Fondements de l'informatique: Examen Durée: 3h

Sujet proposé par Olivier Bournez
Version 1.3

L'énoncé comporte 5 parties indépendantes, qui pourront être traitées dans un ordre quelconque. En revanche, dans chaque partie, il peut être utile, dans la réponse à une question, d'utiliser les questions précédentes! On pourra librement admettre le résultat d'une question pour passer aux questions suivantes. La difficulté des questions n'est pas une fonction linéaire ni croissante de leur numérotation.

Dans tout l'énoncé, et en particulier dans la partie "Machines de Turing", on demande des algorithmes et des solutions à un haut niveau: dans aucune des questions il n'est demandé de décrire complètement une machine de Turing, ni même d'en donner une description graphique; on pourra se contenter pour décrire un algorithme de le décrire par exemple en français ou dans un langage de programmation classique comme JAVA, C ou CAML.

1 Machines de Turing

On rappelle qu'une fonction $f : \mathbb{N} \to \mathbb{N}$ est calculable si il existe une machine de Turing qui partant du codage en binaire de n sur son ruban finit, au bout d'un temps fini, par arriver dans son état d'acceptation avec $f(n)$ écrit en binaire sur son ruban.

Question 1.1. *Montrer que l'inverse d'une fonction $f : \mathbb{N} \to \mathbb{N}$ calculable et bijective est calculable.*

Question 1.2. *Soit C un langage sur un alphabet fini Σ. Montrer que C est récursivement énumérable si et seulement si il existe un langage décidable D tel que*

$$C = \{w | \exists u \ (w, u) \in D\}.$$

Remarque: D est un langage de paires (plus précisément de couples) de mots: (w, u) désigne, comme dans les transparents du cours (par un léger abus de notation) un codage fixé de la paire (du couple) (w, u). Le codage précis n'est pas important à partir du moment où l'on peut retrouver w et u à partir du codage de (w, u), et qu'on peut construire le codage de (w, u) à partir de w et de u. Note: dans le polycopié, on distingue les couples/paires (w, u) et leur codage $< w, u >$ (voir page 114): avec les notations du polycopié, on écrirait $C = \{w | \exists u \ < w, u > \in D\}$.

Question 1.3. *Parmi les problèmes suivants, lesquels sont décidables, lesquels sont indécidables. Prouver formellement votre réponse.*

1. *Déterminer si le langage accepté par une machine de Turing M ne contient que des palindromes (un palindrome est un mot dont l'ordre des lettres reste le même qu'on le lise de gauche à droite ou de droite à gauche, comme "anna").*

2. *Déterminer si une machine de Turing M et un mot w sont tels que M accepte l'entrée w en n'utilisant (= écrivant) aucune autre case que celles qui contiennent initialement le mot w.*

3. *Déterminer si une machine de Turing M et un mot w sont tels que M accepte l'entrée w en n'utilisant aucune des cases initialement blanches à gauche de celles qui contiennent initialement w.*

2 Elimination des quantificateurs

Rappels:

- Une théorie T est un ensemble de formules sur une signature Σ.
- Deux formules F et F' sont équivalentes dans une théorie T si et seulement si $F \Leftrightarrow F'$ est une conséquence de T.
- Une formule sans quantificateur est une formule qui ne contient aucun symbole $\exists$ et aucun symbole $\forall$.

On dit qu'une théorie T sur la signature Σ permet l'élimination des quantificateurs dans une formule F (sur Σ) s'il existe une formule F' (sur Σ) sans quantificateur qui lui est équivalente dans la théorie T.

On dit que T permet l'élimination des quantificateurs sur Σ si T permet l'élimination des quantificateurs dans toute formule F sur Σ.

Par exemple, on peut montrer[1] que la théorie T qui correspond aux corps réels clos permet l'élimination des quantificateurs: par exemple, la formule

$$\exists x \; a * x * x + b * x + c = 0$$

(qui possède un quantificateur existentiel) est équivalente à la formule

$$(b * b - 4 * a * c \geq 0 \wedge a \neq 0) \vee (a = 0 \wedge b \neq 0) \vee (a = 0 \wedge b = 0 \wedge c = 0)$$

(qui n'en possède pas)[2] sur la théorie T.

Le but de cet exercice est de prouver qu'une autre théorie T, la théorie des ordres denses avec premier et dernier élément, permet l'élimination des quantificateurs.

Débutons par quelques généralités, avant de définir cette théorie T.

2.1 Généralités

Question 2.1. *Montrer que si une théorie T sur une signature Σ permet l'élimination des quantificateurs sur une formule F, alors elle permet l'élimination des quantificateurs sur la formule $\neg F$.*

On rappelle que toute formule F est équivalente à une formule en forme normale prénexe (Proposition 5.4 page 68 du polycopié).

Question 2.2. *Montrer que pour qu'une théorie T permette l'élimination des quantificateurs sur la signature Σ, il faut et il suffit que T permette l'élimination des quantificateurs dans toute formule de la forme $\exists x \; H$, où H est sans quantificateur.*

[1] Ce qui est hors de l'ambition de ce sujet.

[2] Bien entendu, ici 4 désigne $1 + 1 + 1 + 1$. De même, $a \neq 0$, par exemple, représente $\neg(a = 0)$.

Question 2.3. *Montrer que pour qu'une théorie T permette l'élimination des quantificateurs sur Σ, il faut et il suffit que T permette l'élimination des quantificateurs dans toute formule F sur Σ de la forme $\exists x\ (\alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_r)$, où chaque α_i est soit une formule atomique sur Σ ou la négation d'une telle formule.*

On pourra utiliser le fait que $\exists x(A \vee B)$ est équivalent à $(\exists xA) \vee (\exists xB)$, pour des formules A et B.

2.2 Ordres denses avec premier et dernier élément

On considère la signature Σ suivante: 2 symboles de constantes 0 et 1; 2 symboles de relations $<$ et $=$ d'arité 2.

La théorie T des *ordres denses avec premier et dernier élément* est la théorie sur la signature Σ qui contient les formules suivantes:

1. $\forall x \neg(x < x)$; (ordre strict)
2. $\forall x \forall y \forall z((x < y \wedge y < z) \Rightarrow x < z)$ (ordre transitif)
3. $\forall x \forall y(x = y \vee x < y \vee y < x)$ (axiome de relation d'ordre total)
4. $\forall x \forall y \exists z(x < y \Rightarrow (x < z \wedge z < y))$ (axiome de densité)
5. $\forall x(x = 0 \vee 0 < x)$ (axiome du 1ier élément)
6. $\forall x(x = 1 \vee x < 1)$ (axiome du dernier élément)

On suppose la théorie égalitaire[3]: cela revient soit à considérer qu'on impose que le symbole $=$ est nécessairement interprété par l'égalité, soit à supposer que T contient aussi les axiomes de l'égalité: $\forall x\ x = x$, $\forall x \forall y\ (x = y \Rightarrow y = x)$ et $\forall x \forall y \forall z(((x = y) \wedge (y = z)) \Rightarrow x = z)$, $\forall x \forall x' \forall y, (x = x') \Rightarrow ((x < y) \Leftrightarrow (x' < y))$, $\forall x \forall y \forall y', (y = y') \Rightarrow ((x < y) \Leftrightarrow (x < y'))$.

Comme annoncé, on veut montrer que T permet l'élimination des quantificateurs sur Σ.

Question 2.4. *Montrer qu'on est ramené à prouver l'élimination des quantificateurs dans une formule de la forme $\exists x(\alpha_1 \wedge \cdots \wedge \alpha_r)$, où chaque α_i est de la forme $t_1 < t_2$ ou $t_1 = t_2$.*

Question 2.5. *Prouver que T permet l'élimination des quantificateurs sur Σ.*

3 Spectres

Soit Σ une signature, et F une formule close sur Σ. On appelle *spectre* de F, et on note $Sp(F)$ l'ensemble des entiers naturels n tels que F admette au moins un modèle dont l'ensemble de base possède exactement n éléments.

Question 3.1. *Proposer une signature Σ et une formule close F dont le spectre est l'ensemble des entiers naturels pairs.*

Question 3.2. *Montrer que toute théorie dont le spectre est infini possède au moins un modèle dont l'ensemble de base est infini. (indication: on pourra utiliser le théorème de compacité et et une famille de formules bien choisie).*

[3] Voir le polycopié page 72.

4 NP-complétude

On a vu en cours que le problème de décision 3-COLORABILITE*:*

- ***Donnée****: Un graphe (fini)* $G = (V, E)$*.*
- ***Réponse****: Décider s'il existe un coloriage du graphe utilisant au plus* 3 *couleurs*

est NP-complet (On rappelle que le coloriage d'un graphe est un façon de colorer les sommets du graphe de telle sorte qu'aucune arête possède deux extrémités de la même couleur).

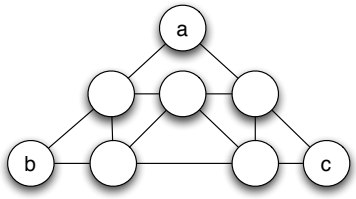
On s'intéresse à prouver la NP-complétude de différentes variantes du problème 3-COLORABILITE dans cet exercice: on utilisera (ou pourra utiliser) à chaque fois la NP-complétude du problème 3-COLORABILITE.

Question 4.1. *Montrer que le problème de décision* 4-COLORABILITE*:*

- ***Donnée****: Un graphe (fini)* $G = (V, E)$*.*
- ***Réponse****: Décider s'il existe un coloriage du graphe utilisant au plus* 4 *couleurs.*

est NP-complet.

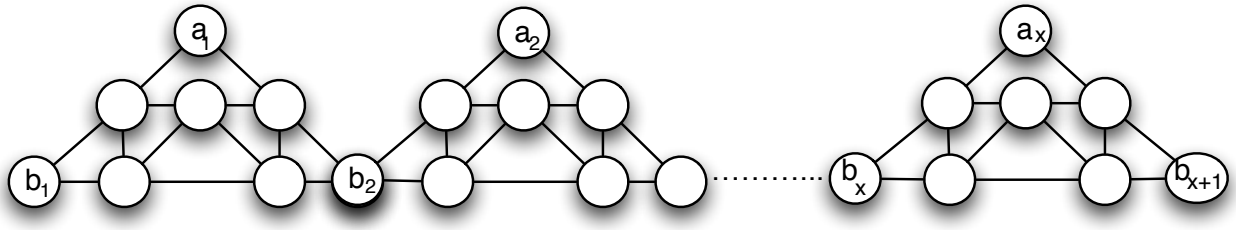
On considère dans les questions qui suivent le graphe H suivant:



Question 4.2. *Montrer que ce graphe ne possède pas de coloration du graphe utilisant au plus* 2 *couleurs.*

Question 4.3. *Montrer que dans toute coloration du graphe utilisant au plus* 3 *couleurs,* a*,* b *et* c *sont de même couleur.*

Question 4.4. *On considère le graphe* H_x *correspondant à* x *copies de* H *de la façon suivante :*



Montrer que dans toute coloration du graphe H *utilisant au plus* 3 *couleurs, les sommets* $(a_i)_{1 \leq i \leq x}$*,* $(b_i)_{1 \leq i \leq x+1}$ *sont de même couleur.*

Question 4.5. *Prouver que le problème de décision* 4DEG-3-COLORABILITE*:*

- ***Donnée****: Un graphe (fini)* $G = (V, E)$ *dont tous les sommets sont de degré au plus* 4
- ***Réponse****: Décider s'il existe un coloriage du graphe utilisant au plus* 3 *couleurs.*

est NP-complet.

(Rappel: on admet la NP-complétude du problème 3-COLORABILITE*).*

5 Fonctions affines par morceaux

La configuration d'une machine de Turing (à 1 ruban) peut se coder par un triplet (q, u, v) comme dans le transparent 16 du cours 4: $q \in Q$ désigne l'état de la machine, et u, v désignent le contenu respectivement à gauche et à droite de la tête de lecture du ruban, la tête de lecture étant sur la première lettre de v. On suppose que v est écrit de gauche à droite, et u de droite à gauche.

On peut supposer sans perte de généralité que l'alphabet d'entrée et de travail (hormis le caractère B de blanc) est l'alphabet $\Sigma = \{0, 1\}$. On peut aussi supposer que $Q = \{0, 1, \ldots, m-1\}$.

Un mot $u = u_1 u_2 \ldots u_n \in \Sigma^*$ sur l'alphabet $\Sigma = \{0, 1\}$ peut se coder par le rationnel

$$\gamma(u) = \sum_{i=1}^{n} \frac{2u_i}{4^i}.$$

La configuration $C = (q, u, v)$ peut alors se coder par $(q, \gamma(u), \gamma(v)) \in [0, m] \times [0, 1]^2$.

Question 5.1. *Supposons que le programme de la machine de Turing ordonne, dans la configuration C, d'écrire le symbole $a \in \{0, 1\}$, de se déplacer vers la gauche et d'aller dans l'état q'.*

Montrer que le codage de la nouvelle configuration C' à partir de celle de C s'obtient en appliquant une fonction affine par morceaux à chaque composante de celle de C. Montrer que l'on peut supposer la fonction affine par morceaux continue.

Question 5.2. *Prouver que pour toute machine de Turing M, on peut construire une fonction continue affine par morceaux $f : \mathbb{R}^3 \to \mathbb{R}^3$ dont les itérations simulent l'évolution de M.*

Question 5.3. *Prouver que le problème suivant est indécidable: on se donne une fonction continue affine par morceaux $f : \mathbb{R}^3 \to \mathbb{R}^3$, un point de départ $x_0 \in \mathbb{R}^3$, un rectangle[4] R, et on veut savoir si la suite définie par $x_{t+1} = f(x_t)$ est telle que $x_t \in R$ pour un certain entier t.*

[4] sous-ensemble rectangulaire d'un hyperplan.