des cases.

Définition. ***(Lecture unique)***

Un ruban d'une machine de Turing est dit *en lecture unique* si après avoir lu une case du ruban, la machine doit se déplacer à droite.

2.1.3 Fonctions booléennes

⟶ *Notation.* On note $\{0,1\}^* = \bigcup_{n=1}^{\infty} \{0,1\}^n$ l'ensemble des suites finies de 0 et de 1.

Définition. ***(Fonction booléenne)***

Une *fonction booléenne* est une fonction $\phi : \{0,1\}^* \to \{0,1\}$.

2.1.4 Temps de calcul

Définition. ***(Calcul d'une fonction par une machine)***

Soit ϕ une fonction booléenne. Soit T une machine de Turing. On dit que T *calcule* ϕ si pour chaque $x \in \{0,1\}^*$, si x est l'entrée de T (à comprendre : avec des symboles blancs dans les cases hors du mot formé par x), $\phi(x)$ est la sortie.

Définition. ***(Temps de calcul)***

Soit ϕ une fonction booléenne. Soit T une machine de Turing calculant ϕ. Soit $x \in \{0,1\}^*$. Le *temps de calcul* L de $\phi(x)$ par T est le nombre d'étapes de T avant que T ne s'arrête.

Le temps de calcul peut être infini.

2.1.5 Machine de Turing non déterministe

Définition. ***(Machine de Turing non déterministe)***

Une *machine de Turing non déterministe* ou *non déterminée* ou *MTND* est une machine avec deux fonctions de transitions, par exemple δ_0 et δ_1.

Définition. ***(Calcul d'une fonction par une machine non déterministe)***

Soit ϕ une fonction booléenne. Soit T une machine de Turing non déterministe. On dit que T *calcule* ϕ si pour chaque $x \in \{0,1\}^*$, alors

(i) $\phi(x) = 1$ si et seulement si l'on peut trouver une suite de configuartions $C_0,...,C_m, m \in \mathbb{N}$, où C_0 est la configuartion initiale avec entrée x, de sorte que l'état de T dans C_m est sa halte, sa sortie est 1, et pour tout $i \in [\![1,m]\!]$, $C_i = \delta_0(C_{i-1})$ ou $\delta_1(C_{i-1})$;

(ii) $\phi(x) = 0$ si et seulement si l'on peut trouver une suite de configuartions $C_0,...,C_m, m \in \mathbb{N}$, où C_0 est la configuartion initiale avec entrée x, de sorte que l'état de T dans C_m est sa halte, sa sortie est 0, et pour tout $i \in [\![1,m]\!]$, $C_i = \delta_0(C_{i-1})$ ou $\delta_1(C_{i-1})$.

2.2 Circuits booléens

2.2.1 Définitions

Définition. ***(Circuit booléen)***

Un *circuit booléen* est un graphe orienté acyclique fini, dont les sommets sont étiquetés par $\wedge$, $\vee$ ou $\neg$, sauf les sommets de degré d'entrée 0, qui sont les *entrées*. Les sommets de degré de sortie 0 sont les *sorties*.

Méthode. ***(Application pseudo-booléenne associée à un circuit booléen)***

À un circuit booléen à n entrées et m sorties, on peut associer une application $\varphi : \{0,1\}^n \to \{0,1\}^m$ défini par récursivement : si les entrées sont $v_1,...,v_n$, pour calculer un certain $\phi(x)$, on donne la valeur x_i en v_i, en respectant les règles suivantes.

1. La valeur d'un sommet étiqueté par $\wedge$ (une *porte* ET) est 1 si et seulement si tous ses enfants ont la valeur 1.
2. La valeur d'un sommet étiqueté par $\vee$ (une *porte* OU) est 1 si et seulement si il a un enfant de valeur 1.
3. Un sommet étiqueté par $\neg$ (une *porte* NON) a un seul enfant, et sa valeur est l'opposée de la valeur de l'enfant.

Lemme

Chaque application $\phi : \{0,1\}^n \to \{0,1\}$ eut être calculée par un circuit booléen de taille $\leqslant 2^{n+1} + n + 1$.

▷ Soit $X = \{x \in \{0,1\}^n \mid \phi(x) = 1\}$. Pour chaque $x \in X$, il existe un circuit tel que la sortie est 1 si et seulement si l'entrée est x. Si les sommes d'entrée sont $v_1,...,v_n$, si $x_i = 0$, on attache une porte NON à v_i, puis l'on attache une porte ET à toutes ses portes NON et aux sommets v_i tel que $x_i = 1$. On fait cela pour chaque $x \in X$, et l'on attache enfin une porte OU à toutes ses portes ET . ■

2.3 Classes de complexité usuelles

2.3.1 La classe P

Définition. ***(Classe de complexité*** P***)***

Soit ϕ une fonction booléenne. Alors $\phi \in$ P la *classe de complexité* P s'il existe un polynôme p en une variable tel que pour tout $x \in \{0,1\}^*$, il existe une machine de Turing T qui calcule $\phi(x)$ et le temps de calcul est au plus $p(|x|)$ où $|x|$ est la longueur de x. Autrement dit, toute entrée de ϕ se calcule par une TM en temps polynomial indépendant de la machine.

Exemples. ***(Problèmes de classe*** P***)***

1. Étant donné un graphe fini et deux sommets de ce graphe, pour $k \in \mathbb{N}$, le problème d'existence d'un chemin de longueur k sur ce graphe entre ces deux sommets, est encodable par une machine de Turing et se calcul en temps polynomial.

2.3.2 La classe NP

Définition. ***(Classe de complexité*** NP***)***

Soit ϕ une fonction booléenne. Alors $\phi \in P$ la *classe de complexité* NP s'il existe un polynôme p en une variable tel que pour tout $x \in \{0,1\}^*$, il existe une machine de Turing non nécessairement déterministe T calcule $\phi(x)$ et le temps de calcul est au plus $p(|x|)$ où $|x|$ est la longueur de x. Autrement dit, toute entrée de ϕ se calcule par une TM non déterminée en temps polynomial indépendant de la machine.

Exemples. ***(Problèmes de classe*** NP***)***

1. Le problème d'existence d'un cycle hamiltonien dans un graphe est de classe NP.

Conjecture ?

P = NP.

Théorème. ***(Définition équivalente de la classe*** NP***)***

Une fonction booléenne $\phi \in$ NP si et seulement s'il existe un polynôme $p \in \mathbb{R}[X]$ et une application $\psi \in$ P telle que $\phi(x) = 1 \iff \exists y \in \{0,1\}^{p(|x|)} \quad \psi(x,y) = 1$.

▷ Si ϕ peut être calculée en temps polynomial par une machine de Turing non déterministe, pour chaque x on eut choisir une suite $\varepsilon_1,...,\varepsilon_m$ où $m \in \mathbb{N}, m \leqslant p(|x|)$ tel que si l'on choisit les fonctions de transition $\delta_{\varepsilon_1},...,\delta_{\varepsilon_m}$, on calcule $\phi(x)$. Alors, soit ψ l'application qui calcule $\phi(x)$ étant donnés x et $\varepsilon_1,...,\varepsilon_m$. Il est clair que $\psi \in P$.

Réciproquement, si une telle ψ existe, on peut utiliser une machine de Turing non déterministe pour écrire y et calculer $\psi(x,y)$. ■

La quantification introduite dans cette caractérisation induit la description de plusieurs autres classes de complexité.

Définition. ***(Classe de complexité*** co-NP***)***

Une fonction booléenne $\phi \in$ co-NP si et seulement s'il existe un polynôme $p \in \mathbb{R}[X]$ et une application $\psi \in$ P telle que $\phi(x) = 1 \iff \forall y \in \{0,1\}^{p(|x|)} \quad \psi(x,y) = 1$.

Reformulation pratique. ***(Calcul de*** co-NP***)***

co-NP $= 1 - NP = \{1 - \phi, \phi \in \text{NP}\}$.

Reformulation pratique. ***(Définition de la classe*** co-NP***)***

Cette autre définition n'est donc pas du tout symétrique !

Conjecture ?

NP = co-NP.

2.3.3 La classe PH

On définit par récurrence :

- ⋆ $\Sigma_0 = \Pi_0 =$ P ;
- ⋆ $\Sigma_1 =$ NP, $\Pi_1 =$ co-NP ;
- ⋆ soit $i \in \mathbb{N}$. On caractérise : $\phi : \Sigma_{i+1}$ s'il existe $\psi \in \Pi_i$ telle qu'il existe p polynomiale

telle que $\phi(x) = 1 \iff \exists y \in \{0,1\}^{p(|x|)} \quad \psi(x,y) = 1$; de même, $\phi \in \Pi_{i+1}$ s'il existe $\psi \in \Sigma_i$ telle qu'il existe p polynomiale telle que $\phi(x) = 1 \iff \forall y \in \{0,1\}^{p(|x|)} \quad \psi(x,y) = 1$.

Définition. ***(Classe de complexité*** PH***)***

On définit $\mathrm{PH} = \bigcup_{n=0}^{\infty} (\Sigma_n \cup \Pi_n)$.

Heuristique

On croit que la hiérarchie polynomiale est une vraie hiérarchie, mais en général, il est très difficile de montrer qu'une classe de complexité est strictement incluse dans une autre. Au-delà du problème très célèbre de savoir si oui ou non P = NP, on ignore encore si l'égalité est infirmée pour des classes de complexité beaucoup plus faibles.

2.3.4 Les classes EXPTIME et NEXPTIME

Définition. ***(Classe de complexité*** EXPTIME***)***

Une fonction booléenne $\phi \in$ EXPTIME si et seulement si on peut calculer $\phi(x)$ en temps $\exp(p(|x|))$ pour un certain polynôme p.

Définition. ***(Classe de complexité*** NEXPTIME***)***

Une fonction booléenne $\phi \in$ NEXPTIME si et seulement si on peut calculer $\phi(x)$ avec une MTND en temps $\exp(p(|x|))$ pour un certain polynôme p.

2.3.5 La classe PSPACE

Définition. ***(Classe de complexité*** PSPACE***)***

Une fonction booléenne $\phi \in$ PSPACE si et seulement si on peut trouver une machine de Turing et un polynôme p tels que T calcule φ et pour chaque x, la position de T pendant le calcul est otujours entre $-p(|x|)$ et $p(|x|)$.

Exemple fondamental. ***(Problème de classe*** PSPACE***)***

Le problème de détermination du gagnant au jeu de go, si les joueurs utilisent des stratégies optimales, n'est pas PH mais est PSPACE.

2.3.6 La classe L

On veut définir la classe des problème n'utilisant que des ressources en quantité logarithmique. Problème, c'est trop fort : simplement l'écriture de l'entrée n'est pas logarithmique ! Il faut donc

y aller plus subtilement.

Définition. ***(Classe de complexité*** L***)***

Une fonction booléenne $\phi \in \mathrm{L}$ si et seulement si on peut trouver une machine de Turing avec deux rubans dont le premier est à lecture seule, contient l'entrée x, et le deuxième est le *ruban de travail*, de sorte que pour chaque x, la machine ne regarde que $\mathcal{O}(\log(|x|))$ de cases du ruban de travail.

Heuristique

C'est la complexité minimale raisonnable.

Conjecture ?

$\mathrm{L} \subsetneq \mathrm{PH}$.

Mnémonik : on ne peut pas prouver que les miracles n'existent pas...

Définition. ***(Classe de complexité*** NL***)***

Une fonction booléenne $\phi \in \mathrm{L}$ si et seulement si on peut trouver une machine de Turing non déterministe avec deux rubans dont le premier est à lecture seule, qui contient l'entrée x, et le deuxième est le *ruban de travail*, de sorte que pour chaque x, la machine ne regarde que $\mathcal{O}(\log(|x|))$ de cases du ruban de travail.

On voudrait dire : *"une fonction booléenne $\phi \in \mathrm{NP}$ si et seulement s'il existe un polynôme $p \in \mathbb{R}[X]$ et une application $\psi \in \mathrm{L}$ telle que $\phi(x) = 1 \iff \exists y \in \{0,1\}^{p(|x|)} \quad \psi(x,y) = 1$"*, mais ce n'est pas le cas.

On peut bel et bien caractériser, de même qu'avec P et NP, la classe NL à partir de L, mais il faut être plus fin.

Théorème. ***(Définition équivalente de la classe*** NL***)***

Une fonction booléenne $\phi \in \mathrm{NL}$ si et seulement s'il existe une machine de Turing à trois rubans dont le premier est à lecture seule et contient l'entrée x, le deuxième est à lecture unique et le troisième est un ruban de travail, dont la machine n'utilise qu'une quantité logarithmique de cases, avec : $\phi(x) = 1$ si et seulement s'il existe y et ψ, calcule par une machine de Turing avec ses propriétés telle que si x et dans le remier ruban et y est dans le deuxième, alors la sortie de T est $\varphi(x)$.

⊛ *(Idée de la preuve.)* C'est un peu plus subtil. ■

2.3.7 La classe P/poly

Définition. ***(Classe de complexité*** P/poly***)***

Une fonction booléenne $\phi \in$ P/poly si et seulement si on peut trouver un polynôme p et pour chaque $n \in \mathbb{N}$, il existe une suite finie $(y_n) \in \{0,1\}^{p(n)}$ et une application $\psi : \{0,1\}^* \to \{0,1\}$ dans P telle que $\forall x \quad \varphi(x) = \psi(x,y_n)$. On appelle (y_n) une *suite de conseil*.

Chapitre 3

Exercices

Difficulté des exercices :

- ● ○ ○ ○ ○ Question de cours, application directe, exercice purement calculatoire sans réelle difficulté technique
- ● ● ○ ○ ○ Exercice faisable, soit intuitivement, soit en employant des moyens rudimentaires ou des techniques déjà vues
- ● ● ● ○ ○ Exercice relativement difficile et dont la résolution appelle à une réflexion plus importante à cause d'obstacles techniques ou conceptuels, qui cependant devraient être à la portée de la plupart des étudiants bien entraînés
- ● ● ● ● ○ Exercice très exigeant, destiné aux élèves prétendant aux concours les plus difficiles, exercice « classique ».
- ● ● ● ● ● La résolution de l'exercice requiert un raisonnement et des connaissances extrêmement avancés, dépassant les attentes du prérequis. Il est presque impossible de le mener à terme sans indication. Bien qu'exigibles à très peu d'endroits, ces exercices sont très intéressants et présentent souvent des résultats forts.

Appendice

Bibliographie

[1] *Titre du livre*, Auteur du livre, date, maison d'édition

Table des figures

Liste des tableaux