Corrigé de l'Épreuve d'Informatique
Concours C/S ENS Ulm et Lyon 1999

Bruno DURAND [*], Jean MAIRESSE [†] et Yves VERHOEVEN [‡]

Avant-propos de l'éditeur : Deux corrigés complets ont été reçus pour cette épreuve. L'un proposé par les auteurs du sujet, B. Durand et J. Mairesse, et l'autre par Y. Verhoeven, élève à l'ENS. La version proposée ci-dessous est celle des auteurs du sujet, et il y a été ajouté la solution de Y. Verhoeven lorsque celle-ci était différente ou complémentaire.

Thème du sujet : Automates cellulaires : propriétés de surjectivité, d'injectivité et de bijectivité sur différents types de configurations (notamment périodiques et finies) ; décidabilité de ces propriétés en dimension 1, puis en dimension supérieure.

Commentaire général : Une particularité du sujet est qu'il ne demande l'utilisation d'aucun résultat spécifique ou connaissance particulière. En revanche une bonne familiarité avec les structures discrètes et leur manipulation par des algorithmes s'avère très utile.

1 Introduction

Question 1.1

La question ne présente aucune difficulté. La fonction locale $\delta_{\mathcal{X}}$ correspond à l'opérateur booléen 'ou exclusif' ou encore à l'addition dans $\mathbb{Z}/2\mathbb{Z}$.

Question 1.2

On traite la seconde partie de la question. Soit l'automate cellulaire $\mathcal{A} = (1, \{0,1\}, 0, \delta)$ avec $\delta(0) = \delta(1) = 0$. Pour tout c, on a $\mathcal{A}(c)(i) = 0, i \in \mathbb{Z}$. En particulier $\forall p,\ \mathcal{A}(\mathcal{P}_p) \neq \mathcal{P}_p$ et $\mathcal{A}(\mathcal{P}) \neq \mathcal{P}$.

Remarque : De nombreux candidats proposent comme contre-exemple l'automate $\mathcal{X}$ de la Question 1.1. Pour celui-ci, on a bien $\forall p,\ \mathcal{X}(\mathcal{P}_p) \neq \mathcal{P}_p$. Par exemple pour une configuration de

[*]LIM - CMI, Université de Provence, 39 rue Joliot-Curie, 13453 Marseille Cedex 13. E-mail: Bruno.Durand@cmi.univ-mrs.fr

[†]LIAFA, CNRS - Université Paris 7, Case 7014, 2 place Jussieu, 75251 Paris Cedex 05. E-mail: mairesse@liafa.jussieu.fr

[‡]Élève de l'École Normale Supérieure, 45 rue d'Ulm, 75230 Paris Cedex O5. E-mail: Yves.Verhoeven@ens.fr

période $0^n1, n \in \mathbb{N}$, les antécédents ont pour période $0^{n+1}1^{n+1}$. En revanche on a $\mathcal{X}(\mathcal{P}) = \mathcal{P}$. Ainsi, l'automate $\mathcal{X}$ ne permet pas de répondre complètement à la question posée.

Solution proposée par Y.V.

Soient $p = (p_1, \dots, p_n) \in (\mathbb{N} \setminus \{0\})^n$, $C \in \mathcal{P}_p$ et $u \in \mathbb{Z}^n, (\lambda_1, \dots, \lambda_n) \in \mathbb{Z}^n$. Pour montrer que $\mathcal{A}(\mathcal{P}_p) \subset \mathcal{P}_p$, il suffit de montrer que $\mathcal{A}(C)(u + (\lambda_1 p_1, \dots, \lambda_n p_n)) = \mathcal{A}(C)(u)$. Remarquons tout d'abord que $C(u + (\lambda_1 p_1, \dots, \lambda_n p_n)) = C(u)$. Grâce à ceci, un simple calcul nous donne alors

$$\begin{aligned}
\mathcal{A}(C)(u + (\lambda_1 p_1, \dots, \lambda_n p_n)) &= \delta(C(u + (\lambda_1 p_1, \dots, \lambda_n p_n) + v_1), \dots, \\
&\qquad\qquad C(u + (\lambda_1 p_1, \dots, \lambda_n p_n) + v_n)) \\
&= \delta(C(u + v_1), \dots, C(u + v_n)) \\
&= \mathcal{A}(C)(u) .
\end{aligned}$$

Mais $\mathcal{P} = \bigcup_{p \in (\mathbb{N}\setminus\{0\})^n} \mathcal{P}_p$. Il vient alors immédiatement

$$\mathcal{A}(\mathcal{P}) \ = \ \mathcal{A}(\bigcup_p \mathcal{P}_p) \ = \ \bigcup_p \mathcal{A}(\mathcal{P}_p) \ \subset \ \bigcup_p \mathcal{P}_p \ = \ \mathcal{P} .$$

C'est-à-dire $\mathcal{A}(\mathcal{P}) \subset \mathcal{P}$.

Montrons que les inclusions ne sont pas nécessairement strictes. Donnons nous un ensemble d'états S possédant $n \geq 2$ éléments distincts. Tout automate cellulaire dont la fonction locale ne prend que $n-1$ valeurs convient, car la configuration constante égale à la n-ième valeur appartient à $\mathcal{P}_p$, pour tout p, mais ne peut être atteinte.

Question 1.3

Soit c une configuration de $\mathcal{F}_q$. Par définition, l'ensemble $V = \{u \in \mathbb{Z}^n \mid c(u) \neq q\}$ est fini. Notons, pour v dans $\mathbb{Z}^n$, $V - v = \{u - v, u \in V\}$. L'ensemble $(V - v_1) \cup \dots \cup (V - v_l)$ est fini. Pour tout $u \notin (V - v_1) \cup \dots \cup (V - v_l)$, on a

$$\mathcal{A}(c)(u) = \delta(c(u + v_1), \dots, c(u + v_l)) = \delta(q, \dots, q) .$$

On en déduit que $\mathcal{A}(\mathcal{F}_q) \subset \mathcal{F}_{\delta(q,\dots,q)}$.

Question 1.4

Démontrons le résultat par récurrence. Supposons que $\mathcal{A}^k$ est un automate cellulaire de la forme $(n, S, V^{(k)}, \delta^{(k)})$. Pour alléger les notations, on écrit pour c dans $S^{\mathbb{Z}^n}$ et u dans $\mathbb{Z}^n$, $\mathcal{A}^k(c)(u) = \delta^{(k)}(c(u + v), v \in V^{(k)})$. On a

$$\begin{aligned}
\mathcal{A}^{k+1}(c)(u) &= \mathcal{A}[\mathcal{A}^k(c)](u) \\
&= \delta(\mathcal{A}^k(c)(u + v_1), \dots, \mathcal{A}^k(c)(u + v_l)) \\
&= \delta(\delta^{(k)}(c(u + v_1 + v), v \in V^{(k)}), \dots, \delta^{(k)}(c(u + v_l + v), v \in V^{(k)})) . \qquad (1)
\end{aligned}$$

Si $V^{(k)} = \{v_1^{(k)}, \dots, v_{l_k}^{(k)}\}$, on définit le $(l \times l_k)$-uplet

$$V^{(k+1)} = \{v_1 + v_1^{(k)}, \dots, v_1 + v_{l_k}^{(k)}, v_2 + v_1^{(k)}, \dots, v_2 + v_{l_k}^{(k)}, \dots, v_l + v_1^{(k)}, \dots, v_l + v_{l_k}^{(k)}\} .$$

On obtient que $\mathcal{A}^{k+1}$ est un automate cellulaire de la forme $(n, S, V^{(k+1)}, \delta^{(k+1)})$, où $\delta^{(k+1)}$ est défini par l'équation de récurrence (1). Il est difficile de décrire la fonction locale $\delta^{(k+1)}$ de façon plus compacte.

Soit q_0 un état quelconque. Définissons la suite $(q_i)_{i\in\mathbb{N}}$ par récurrence par $q_{i+1} = \delta(q_i, \ldots, q_i)$. Cette suite ne pouvant prendre qu'un nombre fini de valeurs, elle est ultimement périodique. Il existe donc i_0 et k tels que $q_{i_0+k} = q_{i_0}$. On en déduit que $\mathcal{A}^k(\mathcal{F}_{q_0}) \subset \mathcal{F}_{q_0}$.

2 Propriétés générales

Commençons par mentionner deux erreurs graves retrouvées dans plusieurs copies. Certains candidats pensent que 'δ injectif (resp. surjectif)' implique '$\mathcal{A}$ injectif (resp. surjectif)'. Le lien entre la fonction locale et la fonction globale d'un automate cellulaire est évidemment beaucoup plus subtile. D'autres candidats traitent $\mathcal{A}_{|\mathcal{P}}$ ou $\mathcal{A}_{|\mathcal{F}_q}$ comme s'ils étaient eux-mêmes des automates cellulaires (leur appliquant par exemple le Théorème de Moore-Myhill). Ces deux erreurs ont comme caractéristique commune de rendre l'épreuve totalement inintelligible aux candidats les ayant effectuées.

Les preuves des résultats de cette partie peuvent aussi s'obtenir par des arguments topologiques (voir [1]), sauf en ce qui concerne le théorème de Moore et Myhill dont on ne connaît que des preuves combinatoires.

Question 2.1

Application directe du Théorème de Moore et Myhill.

Question 2.2

Pour u dans $\mathbb{Z}^n$, on note $\|u\| = \max_i |u_i|$ et on définit $\mathcal{B}_k = \{u \in \mathbb{Z}^n, \|u\| \le k\}$ (cette dernière notation est introduite dans l'énoncé après la Question 2.3). Soit $U = \{u_1, \ldots, u_k\}$ un k-uplet de $\mathbb{Z}^n$, on note $\|U\| = \max_i \|u_i\|$.

Soit c, d, deux configurations de $\mathcal{F}_q$ telles que $\mathcal{A}(c) = \mathcal{A}(d)$. Par définition, $\exists r \in \mathbb{N},\ \forall v \notin \mathcal{B}_r,\ c(v) = d(v) = q$. On définit $\tilde{c}$ et $\tilde{d}$ de la façon suivante

$$\begin{aligned} &\forall u \in \mathcal{B}_{r+2\|V\|}, \qquad \tilde{c}(u) = c(u),\ \tilde{d}(u) = d(u) \\ &\forall u \notin \mathcal{B}_{r+2\|V\|}, \qquad \tilde{c}(u) = \tilde{d}(u) = q_0 \end{aligned}$$

Par construction, on a $\tilde{c}, \tilde{d} \in \mathcal{F}_{q_0}$. De plus, pour tout $u \notin \mathcal{B}_{r+\|V\|}$ et $v \in V$, on a $\tilde{c}(u+v) = \tilde{d}(u+v)$ (égal q ou q_0 suivant les valeurs de v et u). On en déduit que $\mathcal{A}(\tilde{c})(u) = \mathcal{A}(\tilde{d})(u)$ pour $u \notin \mathcal{B}_{r+\|V\|}$. Si $u \in \mathcal{B}_{r+\|V\|}$ et $v \in V$, on a $\tilde{c}(u+v) = c(u+v)$ et $\tilde{d}(u+v) = d(u+v)$. On en déduit que $\mathcal{A}(\tilde{c})(u) = \mathcal{A}(c)(u)$ et $\mathcal{A}(\tilde{d})(u) = \mathcal{A}(d)(u)$. Or, par hypothèse $\mathcal{A}(c) = \mathcal{A}(d)$ et on conclut que $\mathcal{A}(\tilde{c})(u) = \mathcal{A}(\tilde{d})(u)$ pour $u \in \mathcal{B}_{r+\|V\|}$.

Ainsi on a prouvé que $\mathcal{A}(\tilde{c}) = \mathcal{A}(\tilde{d})$. Par injectivité de $\mathcal{A}$ sur $\mathcal{F}_{q_0}$, on en déduit que $\tilde{c} = \tilde{d}$ ce qui entraîne que $c = d$.

Remarque : Noter qu'il est nécessaire de définir $\tilde{c}$ et $\tilde{d}$ par rapport à la boule de rayon $r + \mathbf{2}\|V\|$. Une définition analogue par rapport à la boule de rayon $r + \|V\|$ ne permet pas de conclure.

Dans le même ordre d'idées, la réponse suivante, proposée par de nombreux candidats, est clairement incorrecte.

Soit β_{q,q_0} l'application de $S^{\mathbb{Z}^n}$ dans $S^{\mathbb{Z}^n}$ définie par

$$\beta_{q,q_0}(c)(u) = \begin{cases} q_0 & \textit{si } c(u) = q \\ c(u) & \textit{sinon.} \end{cases}$$

L'application $(\mathcal{A} \circ \beta_{q,q_0})$ est injective sur $\mathcal{F}_q$ et donc $\mathcal{A}$ est injective sur $\mathcal{F}_q$.

Question 2.3

Supposons que $\mathcal{A}_{|\mathcal{P}}$ est injectif. Soit $p = (p_1, \ldots, p_n) \in (\mathbb{N} \setminus \{0\})^n$. De par la Question 1.2, on a $\mathcal{A}(\mathcal{P}_p) \subset \mathcal{P}_p$. D'autre part $\mathcal{P}_p$ est fini (une configuration de $\mathcal{P}_p$ est entièrement déterminée par sa valeur sur $[0, p_1 - 1] \times \cdots \times [0, p_n - 1]$). Comme $\mathcal{A}_{|\mathcal{P}_p}$ est injectif, on en déduit que $\mathcal{A}(\mathcal{P}_p) = \mathcal{P}_p$ et donc que $\mathcal{A}_{|\mathcal{P}_p}$ est bijectif. Comme $\mathcal{P} = \cup_p \mathcal{P}_p$, il vient directement que $\mathcal{A}_{|\mathcal{P}}$ est surjectif. Pour montrer l'injectivité de $\mathcal{A}_{|\mathcal{P}}$, il suffit de remarquer que deux configurations périodiques admettent une période commune et d'utiliser l'injectivité des $\mathcal{A}_{|\mathcal{P}_p}$.

Question 2.4

a) L'ensemble infini $\{d_k(x_0), k \in \mathbb{N}\}$ prend un nombre fini de valeurs S, d'où le résultat.

b) Même raisonnement à partir de $\{d_k(x_1), k \in I_0\}$.

c) Soit ϕ une application bijective de $\mathbb{N}$ dans $\mathbb{Z}^n$ et soit $x_i = \phi(i)$. On définit récursivement les sous-ensembles $I_i \subset \mathbb{N}$ et les valeurs $d(x_i) \in S$ vérifiant pour tout i :

$$I_i \subset I_{i-1}, \ |I_i| = +\infty, \ \{d_k(x_i), k \in I_i\} = \{d(x_i)\} .$$

Montrons que $\mathcal{A}(d) = c$. On choisit $u \in \mathbb{Z}^n$ et on note $i_1, \ldots, i_l$, les entiers tels que $u + v_1 = x_{i_1}, \ldots, u + v_l = x_{i_l}$. Posons $J = I_{i_1} \cap \cdots \cap I_{i_l}$. On a

$$\begin{aligned} \mathcal{A}(d)(u) = \delta(d(u+v_1), \ldots, d(u+v_l)) &= \delta(d(x_{i_1}), \ldots, d(x_{i_l})) \\ &= \delta(d_k(x_{i_1}), \ldots, d_k(x_{i_l})), \quad \forall k \in J \\ &= \mathcal{A}(d_k)(u), \quad \forall k \in J \\ &= c_k(u), \quad \forall k \in J . \end{aligned}$$

Comme J est de cardinal infini, on en déduit l'existence de $k_0 \in J$ tel que $u \in \mathcal{B}_{k_0}$. On en conclut que $\mathcal{A}(d)(u) = c_{k_0}(u) = c(u)$.

Question 2.5

a) Soit q' tel que $\mathcal{A}(\mathcal{F}_q) = \mathcal{F}_{q'}$ et soit c une configuration de $S^{\mathbb{Z}^n}$. On définit les configurations q'-finies $c_k, k \in \mathbb{N}$, de la façon suivante

$$c_k(u) = \begin{cases} c(u) & \text{si } u \in \mathcal{B}_k \\ q' & \text{sinon.} \end{cases}$$

Par surjectivité de $\mathcal{A}_{|\mathcal{F}_q}$, il existe $d_k \in \mathcal{F}_q$ tel que $\mathcal{A}(d_k) = c_k$, pour tout k. Par application de la Question 2.4, on en déduit l'existence de d tel que $\mathcal{A}(d) = c$.

b) Le principe est le même qu'en a). On choisit une configuration c et on définit c_k comme l'unique configuration totalement périodique de périodes $(2k+1, 0, \ldots, 0), \ldots, (0, \ldots, 0, 2k+1)$ et vérifiant $c_k|_{\mathcal{B}_k} = c|_{\mathcal{B}_k}$.

Question 2.6

On rappelle que pour $v \in \mathbb{Z}^n$, l'opérateur de translation $\sigma(v)$ a été défini dans l'énoncé, juste avant la Question 1.1.

a) Soit $U = \{u \in \mathbb{Z}^n \mid b(u) \neq q\}$ et soit $r \in \mathbb{N}$ tel que $\{u \in \mathbb{Z}^n \mid a(u) \neq q'\} \subset \mathcal{B}_r$. Le cardinal de $(U \cap (\mathbb{Z}^n \setminus \mathcal{B}_k))$ est infini pour tout k. On en déduit qu'il existe une suite $\{u_i, i \in \mathbb{N}\}$ de points de $\mathbb{Z}^n$ tels que :

$$u_i \in U,\ \|u_i\| > r + i\,.$$

On pose $a_k = \sigma_{-u_k}(a)$ et $b_k = \sigma_{-u_k}(b)$. On vérifie immédiatement que l'on a $\mathcal{A}(b_k) = a_k$. De plus, par construction, on a $a_k|_{\mathcal{B}_k} = \overrightarrow{q}'|_{\mathcal{B}_k}$. Comme en Question 2.4, on construit par extraction à partir des b_k une configuration d telle que $\mathcal{A}(d) = \overrightarrow{q}'$. Toujours par construction, on a $b_k((0, \ldots, 0)) = b(u_k) \neq q$, pour tout k. On en déduit que $d((0, \ldots, 0)) \neq q$. On conclut que $\mathcal{A}(d) = \mathcal{A}(\overrightarrow{q}) = \overrightarrow{q}'$ avec $d \neq \overrightarrow{q}$.

b) Si $\mathcal{A}$ est bijectif alors $\mathcal{A}$ est injectif et $\mathcal{A}\big|_{\mathcal{F}_q}$ est injectif pour tout q. Supposons qu'il existe q tel que $\mathcal{A}\big|_{\mathcal{F}_q}$ ne soit pas surjectif. Posons $q' = \delta(q, \ldots, q)$. Comme $\mathcal{A}$ est surjectif et que $\mathcal{A}\big|_{\mathcal{F}_q}$ ne l'est pas, il existe des configurations b et c telles que :

$$c \in \mathcal{F}_{q'},\ b \notin \mathcal{F}_q,\ \mathcal{A}(b) = c\,.$$

D'après la Question 2.6.a), la configuration $\overrightarrow{q}'$ admet au moins deux antécédents, ce qui contredit l'injectivité de $\mathcal{A}$. On en déduit que $\mathcal{A}\big|_{\mathcal{F}_q}$ est surjectif pour tout q.

Solution proposée par Y.V.

Le résultat de la Question 2.6.b) peut se traiter sans passer par la Question 2.6.a) si l'on se permet d'utiliser le théorème 2 de la partie 4 (dont la preuve ne repose pas sur le résultat que l'on veut démontrer). L'automate cellulaire $\mathcal{A}$ étant bijectif, d'après le théorème 2, $\mathcal{A}^{-1}$ est la fonction globale d'un automate cellulaire (n, S, V', δ'). Mais comme $\mathcal{A}(\overrightarrow{q}) = \overrightarrow{q}'$, on a $\mathcal{A}^{-1}(\overrightarrow{q}') = \overrightarrow{q}$, et par conséquent $\delta'(q', \ldots, q') = q$. Ainsi $\mathcal{A}^{-1}(\mathcal{F}_{q'}) \subset \mathcal{F}_q$, et l'image d'une configuration $c' \in \mathcal{F}_{q'}$ par $\mathcal{A}^{-1}$ est un antécédent q-fini de c' pour $\mathcal{A}$. Donc c' admet un antécédent q-fini pour $\mathcal{A}$, ce qui entraîne la surjectivité de $\mathcal{A}|_{\mathcal{F}_q}$. Enfin $\mathcal{A}|_{\mathcal{F}_q}$ est clairement injectif.

Question 2.7

Il reste à démontrer l'équivalence : ' $\exists q, \mathcal{A}\big|_{\mathcal{F}_q}$ bijectif $\iff \exists q, \mathcal{A}\big|_{\mathcal{F}_q}$ surjectif'. En suivant les implications déjà démontrées de la figure 1, on obtient

$$\exists q_0, \mathcal{A}\big|_{\mathcal{F}_{q_0}} \text{ surjectif} \implies \forall q, \mathcal{A}\big|_{\mathcal{F}_q} \text{ injectif} \implies \mathcal{A}\big|_{\mathcal{F}_{q_0}} \text{ injectif} \implies \mathcal{A}\big|_{\mathcal{F}_{q_0}} \text{ bijectif}\,.$$

$$\boxed{\mathcal{A} \text{ bijectif} \overset{\textbf{Q. 2.1}}{\Longleftrightarrow} \mathcal{A} \text{ injectif}}$$

$$\textbf{Q. 2.6 b)} \Downarrow \qquad \Downarrow \textbf{Trivial}$$

$$\boxed{\begin{array}{c} \exists q,\ \mathcal{A}\big|_{\mathcal{F}_q} \text{ bijectif} \\ \Updownarrow \\ \exists q,\ \mathcal{A}\big|_{\mathcal{F}_q} \text{ surjectif} \end{array}} \qquad \begin{array}{c} \boxed{\mathcal{A}|_{\mathcal{P}} \text{ bijectif} \overset{\textbf{Q. 2.3}}{\Longleftrightarrow} \mathcal{A}|_{\mathcal{P}} \text{ injectif}} \\ \Downarrow \textbf{Trivial} \\ \boxed{\mathcal{A}|_{\mathcal{P}} \text{ surjectif}} \end{array}$$

$$\textbf{Q. 2.5 a)} \Downarrow \qquad \Downarrow \textbf{Q. 2.5 b)}$$

$$\boxed{\forall q,\ \mathcal{A}\big|_{\mathcal{F}_q} \text{ injectif} \overset{\textbf{Q. 2.2}}{\Longleftrightarrow} \exists q,\ \mathcal{A}\big|_{\mathcal{F}_q} \text{ injectif} \overset{\textbf{Moore-Myhill}}{\Longleftrightarrow} \mathcal{A} \text{ surjectif}}$$

FIG. 1: Diagramme des implications.

3 Algorithmes de décision en dimension 1

Question 3.1

On a $\mathcal{X} = (\{0,1\}, 2, \delta_{\mathcal{X}})$ et $\mathcal{G}(\mathcal{X}) = (\{0,1\}, \{0,1\}^2, \{0,1\}, \delta_{\mathcal{X}})$. Le graphe $\mathcal{G}(\mathcal{X})$ est représenté en figure 2.

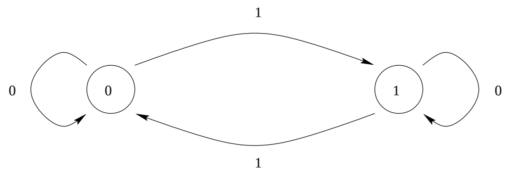

FIG. 2: Le graphe $\mathcal{G}(\mathcal{X})$.

Question 3.2

Pour $x \in S^{r-1}$, on note $l(x)$ la première lettre de x. Étant donné un chemin bi-infini $(x_i)_{i\in\mathbb{Z}}$, on définit $c \in S^{\mathbb{Z}}$ par $c(i) = l(x_i)$. Étant donné une configuration $c \in S^{\mathbb{Z}}$, on définit un chemin bi-infini $(x_i)_{i\in\mathbb{Z}}$ par : $x_i = c(i)c(i+1)\cdots c(i+r-2)$.

Soit c une configuration et $(x_i)_{i\in\mathbb{Z}}$ le chemin bi-infini associé. Si c est totalement périodique de période k, on a $x_{i+k} = x_i$, $\forall i$. Si c est q-fini, il existe R tel que $x_i = q^{r-1}$ pour $|i| \geq R$. Enfin, on a $\mathcal{A}(c)(i) = e((x_i, x_{i+1}))$.

Question 3.3

Soit x et y dans N avec $x = x_1 \cdots x_{r-1}, x_i \in S$, et $y = y_1 \cdots y_{r-1}, y_i \in S$. Le chemin

$$(x_1 \cdots x_{r-1}\ ,\ x_2 \cdots x_{r-1}y_1\ ,\ x_3 \cdots x_{r-1}y_1y_2\ , \ldots ,\ x_{r-1}y_1y_2 \cdots y_{r-2}\ ,\ y_1 \cdots y_{r-1})$$

est un chemin de $\mathcal{G}$.

Il est également possible de répondre à la question sans construire explicitement le chemin. Pour tout x et y dans S^{r-1}, il existe une configuration c telle que :

$\exists i < j,\ c(i)\cdots c(i+r-2) = x,\ c(j)\cdots c(j+r-2) = y$. On en déduit qu'il existe un chemin dans $\mathcal{G}$ de x vers y.

Question 3.4

L'implication '$\mathcal{A}|_{\mathcal{P}}$ surjectif $\Longrightarrow$ $\mathcal{A}$ surjectif' est vraie en toute dimension, voir Question 2.5.b.

Supposons $\mathcal{A}$ surjectif. Soit $c \in \mathcal{P}$, de période p, et soit $b \in S^{\mathbb{Z}}$ tel que $\mathcal{A}(b) = c$. Soit $(x_i)_{i\in\mathbb{Z}}$ le chemin bi-infini associé à b. On considère la suite de nœuds $(x_{ip})_{i\in\mathbb{Z}}$. Comme l'ensemble N est fini, il existe i_1 et i_2, $i_1 < i_2$, tels que $x_{i_1p} = x_{i_2p}$. Comme on a $e(x_i, x_{i+1}) = e(x_{i+p}, x_{i+1+p})$ pour tout $i \in \mathbb{Z}$, on en d'eduit que

$$e(x_{i_1p}, x_{i_1p+1})e(x_{i_1p+1}, x_{i_1p+2})\cdots e(x_{i_2p-1}, x_{i_2p}) = [c(0)\cdots c(p-1)]^{i_2-i_1} .$$

On définit $(y_i)_{i\in\mathbb{Z}}$ comme le chemin périodique de période $(i_2 - i_1)p$ vérifiant $y_i = x_i$ pour $i = i_1p, \ldots, i_2p-1$. On déduit de ce qui précède que la suite des étiquettes de $(y_i)_{i\in\mathbb{Z}}$ est c. Si on note a la configuration associée à $(y_i)_{i\in\mathbb{Z}}$, on obtient $\mathcal{A}(a) = c$ et $a \in \mathcal{P}_{(i_2-i_1)p}$. On conclut que $\mathcal{A}|_{\mathcal{P}}$ est surjectif.

Remarque : Certains candidats énoncent et utilisent la propriété suivante :

Si $\mathcal{A}$ est surjectif, alors pour tout nœud x du graphe $\mathcal{G}(\mathcal{A})$ et pour toute étiquette q, il existe un arc étiqueté par q partant de x.

Cette propriété est fausse. Pour le voir, il suffit de considérer l'automate 'identité' défini par $(\{0,1\}, 2, \delta)$ avec $\delta(x, y) = x$. Une autre erreur commise par certains candidats consiste à construire a par répétition périodique d'un circuit du chemin bi-infini associé à b (les notations sont celles du corrigé ci-dessus).

Solution proposée par Y.V.

Il suffit, d'après la Question 2.5, de montrer que si $\mathcal{A}$ est surjectif, alors $\mathcal{A}|_{\mathcal{P}}$ est surjectif. Supposons donc $\mathcal{A}$ surjectif. Soit c une configuration p-périodique. L'automate $\mathcal{A}$ étant surjectif, c admet un antécédent c' par $\mathcal{A}$; l'ensemble S étant fini, il existe $k_1 < k_2$ tels que les fonctions $x \mapsto c'(x + k_1p)$ et $x \mapsto c'(x + k_2p)$ soient égales sur $\{0, \ldots, r-1\}$. Soit alors c'' une configuration $(k_2 - k_1)p$-périodique définie sur $x \in \{0, \ldots, (k_2 - k_1)p\}$ par $c''(x) = c'(x + k_1p)$. Il est alors aisé de vérifier que par cette construction on a $\mathcal{A}(c'') = c$. De plus, c'' étant périodique, on a obtenu le résultat cherché : toute configuration périodique admet un antécédent périodique par $\mathcal{A}$.

Question 3.5

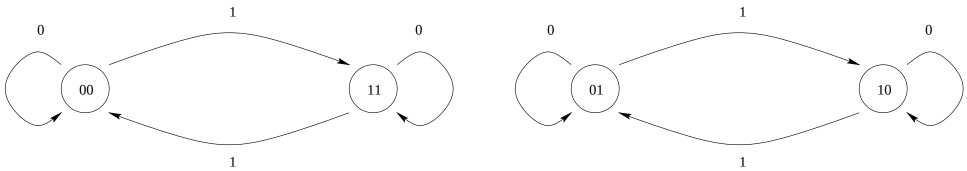

FIG. 3: Le graphe produit associé à $\mathcal{X}$.

Question 3.6

Pour x, y dans N', la notation $x \to y$ signifie qu'il existe un chemin (de longueur non-nulle) de x vers y dans $\mathcal{H}$. Un nœud v appartient à $\mathcal{D}$ si et seulement si il existe x, y, tels que $x \to x \to v \to y \to y$. Une procédure possible pour déterminer $\mathcal{D}$ consiste donc à 'élaguer' de façon itérative le graphe en éliminant tous les nœuds sans successeur ou sans prédécesseur. Il est très facile de trouver un algorithme effectuant cette tâche. Ce qui est plus difficile c'est d'en trouver un ayant la bonne complexité. Vue la borne donnée dans l'énoncé, il s'agit d'effectuer un nombre d'opérations borné par le nombre d'arcs de $\mathcal{H}$, à une constante mutiplicative près. En effet, comme il y a au plus s^2 arcs partant de chaque nœud dans (N', A'), on obtient $|N'| = s^{2(r-1)}$ et $|A'| \leq s^{2r}$.

Il suffit d'écrire une procédure pour sélectionner les nœuds à partir desquels on peut construire un chemin infini. Il suffira d'effectuer une seconde fois cet algorithme en inversant le sens des arcs pour obtenir les nœuds pouvant appartenir à un chemin bi-infini.

Algorithme : Prendre tous les nœuds et arcs de $\mathcal{H}$. Chercher les nœuds sans successeur. Les supprimer ainsi que les arcs pointant vers ces nœuds. Recommencer ce procédé tant qu'il reste des nœuds sans successeur à supprimer.

Preuve : Cet algorithme effectue bien la tâche qu'on attend de lui. Si tous les successeurs d'un nœud ont au moins un successeur, et que ces successeurs ont eux-mêmes au moins un successeur, etc, on construit facilement un circuit et donc un chemin infini à partir de ce nœud. Comme la condition d'arrêt de l'algorithme est qu'il n'existe plus de nœuds sans successeurs dans le graphe, alors si l'algorithme s'arrête, dans le graphe obtenu tous les nœuds sont sur des chemins infinis. Il faut encore montrer qu'on n'enlève pas trop de nœuds. Il est évident que les chemins infinis ne sont pas affectés par les suppressions d'arcs et de nœuds effectuées par l'algorithme. Il suffit maintenant de montrer que l'algorithme s'arrête effectivement. À chaque itération de la boucle le nombre de nœuds décroît strictement. Donc l'algorithme s'arrête au bout d'un temps fini.

Complexité : Pour manipuler facilement le graphe, il est pratique de tenir à jour pour chaque nœud la liste de ses successeurs et prédécesseurs directs. Cette opération a un coût en $\mathcal{O}(s^{2r})$. En effet pour chaque arc (u, v) supprimé, il suffit d'actualiser la liste des successeurs de v (le nœud u a lui été supprimé).

L'opération qui consiste à chercher les nœuds sans successeurs est de complexité proportionnelle au nombre de nœuds, soit $\mathcal{O}(s^{2(r-1)})$. Le nombre de nœuds décroissant strictement à chaque itération de la boucle, le nombre d'itérations est borné par $\mathcal{O}(s^{2(r-1)})$. La complexité totale de l'algorithme est donc de $\mathcal{O}(s^{4(r-1)})$. Il existe plusieurs solutions pour se rammener à la complexié en $\mathcal{O}(s^{2r})$ requise par l'énoncé. La plus élégante consiste à améliorer l'algorithme en n'effectuant la recherche des nœuds sans successeur qu'une seule fois au début. Ensuite les seuls nœuds susceptibles de devenir sans successeur sont les prédécesseurs des nœuds supprimés à l'itération précédente. Avec cette procédure, un nœud peut être examiné plusieurs fois. Cependant, si on regarde l'exécution de l'algorithme dans sa globalité, on est sûr que sa complexité sera bien en $\mathcal{O}(s^{2r})$. En effet, pour chaque arc supprimé le nombre d'opérations engendrées est borné par une constante.

Pseudo-code : Étant donné un graphe (N, A), on note $\alpha^-(u)$ et $\alpha^+(u)$ l'ensemble des arcs prédécesseurs et successeurs directs de u ($(v, u) \in \alpha^-(u)$ si $(v, u) \in A$; $(u, v) \in \alpha^+(u)$ si

$(u,v) \in A$). On note $\nu^-(u)$ et $\nu^+(u)$ l'ensemble des nœuds prédécesseurs et successeurs directs de u ($v \in \nu^-(u)$ si $(v,u) \in A$; $v \in \nu^+(u)$ si $(u,v) \in A$).

Dans le pseudo-code ci-dessous, les ensembles $\alpha^-(v), \alpha^+(v), \nu^-(v)$ et $\nu^+(v)$ sont toujours définis par rapport au graphe courant (Noeud,Arcs). On ne détaille pas dans le pseudo-code les opérations permettant de tenir à jour ces ensembles.

```
Entrée : (N', A')
  1. Noeuds ← N', Arcs ← A', Atester ← N'
  2. répéter jusqu'à ce que Aeliminer = ∅
     %% Dans une boucle 'répéter jusqu'à ce que' le test est effectué en fin de boucle
     - Aeliminer ← ∅
     - pour tout v ∈ Atester faire
       - si ν⁺(v) ∩ Noeuds = ∅ alors Aeliminer ← Aeliminer ∪ {v}
       - %% La variable Aeliminer contient les nœuds sans successeur
     - Atester ← ∅
     - pour tout v ∈ Aeliminer faire
       - Noeuds ← Noeuds\{v}, Arcs ← Arcs\α⁻(v)
       - Atester ← Atester ∪ (ν⁻(v) ∩ Noeuds)
     - %% La variable Atester contient les prédécesseurs des nœuds sans successeur
  3. Atester ← Noeuds, Aeliminer ← ∅
  4. Répétition de l'instruction 2 en inversant les rôles de ν⁺(v) et de ν⁻(v) d'une part,
     et en remplaçant α⁻(v) par α⁺(v) d'autre part
Sortie : (Noeuds,Arcs)
```

Remarque : Avant de répondre aux Questions 3.7-3.10, faisons un commentaire sur la notion de forte connexité. La définition proposée dans l'énoncé a deux interprétations possibles (les auteurs de ce sujet ne s'en sont bien sûr rendu compte que trop tard !). En effet, il n'est pas précisé si un graphe contenant un seul nœud et aucun arc est fortement connexe ou non (autrement dit les chemins de longueur 0 sont-ils admis ou non). Les deux conventions sont possibles. Dans le cas présent, cette imprécision n'était pas susceptible de gêner les candidats. En effet les énoncés des Questions 3.7-3.10 sont valides aussi bien avec une convention qu'avec l'autre. Dans le corrigé proposé ci-dessous, on considère que le graphe trivial est fortement connexe.

Question 3.7

Si $\mathcal{A}$ est injectif, un chemin bi-infini de $\mathcal{D}$ ne peut contenir que des nœuds de la forme $(x,x), x \in N$. On en déduit que $\mathcal{D} = \overline{G} = \tilde{G}$.
Si $\overline{G} = \tilde{G}$ est l'unique cfcm de $\mathcal{D}$, on obtient que $\mathcal{D} = \overline{G} = \tilde{G}$ (car tout nœud appartient à une cfcm). On en déduit que $\mathcal{A}$ est injectif.

Question 3.8

Soit q tel que $\mathcal{A}\big|_{\mathcal{F}_q}$ soit injectif. Soit C la cfcm de $\mathcal{D}$ contenant (q^{r-1}, q^{r-1}). Par injectivité, C ne contient que des nœuds de la forme $(x,x), x \in N$, c'est-à-dire que l'on a $C \subset \tilde{\mathcal{G}} \subset \overline{\mathcal{G}}$. Comme C et $\overline{\mathcal{G}}$ sont des cfcm, on en déduit que $C = \tilde{\mathcal{G}} = \overline{\mathcal{G}}$.
Supposons que $\tilde{\mathcal{G}} = \overline{\mathcal{G}}$. Considérons a, b dans $\mathcal{F}_q$ tels que $\mathcal{A}(a) = \mathcal{A}(b)$. Soit $(x_i)_{i \in \mathbb{Z}}$ le chemin bi-infini de $\mathcal{D}$ correspondant à la configuration (a,b). On a $x_i = (q^{r-1}, q^{r-1})$ sauf pour un

nombre fini d'indices i. En particulier, cela implique que le chemin $(x_i)_{i\in\mathbb{Z}}$ est inclus dans la cfcm de (q^{r-1}, q^{r-1}), c'est-à-dire dans $\overline{\mathcal{G}} = \tilde{\mathcal{G}}$. On en déduit que $a = b$ et donc que $\mathcal{A}\big|_{\mathcal{F}_q}$ est injective.

Question 3.9

Pour écrire l'algorithme demandé, on peut commencer par construire $\mathcal{D}$. Ceci nécéssite l'utilisation de l'algorithme de la Question 3.6, dont la complexité n'est pas supérieure à celle recherchée. On utilise ensuite les caractérisations suivantes (voir Questions 3.7 et 3.8, et le Théorème de Moore-Myhill).

- L'automate cellulaire est surjectif si et seulement si $\tilde{\mathcal{G}} = \overline{\mathcal{G}}$.
- L'automate cellulaire est bijectif si et seulement si $\tilde{\mathcal{G}} = \mathcal{D}$.

On en déduit l'algorithme suivant pour tester la bijectivité.

Algorithme pour la bijectivité : Tester tous les nœuds de $\mathcal{D}$. Si l'un d'eux n'est pas dans $\tilde{\mathcal{G}}$, alors répondre "non-bijectif", sinon répondre "bijectif".

Preuve et complexité : La preuve est immédiate et la complexité de cet algorithme est en $\mathcal{O}(s^{2(r-1)})$ (le nombre de nœuds de $\mathcal{D}$ est majoré par $s^{2(r-1)}$).

Reste à tester la surjectivité. Pour cela, colorions en bleu (resp. en rouge) chaque nœud de $\mathcal{D}$ qui mène à un nœud de $\tilde{\mathcal{G}}$ (resp. qui en vient). Par définition, les nœuds de $\bar{\mathcal{G}}$ sont les nœuds qui sont à la fois rouges et bleus. L'automate est donc surjectif si et seulement si seuls les nœuds de $\tilde{\mathcal{G}}$ sont coloriés à la fois en rouge et en bleu.

On décrit ci-dessous uniquement la procédure de coloration en rouge. Il suffit de renverser le sens des arcs de $\mathcal{D}$ et d'appliquer la même procédure pour obtenir la coloration en bleu. En entrée de l'algorithme, on dispose des graphes $\mathcal{D}$ et $\tilde{\mathcal{G}}$. Il est pratique de considérer que l'on dispose aussi des ensembles d'arcs prédecesseurs et successeurs directs dans $\mathcal{D}$ des nœuds de $\mathcal{D}$. Pour un nœud u de $\mathcal{D}$, on note ces ensembles $\alpha^-(u)$ et $\alpha^+(u)$ respectivement (contrairement au cas de l'algorithme de la Question 3.6, ces ensembles sont fixés et ne sont pas réactualisés en cours de procédure).

Algorithme de coloration : Initialiser un ensemble de nœuds ν et un ensemble d'arcs α à ceux de $\tilde{\mathcal{G}}$. Pour chaque nœud u de ν, déterminer l'ensemble des arcs de $\alpha^+(u)$ dont l'extrémité finale n'appartient pas à ν. Ajouter ces arcs et les nœuds auxquels ils mènent à α et ν respectivement. Recommençer la procédure tant qu'on trouve des arcs sortant de ν. L'ensemble ν est alors l'ensemble des nœuds rouges.

Preuve et complexité : La justification de l'algorithme est contenue dans les remarques précédant son exposition. Cette procédure termine puisque le nombre de nœuds de ν croît strictement à chaque étape et est borné par le nombre de nœuds de $\mathcal{D}$. Pour obtenir la complexité souhaitée, on utilise la même technique qu'en Question 3.6. On n'effectue la procédure décrite dans l'algorithme que lors de la première itération. Ensuite on ne teste que les nœuds ajoutés à ν lors de l'itération précédente. Le nombre d'opérations est alors borné par le nombre d'arcs de $\mathcal{D}$ (à une constante multiplicative près) et la compléxité est donc en $\mathcal{O}(s^{2r})$. L'algorithme de coloration proposé procède par une recherche en *largeur*, il est également possible de procéder par une recherche en *profondeur*, voir la solution proposée par Y. Verhoeven.

Pseudo-code : Non-détaillé. On peut s'inspirer de celui proposé en Question 3.6.

Remarque : De façon générale, il est courant en algorithmique que la démarche consistant à faire une preuve soit analogue à celle consistant à construire un algorithme. Ceci se formalise par des théorèmes puissants comme celui de l'isomorphisme de Curry-Howard.

Extrait de la solution proposée par Y.V.

L'algorithme Cfcm crée l'ensemble des nœuds de la cfcm contenant un élément x. Pour cela, il utilise deux procédures Parcours-amont et Parcours-aval, et trois tableaux : les tableaux `amont` et `aval` contiennent une entrée booléenne fixée initialement à *faux* pour chaque noeud. Le troisième tableau `table-cfcm` contient une entrée booléenne pour chaque noeud, indiquant finalement si le noeud indiqué appartient à la cfcm cherchée.

```
Parcours-amont (x) :
  amont[x] ← vrai
  Pour tout y tel que (y, x) dans D faire
    Si amont[y] est faux alors
      Parcours-amont (y)

Parcours-aval (x) :
  aval[x] ← vrai
  Pour tout y tel que (x, y) dans D faire
    Si aval[y] est faux alors
      Parcours-aval (y)

Cfcm (x) :
  Parcours-amont (x)
  Parcours-aval (x)
  Pour tout y de D faire
    Si (amont[y] et aval[y]) est vrai alors
      table-cfcm[y] ← vrai
```

On considère un nœud de la forme (x, x) et on lui applique la procédure Cfcm. Tester la surjectivité, respectivement la bijectivité, revient à tester si la cfcm obtenue est $\tilde{\mathcal{G}}$, respectivement $\mathcal{D}$. Ceci se fait à l'aide du tableau `table-cfcm`.

Question 3.10

D'après la Question 2.3, si $\mathcal{A}$ est bijective alors $\mathcal{A}\,|_{\mathcal{P}}$ est bijective.

Supposons que $\mathcal{A}\,|_{\mathcal{P}}$ est bijective. D'après la Partie 2, il existe q tel que $\mathcal{A}\,\big|_{\mathcal{F}_q}$ est injective. D'après la Question 3.8, on en déduit que $\overline{\mathcal{G}} = \tilde{\mathcal{G}}$. Si on a $\mathcal{D} = \tilde{\mathcal{G}}$, cela entraîne que $\mathcal{A}$ est injective (Question 3.7) et donc que $\mathcal{A}$ est bijective (Question 2.1). Il reste donc à démontrer que $\mathcal{D} = \tilde{\mathcal{G}}$. Supposons que $\mathcal{D} \neq \tilde{\mathcal{G}}$ et notons C une cfcm de $\mathcal{D}$ différente de $\tilde{\mathcal{G}}$. La cfcm C contient au moins un circuit. Par concaténation infinie de ce circuit, on obtient un chemin bi-infini périodique dans $\mathcal{D}$. Soit x et y les deux configurations correspondant aux deux coordonnées du chemin. On a $x \neq y$ ($C \cap \tilde{\mathcal{G}} = \emptyset$) et $x, y \in \mathcal{P}$ (Question 3.2). Enfin, par définition même du graphe produit, on a $\mathcal{A}(x) = \mathcal{A}(y)$, ce qui contredit l'injectivité de $\mathcal{A}\,|_{\mathcal{P}}$. On en conclut que $\mathcal{D} = \tilde{\mathcal{G}}$.

4 Indécidabilité en dimension 2

Question 4.1

L'idée est d'énumérer les couples de configurations périodiques distinctes, et pour chaque couple de vérifier si les images obtenues sont distinctes.

On se sert de la propriété suivante : les configurations périodiques ont des images périodiques de même période (Question 1.2). Pour déterminer $\mathcal{A}(c)$ pour $c \in \mathcal{P}_p,\ p = (p_1, p_2)$, il suffit donc de calculer $\mathcal{A}(c)(u),\ u \in [0, p_1 - 1] \times [0, p_2 - 1]$. Pour cela, il suffit de connaître $c(u)$ pour $u \in [0, p_1 - 1] \times [0, p_2 - 1]$. Ainsi, pour $c_1, c_2 \in \mathcal{P}_p$, vérifier si $\mathcal{A}(c_1) = \mathcal{A}(c_2)$ est réalisable en temps fini. Comme $|\mathcal{P}_p| = |S|^{p_1 \times p_2} < \infty$, vérifier si $\mathcal{A}$ est surjectif sur $\mathcal{P}_p$ est réalisable en temps fini.

Pour l'énumération, on peut par exemple remarquer que toute configuration périodique est périodique de période $(n, n), n \in \mathbb{N}$, et ne tester que les périodes de cette forme en suivant l'ordre naturel de $\mathbb{N}$. Alternativement, on peut utiliser toute bijection de $\mathbb{N}$ dans $(\mathbb{N} \setminus \{0\})^2$.

Question 4.2

Soit $\mathcal{A}$ l'automate cellulaire à tester. L'algorithme consiste à énumérer tous les automates cellulaires $\mathcal{B}$ ayant le même ensemble d'états que $\mathcal{A}$ mais éventuellement un voisinage différent. Ensuite, on calcule $\mathcal{A} \circ \mathcal{B}$ et $\mathcal{B} \circ \mathcal{A}$ et on s'arrête si $\mathcal{A} \circ \mathcal{B} = \mathcal{B} \circ \mathcal{A} = \mathrm{Id}$. D'après le théorème de Richardson, l'algorithme s'arrête si et seulement si $\mathcal{A}$ est bijectif. Il reste à démontrer que chaque itération est réalisable en temps fini.

Soit deux automates cellulaires $\mathcal{C}$ et $\mathcal{D}$ de même dimension et ensemble d'états. En adaptant légèrement l'analyse de la Question 1.4, on obtient que $\mathcal{C} \circ \mathcal{D}$ est un automate cellulaire de même dimension et ensemble d'états que $\mathcal{C}$ et $\mathcal{D}$, et dont le voisinage et la fonction locale se déterminent à partir de ceux de $\mathcal{C}$ et $\mathcal{D}$.

Maintenant vérifier qu'un automate cellulaire $(n, S, \{v_1, \ldots, v_l\}, \delta)$ réalise la fonction identité consiste à vérifier que : (1) $\exists i,\ v_i = (0, \ldots, 0)$; (2) $\delta(x_1 \cdots x_l) = x_i,\ \forall (x_1 \cdots x_l) \in S^l$.

On déduit des deux points précédents qu'étant donné $\mathcal{A}$ et $\mathcal{B}$, déterminer si '$\mathcal{A} \circ \mathcal{B} = \mathcal{B} \circ \mathcal{A} = \mathrm{Id}$' est réalisable en temps fini.

Pour l'énumération, on pourra ordonner les automates cellulaires suivant la 'dimension' $\|V\|$ de leur voisinage V.

Remarque : Les voisinages d'un automate cellulaire et de son inverse peuvent être très différents. En dimension 1, on trouve des bornes pour la dimension du voisinage de l'inverse d'un automate cellulaire donné, en utilisant la décomposition du nombre d'états en facteurs premiers. En dimension 2, le Théorème 3 montre qu'il n'existe pas de borne calculable pour la dimension du voisinage de l'inverse !

Question 4.3

Supposons l'équivalence '$\mathcal{A}_{|\mathcal{P}}$ bijectif $\iff$ $\mathcal{A}$ bijectif' vraie en dimension 2. D'après la Partie 2, on a '$\mathcal{A}_{|\mathcal{P}}$ injectif $\iff$ $\mathcal{A}$ bijectif', ou encore '$\mathcal{A}$ non-bijectif $\iff$ $\mathcal{A}_{|\mathcal{P}}$ non-injectif'. Donc en combinant les algorithmes des deux questions précédentes, on obtient un algorithme résolvant le problème de la bijectivité ce qui est impossible d'après le Théorème 3. On en conclut que l'équivalence n'est pas vraie en dimension 2.

Pour être complet, il faut expliquer comment combiner deux algorithmes pour les faire fonctionner “en parallèle”. Plusieurs méthodes sont possibles. La plus simple consiste à intercaler les deux énumérations : On considère le premier couple de configurations périodiques distinctes, on vérifie que $\mathcal{A}$ les envoie sur des images différentes sinon on s'arrête et on déclare $\mathcal{A}_{|\mathcal{P}}$ non-injectif ; ensuite on passe au premier automate cellulaire $\mathcal{B}$ énuméré, si '$\mathcal{A} \circ \mathcal{B} = \mathcal{B} \circ \mathcal{A} = \mathrm{Id}$', alors on s'arrête et on déclare $\mathcal{A}$ bijectif, sinon on continue et on passe au suivant dans l'énumération des couples de configurations périodiques, etc.

5 Pour finir

Question 5.1

Il suffit effectivement de montrer les non-implications numérotées !

L'automate cellulaire $\mathcal{X}$ permet de prouver les non-implications 1 et 2. En reprenant l'analyse faite en Question 1.1, on vérifie que $\mathcal{X}|_{\mathcal{P}}$ est surjective et non-injective. Ceci montre la non-implication 2. D'autre part $\mathcal{X}|_{\mathcal{F}_0}$ n'est pas surjective. En effet, la configuration $(\dots,0,\dots,0,1,0,\dots,0,\dots)$ a deux antécédents, $(\dots,0,0,0,1,1,1\dots)$ et $(\dots,1,1,1,0,0,0,\dots)$, qui ne sont ni 0-finis ni 1-finis. Ceci montre la non-implication 1.

Pour montrer la non-implication 3, il convient “d'améliorer” $\mathcal{X}$. Considérons l'automate cellulaire $\mathcal{Y}$ de dimension 1, avec 3 états $S = \{B, 0, 1\}$, de voisinage $V = (-1, 0)$ et de fonction de transition $\delta_{\mathcal{Y}}$ définie comme suit :

$$\begin{cases} \delta_{\mathcal{Y}}(x, B) = B, & \forall x \in S \\ \delta_{\mathcal{Y}}(B, x) = x, & \forall x \in S \\ \delta_{\mathcal{Y}}(x, y) = x + y \mod 2, & \text{sinon .} \end{cases}$$

Montrons que $\mathcal{Y}$ restreint aux configurations B-finies est surjectif mais que, restreint aux configurations périodiques, il n'est pas injectif.

Soit c une configuration B-finie. Nous lui associons c' défini pour $x \in \mathbb{Z}$ par

$$\begin{cases} c'(x) = B, & \text{si } c(x) = B \\ c'(x) = c(x), & \text{si } c(x-1) = B \\ c'(x) = c(x) + c'(x-1) \mod 2, & \text{sinon .} \end{cases}$$

Par construction $\mathcal{Y}(c') = c$ et $c' \in \mathcal{F}_B$, donc $\mathcal{Y}|_{\mathcal{F}_B}$ est surjectif. D'autre part, l'automate $\mathcal{Y}$ restreint aux configurations à valeurs dans $\{0, 1\}$ coïncide avec l'automate $\mathcal{X} \circ \sigma_1$. Donc en particulier $\mathcal{Y}|_{\mathcal{P}}$ n'est pas injectif.

Question 5.2

Cette question se résolvait indépendamment du reste du problème, à l'aide d'arguments simples d'algèbre linéaire. On décrit ci-dessous une façon possible de procéder.

Les modules n'étant pas au programme, il convient de plonger $\mathbb{Z}^n$ dans un espace vectoriel. L'ensemble $\mathbb{Z}^n$ est un sous-ensemble du $\mathbb{Q}$-espace vectoriel $\mathbb{Q}^n$. On montre aisément qu'une famille libre de $\mathbb{Z}^n$ (définition de l'énoncé) est une famille libre de $\mathbb{Q}^n$ (définition usuelle). On en déduit que la cardinalité maximale d'une famille libre de $\mathbb{Z}^n$ est n.

On montre aisément le résultat préliminaire suivant : soit A une matrice carrée inversible à coefficients dans $\mathbb{Q}$, alors A^{-1} est à coefficients dans $\mathbb{Q}$.

Soit c une configuration admettant pour période une famille libre $(u_1, \ldots, u_n)$ de $\mathbb{Z}^n$. Soit U la matrice formée des lignes $u_1, \ldots, u_n$ (c'est-à-dire $\forall i$, $(U_{i1}, \cdots, U_{in}) = u_i$). La matrice U est inversible (car $(u_1, \ldots, u_n)$ est une base de $\mathbb{Q}^n$) et on note $V = (V_{ij})$ sa matrice inverse. La matrice V est à coefficients dans $\mathbb{Q}$. Soit $(e_1, \ldots, e_n)$ la base canonique de $\mathbb{Q}^n$. On a $e_i = \sum_j V_{ij} u_j$, $\forall i$. Par réduction au même dénominateur et multiplication par ce dénominateur, on obtient l'existence de $\alpha_i \in \mathbb{Z} \setminus \{0\}$ et de $W_{ij} \in \mathbb{Z}$ tels que $\alpha_i e_i = \sum_j W_{ij} u_j$. On a pour tout $u \in \mathbb{Z}^n$,

$$c(u + \alpha_i e_i) = c(u + \sum_j W_{ij} u_j) = c(u) ,$$

en utilisant que les u_i sont des périodes de c. On conclut que c est totalement périodique de périodes $(\alpha_1, 0, \ldots, 0), (0, \alpha_2, 0, \ldots, 0), \ldots, (0, \ldots, 0, \alpha_n)$.

Solution proposée par Y.V.

En considérant la base canonique de $\mathbb{Q}^n$, on constate immédiatement que le cardinal maximal d'une famille de $\mathbb{Z}^n$ libre est supérieur ou égal à n. Soit alors une famille $(u_1, \ldots, u_l)$ de $\mathbb{Z}^n$. de cardinal l supérieur ou égal à $n+1$. Comme $\mathbb{Q}^n$ est un espace vectoriel de dimension n, il existe une combinaison $\mathbb{Q}$-linéaire nulle des vecteurs $(u_1, \ldots, u_l)$. En réduisant les coefficients obtenus au même dénominateur d, et en les multipliant par d, on obtient une combinaison $\mathbb{Z}$-linéaire nulle de la famille de vecteurs. Donc $l \leq n$. On a ainsi montré que le cardinal maximal d'une famille libre de $\mathbb{Z}^n$ est exactement n.

Soit maintenant c une configuration admettant pour périodes les éléments d'une famille libre $(u_1, \ldots, u_n)$ de cardinalité n. On note $p_i : \mathbb{Z}^n \to \mathbb{Z}^{n-1}$ l'application telle que

$$p_i(c_1, \ldots, c_i, \ldots, c_n) = (c_1, \ldots, c_{i-1}, c_{i+1}, \ldots, c_n) .$$

Fixons alors un entier j dans $\{1, \ldots, n\}$. La cardinalité maximale d'une famille libre dans $\mathbb{Z}^{n-1}$ étant $n-1$, il existe une famille $(\lambda_1, \ldots, \lambda_n) \neq (0, \ldots, 0)$ telle que

$$\sum_{i=1}^{n} \lambda_i p_j(u_i) = (0, \ldots, 0) ,$$

ce qui entraîne

$$\exists z \in \mathbb{Z}\backslash\{0\}, \quad v_j = \sum_{i=1}^{n} \lambda_i u_i = (\underbrace{0, \ldots, 0}_{j-1}, z, 0, \ldots, 0)$$

(si z était nul cela contredirait la liberté de la famille des u_i).

Mais c admet, de façon évidente, toute combinaison $\mathbb{Z}$-linéaire des u_i comme période. Le vecteur v_j est donc une période de c. Ceci étant vrai pour tout j, on conclut que c est totalement périodique.

Question 5.3

Si un lecteur trouve une réponse concernant la validité d'une ou plusieurs des trois implications, il est vivement encouragé à faire part aux auteurs de sa solution.

Bilan de l'épreuve

Un quart des candidats n'entrent pas du tout dans le sujet et ne traitent que les questions immédiates. La moitié des candidats n'abordent aucune des questions difficiles et se contentent de répondre aux questions faciles des troix premières parties. Les questions d'algorithmique des graphes (Questions 3.6 et 3.9) n'ont pratiquement pas été traitées. En revanche les algorithmes de semi-décidabilité demandés aux Questions 4.1 et 4.2 ont souvent été trouvés par les rares candidats ayant abordé cette partie.

À titre indicatif, voici une évaluation de la proportion de candidats ayant répondu correctement aux questions posées (pour les questions non-immédiates figurant dans la partie du sujet abordée par la majorité des candidats) :

Question	**1.4**	**2.2**	**2.3**	**2.4**	**2.5**	**2.6**	**3.2**	**3.3**	**3.4**
%	40	10	30	40	50	1	40	40	1

Dans la bibliographie ci-dessous, on a inclus les références originales des théorèmes donnés sans démonstration dans le sujet. La référence [1] est un article de synthèse contenant une bibliographie plus complète. Enfin l'article [6] est consacré à la construction de la partie 3.

Références

[1] B. Durand. Global properties of cellular automata. In E. Goles and S. Martinez, editors, *Cellular Automata and Complex Systems*. Kluwer, 1998.

[2] J. Kari. Reversibility and surjectivity problems of cellular automata. *Journal of Computer and System Sciences*, 48 :149–182, 1994.

[3] E.F. Moore. Machine models of self–reproduction. *Proc. Symp. Appl. Math.*, 14 :13–33, 1962.

[4] J. Myhill. The converse to Moore's garden-of-eden theorem. *Proc. Am. Math. Soc.*, 14 :685–686, 1963.

[5] D. Richardson. Tesselations with local transformations. *Journal of Computer and System Sciences*, 6 :373–388, 1972.

[6] K. Sutner. De Bruijn graphs and linear cellular automata. *Complex Systems*, 5 :19–30, 1991.