



HAL
open science

Automates à galets : un état de l'art.

Marianne Delorme

► **To cite this version:**

Marianne Delorme. Automates à galets : un état de l'art.. [Research Report] LIP RR-1997-23, Laboratoire de l'informatique du parallélisme. 1997, 2+52p. hal-02101923

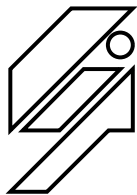
HAL Id: hal-02101923

<https://hal-lara.archives-ouvertes.fr/hal-02101923>

Submitted on 17 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Laboratoire de l'Informatique du Parallélisme

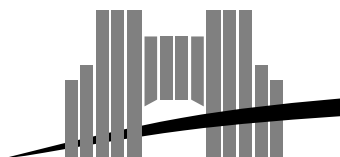
Ecole Normale Supérieure de Lyon
Unité de recherche associée au CNRS n°1398

Automates à galets : un état de l'art.

Marianne Delorme

2 septembre 1997

Research Report N° 97-23



Ecole Normale Supérieure de Lyon

46 Allée d'Italie, 69364 Lyon Cedex 07, France

Téléphone : (+33) (0)4.72.72.80.00 Télécopieur : (+33) (0)4.72.72.80.80

Adresse électronique : lip@lip.ens-lyon.fr

Automates à galets : un état de l'art.

Marianne Delorme

2 septembre 1997

Abstract

The purpose of this paper is to give an overview on pebble automata, which can be encountered in different domains, as figures (families) recognition, complexity theory and labyrinths theory . It gives definitions, examples and some basic theorems with their proofs.

Keywords: Automata, pebbles, graphoids, labyrinths.

Résumé

Ce rapport est un état de l'art sur les automates à galets que l'on rencontre dans divers domaines comme celui de la reconnaissance de (familles de) figures, théorie de la complexité, mais aussi théorie des labyrinthes.

Mots-clés: Automates, galets, graphoïdes, labyrinthes.

Table des matières

1	Introduction	2
2	Automates à galets et théorie de la Complexité	4
3	Les labyrinthes et autres dédales	5
3.1	Le plan discret, différents motifs et les “dédalles à la Blum” (mazes)	6
3.1.1	Figures ou motifs du plan	6
3.1.2	Dédalles	6
3.1.3	Dédalle à la Blum et “son graphe dual”	7
3.2	Dédalles “à la Budach”	8
3.3	Graphoïdes	9
3.3.1	Définition	9
3.3.2	R -graphoïdes	11
3.3.3	C -graphoïde	13
3.3.4	Plongements de graphes et de graphoïdes	13
3.3.5	nD -graphoïdes, $n \in \{2, 3\}$	14
4	Des automates sur des motifs, dans des labyrinthes	15
4.1	Les 4-automates	16
4.2	Automates sur dédales et divers graphoïdes	16
4.3	Sur des performances d’automates finis dans des labyrinthes . . .	20
4.3.1	Reconnaissance et ou décision	20
4.3.2	Plaçons-nous maintenant dans le cadre de l’exploration de dédales à la Blum	25
4.3.3	Dans le cadre des R -graphoïdes	26
4.3.4	Exploration du monde : cas particulier d’un graphe infini	27
5	Automates à galets et quelques autres modèles	29
6	Des automates à galets ou à pointeurs dans des labyrinthes	32
6.1	Le monde vide	32
6.2	Le monde “demi-vidé”	34
6.3	Exploration des dédales à la Blum, finis, ou $2D$ -ficographe . . .	37
7	Labyrinthes infinis, incographe	45
7.1	Principaux résultats	45
7.2	Eléments de preuves	45
7.3	Mais	46
8	Quelques problèmes ouverts	46

1 Introduction

Un automate à galets est un automate fini qui peut se déplacer généralement dans un graphe (fini ou infini) connexe, et qui, de plus, dispose d'un nombre fini de marqueurs, ici appelés galets, dont il sait reconnaître la présence ou l'absence, qu'il peut déposer ou reprendre suivant certaines modalités.

C'est dans le cadre d'une étude systématique des familles de figures (bornées ou non) du plan par des automates cellulaires [Tou97] que s'est naturellement imposée l'idée de définir une notion de rationalité pour de tels objets, et les automates à galets, que nous avons découverts dans un article de Max Garzon [Gar93] - qui cherchait, lui, à caractériser des classes de graphes de Cayley par des machines finitaires - nous ont paru des candidats possibles.

Nous avons également pensé qu'ils pourraient permettre de caractériser des familles de signaux (dans le plan, mais aussi dans l'espace) qui jouent un rôle fondamental dans le comportement des automates cellulaires.

Plus généralement, ces machines - simples - qui se déplacent suivant des trajectoires qui sont a priori très générales nous ont semblé intéressantes en soi, mais également susceptibles d'applications dans le domaine de l'imagerie, par exemple, plus précisément, en tomographie.

Notons ici que le mode de fonctionnement de ces machines se distingue fondamentalement de celui des graphes d'automates (et particulièrement des jeux à galets) puisque le graphe sur lequel va opérer l'automate (ou plus généralement un système d'automates coopérants) est, à l'origine, nu : s'il contient une marque "extérieure", ce ne peut être que l'automate (ou le système d'automates) chargé de tous ses galets sur l'une de ses positions (sommet ou arête), ou une couleur attribuée à une position.

Telles sont, à grands traits, les motivations qui nous ont conduit à une recherche systématique des travaux relatifs à ce modèle de machines (on pourrait y ajouter le défi que représente le passage de la dimension 1 à une dimension supérieure, la dimension 2 étant vue comme une porte ouverte aux autres). Et c'est à *un* état de l'art de ces travaux qu'est consacré ce rapport.

Les recherches menées sur et par les automates à galets peuvent se répartir suivant deux grandes tendances, qui ne sont pas strictement isolées, bien au contraire.

La première est celle de l'exploration d'univers et de la reconnaissance de structures (deux démarches qui sont dans de nombreux cas évidemment "équivalentes", mais que l'on ne peut confondre : la reconnaissance implique une procédure d'arrêt partielle (arrêt sur les entrées reconnues) et la décision - qu'elle généralise - nécessite une procédure d'arrêt totale (arrêt sur toute entrée), ce qui n'est pas le cas de l'exploration). Elle contient la théorie des labyrinthes. On peut la voir comme une étude comparative de la puissance des machines, et par conséquent interpréter certains de ses résultats comme des mesures de complexité.

Mais la seconde est plus explicitement et directement rattachée à la problématique de la complexité, bien qu'elle produise et s'appuie sur des résultats qui

pourraient relever de la première.

Ce dernier courant est particulièrement marqué par les travaux de W. J. Savitch [Sav70], [Sav73], S. A. Cook et C. W. Rackoff [CR80], L. Budach [Bud81], J. H. Chang, O. H. Ibarra et M. A. Palis [CIP86]. Notons cependant immédiatement que les machines considérées ont éventuellement, malgré un nom commun, des définitions qui diffèrent plus ou moins. Par exemple les automates de Cook et Rackoff n'ont pas de localisation sur le graphe, ils ne sont que des contrôles finis qui y dirigent les déplacements des galets.

C'est au premier courant que nous intéresserons plus particulièrement dans ce texte.

On dit que c'est Shannon qui a introduit en 1951, [Sha51], l'idée d'un automate (une souris) se déplaçant dans un espace de dimension 2 [Bud78], [Coy78]. Dans les années 70 un certain nombre de travaux, publiés ou non, sont dédiés à différents types d'automates œuvrant dans des structures de dimension supérieure ou égale à 2. Ainsi, par exemple, M. Paterson ([BS77]) considère des automates finis se mouvant dans l'espace en y laissant une trace (worms automata). D'un autre côté, J. H. Conway popularise par "le jeu de la vie" [Con70], [Con71], les automates cellulaires de dimension 2, un modèle d'une autre veine dû à Von Neumann en 1948. Ce foisonnement de travaux, dont ceux de Minsky [Min67], juste évoqués, esquisse le cadre dans lequel apparaissent les automates à galets.

Plusieurs sources comme [BS77], [CR80], font état d'une conférence de M. Rabin, donnée en 1967, concernant un automate à galets se mouvant le long des arêtes d'un graphe, au cours de laquelle il aurait exposé la preuve qu'aucune telle machine ne peut explorer complètement un graphe non orienté régulier de degré 4.

Blum et Sakoda [BS77] évoquent encore, à cette même époque (1977), par exemple des travaux de M. Sipser (deux automates finis et un galet peuvent simuler un automate à deux compteurs), de Cook et Rackoff (portant sur des bornes sur le nombre de galets permettant d'explorer tout graphe à n sommets), et de B. Tarjan et W. Paul (cherchant à démontrer (sans y parvenir ...) leur conjecture qu'une famille finie d'automates ne peut explorer tout graphe planaire de degré 3).

Cependant le premier article publié semble être celui de M. Blum et C. Hewitt [BH67], qui traite de la puissance de différents types d'automates à galets sur des figures du plan strictement enserrées dans un cadre-frontière carré, avec l'objectif de caractériser des familles de telles figures par de telles machines.

Dans cette même optique de reconnaissance de familles de figures bornées, on peut aussi noter les résultats de J. Mylopoulos [My172], et aussi de A. N. Shah [Sha74]. Plus tard, M. Blum, W. J. Sakoda et D. Kozen s'intéressent à la possibilité d'explorer à l'aide de tels automates (avec ou sans galets) des univers (finis ou infinis) obstrués, appelés *labyrinthes*, puis d'autres familles de graphes [BS77], [BK78].

Dans la même veine, on trouve encore les travaux plus tardifs de A. Szepietowski [Sze82], [Sze83b], [Sze83a].

Parallèlement à ces travaux, que nous pourrions dire “de la branche américaine”, se développent les travaux “de la branche est-allemande” à partir d’une conjecture de K. Döpp : étant donnée une classe d’automates \mathcal{A} et une classe de labyrinthes \mathcal{L} qu’il précise, “*Pour tout automate de \mathcal{A} , on sait définir de façon constructive un labyrinthe de \mathcal{L} qu’il ne maîtrise pas*” [Döp71]), puis autour de L. Budach. La problématique est encore celle de l’exploration ou de la maîtrise de certaines classes de graphes, en particulier de labyrinthes co-finis. Dans son article [Bud78], L. Budach prouve, de façon très sophistiquée, la conjecture de Döpp : pour tout automate fini on sait effectivement construire un labyrinthe qui le piège. Ce travail se prolonge et se précise principalement dans les travaux de Müller [Mül79], F. Hoffmann [Hof81], [Hof85], A. Hemmerling [Hem86b], [Hem87b], [Hem89], [Hem91].

A. Hemmerling [Hem87b] agrmente les automates de plusieurs têtes et montre qu’ils sont strictement plus puissants que les systèmes finis d’automates.

Nous donnerons un aperçu rapide, dans le paragraphe qui suit, des travaux de complexité. Dans le paragraphe 3 seront précisées les définitions des principales classes d’objets sur lesquels agiront les automates, leurs liens et quelques unes de leurs propriétés essentielles. Le paragraphe suivant (4) sera consacré aux automates, à ce que sont leurs comportements dans les différentes structures intéressantes et à ce que l’on entend par exploration, maîtrise et reconnaissance d’un objet ou d’une classe d’objets. Des exemples illustreront les possibilités et les limites de ces automates, justifiant l’introduction, au paragraphe 5, des automates à galets. Dans la voie des limites seront précisés quelques résultats relatifs à l’existence de pièges d’automates. Des résultats fondamentaux illustrant la puissance des automates à galets seront énoncés et éventuellement démontrés aux paragraphes 6 et 7. Nous terminerons, paragraphe 8, par une série de problèmes encore ouverts à notre connaissance, pour une part répertoriés dans [Hem86a], ouvrage qui présente une remarquable synthèse de la *théorie des labyrinthes*.

2 Automates à galets et théorie de la Complexité

Les travaux que nous avons retenus se développent suivant deux voies : celle de la simulation d’algorithmes non déterministes par des algorithmes déterministes, et celle de la comparaison de la puissance de reconnaissance de machines sur des objets qui sont des graphes ou des mots.

- Dans son premier article, W. J. Savitch [Sav70] montre que la question de savoir si les classes de complexité en espace linéaire déterministe et non-déterministe $DSPACE(L(N))$ et $NSPACE(L(N))$ sont égales mène à des problèmes concernant des automates à galets dans certains environnements, qu’il nomme labyrinthes (mazes). Un tel automate est un automate

fini avec alphabet d'entrée $\{0, 1\}$, dont l'ensemble des états a la structure d'un groupe cyclique fini, et qui a la possibilité de manipuler un nombre fini de marques (les galets) dans son environnement.

Dans [Sav73], il montre qu'il existe des automates à galets qui reconnaissent une certaine famille de graphes, qu'il dénomme "threadable mazes" si et seulement si toute machine de Turing non déterministe à ruban $L(n)$ -borné peut être simulée par une machine de Turing déterministe à ruban $L(n)$ -borné, pourvu que $L(n) \geq \log_2(n)$. Mais il conjecture qu'aucun automate à galets ne peut reconnaître cette famille, ce qui mettrait pourtant en évidence une différence entre les classes de complexité en espace déterministe et non déterministe.

- Savitch avait donc montré que l'ensemble des "threadable mazes" est log-space-complet pour $NSPACE(\text{Log}(n))$.

Si cet ensemble était aussi dans $DSPACE(\text{Log}(n))$, alors on aurait $NSPACE(\text{Log}(n)) = DSPACE(\text{Log}(n))$ et, d'après un autre résultat de Savitch, cela entraînerait $NSPACE(L(n)) = DSPACE(L(n))$ pour toute fonction L telle que $L(n) \geq \log_2(n)$. Mais, dans [CR80], il est prouvé que cet ensemble des "threadable mazes" n'est pas dans $DSPACE(L(n))$, au moyen d'un certain type d'automates à "galets sauteurs".

- Par ailleurs dans [Bud81], Budach prouve que pour tout automate à deux galets du type Savitch, il existe un labyrinthe qu'il ne peut reconnaître.
- Le travail de O. Ibarra et de ses coauteurs sur les automates à galets est d'une veine un peu différente. Il est dédié à l'étude de la puissance de machines opérant en espace sous $\log(n)$. Les machines qu'ils considèrent fonctionnent donc sur des mots, et les galets sont déposés ou prélevés sur des rubans de Turing. Ils prouvent alors que dans certains cas un galet accroît la puissance de reconnaissance d'une machine.

D'autres articles peuvent encore être cités dans cette voie de la complexité comme [GH94], [PGK91], et plus anciennement [MF71].

3 Les labyrinthes et autres dédales

Les structures dans lesquelles se déplacent les automates s'étendent du plan discret, éventuellement coloré, considéré assez informellement comme un pavage par des tuiles carrées "unité" (ou cellules) à la structure formelle très générale de graphoïde introduite par Hemmerling [Hem89], en passant par différents types de graphes, comme "les dédales à la Blum ou à la Budach". La terminologie est assez variable. Dans la suite, nous conviendrons d'utiliser le terme de labyrinthe comme un terme générique, pour toutes les structures que nous rappelons et

dans lesquelles nous nous proposons de faire évoluer des automates finis plus ou moins “renforcés”. Mais, dans les énoncés nous devons être, généralement, plus précis.

3.1 Le plan discret, différents motifs et les “dédalles à la Blum” (mazes)

3.1.1 Figures ou motifs du plan

Les premiers articles qui évoquent des automates à galets [BH67], [Myl72], [Sha74], sont consacrés à l’étude de motifs 2-dimensionnels et à la capacité de 4-automates à reconnaître ou caractériser certaines de leurs propriétés. Les motifs sont alors principalement des ensembles finis connexes de la grille carrée classique, dont les points (les cellules) sont coloré(e)s en blanc et noir, et qui ont une frontière. Chaque motif est supposé avoir un point (une cellule) distingué(e) qui sera le point (la cellule) sur lequel (laquelle) l’automate commencera à fonctionner.

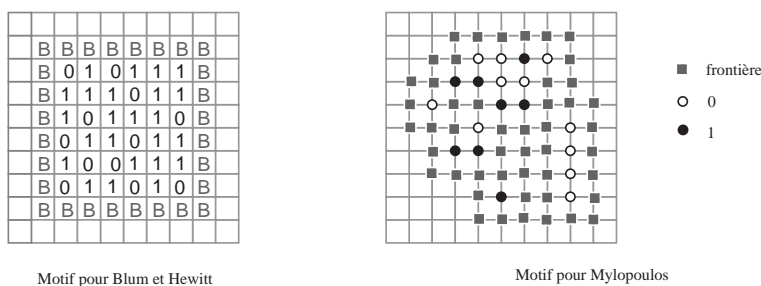


FIG. 1 - Différents motifs ou figures finies du plan

3.1.2 Dédalles

En suivant Blum et ses coauteurs de la fin des années 70, on définira un dédale de la façon suivante.

Par *sommet*, on entendra point du plan euclidien à coordonnées entières, par *arête* tout segment unité joignant deux sommets et parallèle à l’un des axes déterminés par $(0, 0)$, $(1, 0)$ et $(0, 1)$ par cellule l’aire unité déterminée par quatre sommets convenables et les arêtes qui les joignent (arêtes incluses).

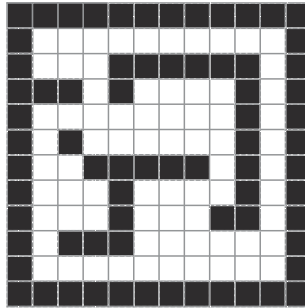
“Un dédale à la Blum” est alors une assignation de la couleur noire ou blanche (exclusivement) à chaque cellule, de telle sorte que :

- il n’y ait qu’un nombre fini de cellules blanches,
- deux cellules blanches quelconques soient liées par un chemin de cellules blanches ayant une arête adjacente commune (4-connexité).

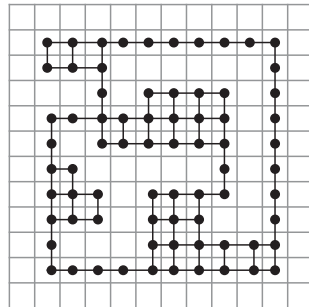
En fait, le dédale est compris, plus brièvement, comme l'ensemble des cellules blanches, et sa définition est étendue implicitement au cas infini.

On appelle *obstacle* tout ensemble maximal de cellules noires simplement connexes (c'est-à-dire ayant au moins un sommet commun, 8-connexité). L'unique obstacle infini d'un tel dédale est encore désigné comme sa frontière, alors que l'ensemble des arêtes d'un obstacle qui le séparent des cellules blanches qui lui sont adjacentes est plutôt désigné comme sa *limite*. Une cellule blanche adjacente à une cellule noire est dite aussi cellule-limite.

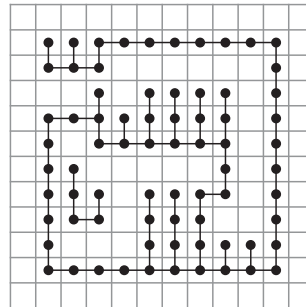
3.1.3 Dédale à la Blum et “son graphe dual”



Dédale à la Blum M (4-connexité)



Graphe $G(M)$ (4-connexité)



Graphe $\text{red}(G(M))$ (8-connexité)

FIG. 2 - Dédale à la Blum et graphes associés

A l'ensemble des cellules blanches d'un dédale M on sait associer canoniquement un graphe planaire, son dual, $G(M)$: à chaque cellule blanche, on associe un sommet du plan discret, de telle sorte que deux cellules blanches adjacentes par une arête e du plan soient représentées par les extrémités d'une arête perpendiculaire à e , qui constitue elle-même une arête de $G(M)$. Seuls les sommets

et les arêtes ainsi obtenues appartiennent à $G(M)$.

Cependant il faut être attentif dans l'usage de cette correspondance. Par exemple, il n'y a pas égalité entre le nombre de faces (voir paragraphe 3.3.2) de $G(M)$ et l'ordre de connectivité de M défini par Rosenfeld [Ros70] comme le nombre de 8-composantes connexes de son ensemble d'obstacles (rappelons que la 8-connectivité est fondée sur le voisinage de Moore dans lequel deux cellules $z = (i_1, i_2)$ et $z' = (i'_1, i'_2)$ sont voisines lorsque $\max(|i_1 - i'_1|, |i_2 - i'_2|) = 1$). Toutefois, on retrouve cette égalité lorsque l'on associe à M , non plus $G(M)$, mais le graphe $\text{red}(G(M))$ réduit de $G(M)$ et obtenu comme suit :

$\text{red}(G(M)) = (V, E_0)$ où V est l'ensemble des sommets de $G(M)$ et où E_0 est obtenu en supprimant de l'ensemble des arêtes de $G(M)$ toutes les arêtes horizontales dont les extrémités ont des voisins sud, eux-mêmes connectés par une arête horizontale. Voir la figure 2.

Alors on a le résultat suivant :

Proposition 1 *Pour tout dédale M , $\text{red}(G(M))$ est un graphe planaire fini connexe, et le nombre de faces de $\text{red}(G(M))$ est exactement l'ordre de connectivité de M .*

Pour l'école allemande, un dédale est plutôt un sous-graphe connexe de la grille classique. Rappelons cependant tout d'abord la définition plus générale de Budach, qui illustre bien le cadre et l'esprit des travaux de cette école.

3.2 Dédales “à la Budach”

- Au départ, Budach se donne un ensemble de directions $D = \{n, s, e, w\}$ tel que si d est une direction, \bar{d} est la direction opposée et $\bar{\bar{d}} = d$.
- Il considère alors le monoïde D^* des mots sur D , le groupe libre F sur $\{n, e\}$, l'homomorphisme $\alpha : D^* \rightarrow F$ qui à n associe n , à e , e , à s , n^{-1} , à w , e^{-1} , et l'application $\gamma : F \rightarrow D^*$, qui à l'élément neutre de F associe le mot vide et à tout $d_1^{\epsilon_1} \dots d_m^{\epsilon_m}$ de F (qui ne contient donc pas de facteur $d_i^{\epsilon_i} d_i^{-\epsilon_i}$), associe $\gamma(d_1^{\epsilon_1}) \dots \gamma(d_m^{\epsilon_m})$, avec $\gamma(d) = d$ et $\gamma(d^{-1}) = \bar{d}$.
Cela permet de définir proprement l'ensemble R des mots (sur D) réduits par $R = \gamma(\alpha(D^+))$, où D^+ est l'ensemble des mots non vides sur D .
- Un *prélabynthe* \mathcal{L} consiste alors en :
 - Un ensemble Z d'éléments appelés points,
 - Un sous-ensemble E de $Z \times R \times Z$, dont les éléments (P, p, Q) sont appelés arêtes (la composante p de (P, p, Q) étant l'étiquette de (P, p, Q)),
ces objets satisfaisant les conditions :
 - Si (P, p, Q) et (P, p', Q') sont des arêtes de \mathcal{L} telles que les premières lettres de p et p' sont les mêmes, alors $p = p'$ et $Q = Q'$,

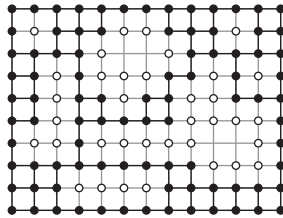
- Si (P, p, Q) est une arête de \mathcal{L} , (Q, \bar{p}, P) aussi (où \bar{p} désigne le mot renversé de p), et elle est appelée l'inverse de (P, p, Q) .

Si, de plus, de tout point P de Z il part au moins une arête, le prélababyrinthe est dit *strict*.

Pour tout point P du prélababyrinthe, $val(P)$ désigne l'ensemble $\{d \in D / \text{il existe } p \in D^* \text{ et } Q \in Z \text{ tels que } (P, dp, Q) \in E\}$, que l'on appelle *valence* de P et dont le cardinal est le *degré* de P .

- Un *labyrinthe* est un prélababyrinthe dans lequel l'étiquette de toute arête est réduite à une lettre de D .
- Un prélababyrinthe (Z', E') est un sous-prélababyrinthe *plein* de (Z, E) lorsque $Z' \subseteq Z$ et $E' = \{(P, p, Q) \in E / P, Q \in Z'\}$.

Les *labyrinthes les plus intéressants* seront les sous-labyrinthes plans pleins de $\mathcal{A} = (\mathcal{Z}^2, E)$ où $(P, d, Q) \in E$ si et seulement si $Q = P + \beta(\alpha(d))$ si β est l'homomorphisme qui envoie canoniquement F sur le groupe abélien libre \mathcal{Z}^2 engendré par $(0, 1)$ et $(1, 0)$. Ce sont donc des sous-graphes de la grille carrée classique, et parmi ces derniers, les *dédales co-finis* (mazes) obtenus à partir d'un ensemble fini M de \mathcal{Z}^2 en considérant les points de \mathcal{Z}^2 origines de chemins infinis de la grille classique ne rencontrant pas M .



Labyrinthe "à la Budach". Il faut imaginer que tous les autres points à coordonnées entières sont colorés en noir, que les points blancs sont interdits, et que les arêtes sont les limites des cellules joignant les sommets noirs.

FIG. 3 - *Dédale à la Budach*

Venons en maintenant, et pour finir ce tour d'horizon de structures dans lesquelles vont évoluer des automates, aux définitions très générales de Hemmerling.

3.3 Graphoïdes

3.3.1 Définition

Un graphoïde est un triplet $G = (V, H, I)$ où :

- V est un ensemble non vide d'éléments appelés sommets de G ,

- H est un ensemble disjoint de V , non vide, dont les éléments sont appelés les demi-arêtes de G ,
- I est une relation binaire symétrique et non réflexive sur $V \cup H$, telle que :
 - $I \cap V^2 = \emptyset$,
 - pour tout h de H , il existe exactement un v de V tel que $(h, v) \in I$. On notera alors $v = ver(h)$. De plus, on note $H(v)$ l'ensemble des demi-arêtes incidentes au sommet v .
 - pour tout h de H , il existe au plus un h' de H tel que $(h, h') \in I$. On dénote alors h' par $hal(h)$.

qui est parfois appelée *relation d'incidence*.

Une demi-arête est dite *libre* lorsqu'elle n'est incidente à aucune autre. Deux demi-arêtes incidentes forment une arête. On peut aussi ordonner les demi-arêtes de manière à obtenir des arêtes orientées ou arcs.

Un graphoïde est dit *ouvert* lorsqu'il contient une demi-arête libre. Dans le cas contraire, on parle de graphe. On définit, par analogie avec les graphes, des graphoïdes finis, simples et connexes (un chemin entre deux sommets est nécessairement constitué d'arêtes).

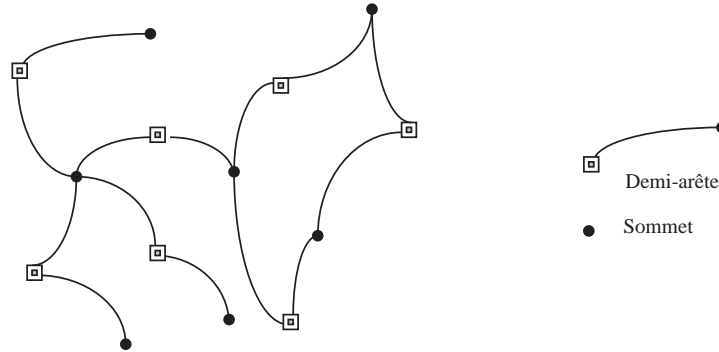


FIG. 4 - *Graphoïdes*

Notons qu'introduire des demi-arêtes a plusieurs avantages : d'une part cela permet de distinguer des sommets par lesquels les automates vont pouvoir en quelque sorte "s'échapper" du graphe sous-jacent, d'autre part cela permet d'envisager un autre type de machines pour lesquelles des marqueurs pourront être déposés sur ces demi-arêtes et non plus (seulement) sur les sommets du graphe sous-jacent, ce qui, dans certaines situations donne des modèles plus puissants.

De plus, un automate n'aura plus nécessairement à se déplacer le long d'une arête du graphoïde mais pourra "sauter" sur une demi-arête d'un sommet adjacent comme l'indique la figure 5.

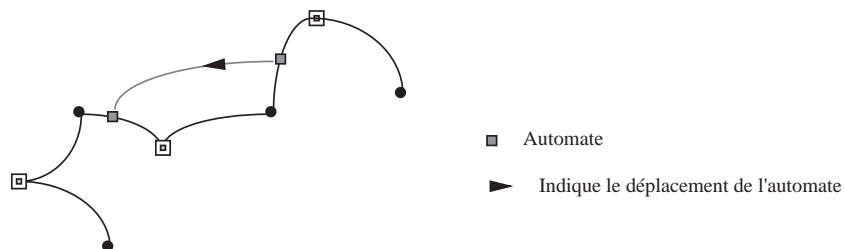


FIG. 5 - Déplacement d'un automate dans un graphoïde

Enfin, cela permet de faire des preuves et d'appliquer de façon rigoureuse certains résultats de géométrie (voir plus loin, par exemple, le théorème de Riemann-Hopf).

Par ailleurs, on pourrait aussi rapprocher cette idée de celle qui consiste à compléter les graphes d'automates afin de pouvoir exploiter des propriétés de graphes réguliers, entre autre des graphes de Cayley, et, à travers eux, la théorie des groupes et une géométrie.

Pour qu'un automate puisse se mouvoir efficacement dans un graphoïde, il doit connaître les demi-arêtes par lesquelles il pourra quitter sa position actuelle. Deux possibilités se présentent : se donner une permutation cyclique pour chaque ensemble $H(v)$ de demi-arêtes admettant v comme sommet adjacent, cela donne naissance aux R -graphoïdes. Ou simplement étiqueter les éléments de tout $H(v)$, ce qui donne lieu aux C -graphoïdes, parmi lesquels on retrouve immédiatement les dédales de la grille \mathbb{Z}^2 classique. C'est ce que nous allons expliciter immédiatement.

Notons auparavant que la définition d'un vecteur de décodage dans les graphes d'automates [RFH72], [Rem92], afin de contrôler les communications entre automates voisins fait précisément des graphes sous-jacents des graphoïdes.

3.3.2 R -graphoïdes

Nous donnons dans ce paragraphe la définition d'un R -graphoïde ainsi que celles de notions nécessaires pour travailler avec ce type de labyrinthe, dont certaines ne sont que des adaptations de notions classiques de géométrie.

- Une rotation ou permutation cyclique d'un ensemble fini non vide S est une bijection r de S sur lui-même telle que $r^{\text{card}(S)}(s) = s$ et $r^i(s) \neq r^j(s)$ pour $1 \leq i < j \leq \text{card}(S)$ et tout $s \in S$.
- Une rotation sur un graphoïde G est une application $r : H \rightarrow H$ telle que, pour tout v de V , $r|_{H(v)}$ est une rotation de $H(v)$.

- Un R -graphoïde est la donnée d'un graphoïde en même temps qu'un système de rotation, c'est donc un quadruplet (V, H, I, r) . Un R -graphe est un R -graphoïde sans demi-arête libre.
- Par *angle* (incident au sommet v) dans un R -graphoïde on entend un chemin $\alpha = (h, v, h')$ où $h \in H, v = \text{ver}(h), h' = r(h)$.
- Deux angles (h_1, v_1, h'_1) et (h_2, v_2, h'_2) sont dits *liés* si $(h'_1, h_2) \in I$ ou $(h_1, h'_2) \in I$.
- Une *face* d'un R -graphoïde est un ensemble minimal non vide d'angles, clos pour la relation de *lien*.
- Un *chemin* $w = (\dots, g_0, g_1, g_2, \dots, g_m, \dots)$ dans un R -graphoïde est dit *suivre une face* lorsque pour trois éléments consécutifs (g_i, g_{i+1}, g_{i+2}) de w où g_{i+1} est un sommet, alors (g_i, g_{i+1}, g_{i+2}) est un angle de cette face.
- L'*ordre* d'un sommet v ou d'une demi-arête h dans un R -graphoïde est le nombre de faces contenant des angles dans lesquels v ou h apparaissent. Plus précisément :

$$\begin{aligned}
 & - \text{ord}(v) = \text{card}\{F / F \text{ est une face contenant un angle incident à } v\}, \\
 & - \text{ord}(h) = \begin{cases} 1 & : \text{ang}(h) = (h, \text{ver}(h), r(h)) \\ & : \text{et } \text{ang}(r_1(h)) = (r_1(h), \text{ver}(h), h) \\ & : \text{appartient à la même face,} \\ 2 & : \text{les angles ci-dessus appartiennent à deux faces} \\ & : \text{différentes} \end{cases}
 \end{aligned}$$

Deux demi-arêtes incidentes ont, par définition, le même ordre, et l'ordre d'une arête (comme d'un arc) est celui de l'une de ses demi-arêtes.

- Lorsque l'on se déplace dans un labyrinthe, il peut être intéressant de connaître quels sont les angles incidents à un sommet v , considérés à partir d'une demi-arête de $H(v)$. On définit alors la relation d'équivalence suivante, appelée *caractéristique de h* , désignée par $\text{cha}(h)$: pour $0 \leq i, j < \text{deg}(v)$, $i \sim_h j$ si et seulement si $\text{ang}(r^i(h))$ et $\text{ang}(r^j(h))$ appartiennent à la même face. Si $\text{cha}(h) = \{\{j / 0 \leq j < \text{deg}(v) \text{ et } j \sim_h i\} / 0 \leq i < \text{deg}(v)\}$, alors $\text{ord}(v) = \text{card}(\text{cha}(h))$ et $\text{ord}(h) = 1$ si et seulement si $0 \sim_h \text{deg}(v) - 1$.
- Dans la suite, un R -ficographe désignera un R -graphe simple fini connexe et un R -incographe désignera un R -graphe simple infini dénombrable connexe.

Remarque 1 Dans le cas fini, cette notion de R -graphoïde est à rapprocher de celle de constellation [Jac70] ou encore de carte combinatoire [Cor84], [Lie89] qui proviennent d'une part de la recherche d'une représentation topologique d'un graphe connexe sur une surface, d'autre part de l'étude systématique des subdivisions d'une surface à partir de cette unique notion de brin.

Rappelons qu'une carte combinatoire est un triplet (B, σ, α) où B est un ensemble fini d'éléments dits brins, α une involution sans point fixe sur B et σ une permutation sur B .

Ces données permettent de définir les arêtes et les sommets de la carte qui sont respectivement les cycles de α et ceux de σ .

Dans le cas des R -graphoïdes, les brins sont évidemment les demi-arêtes, l'involution α est l'application qui à toute demi-arête h associe $h\alpha(h)$, σ est le système de rotation du R -graphoïde dont les cycles caractérisent effectivement les sommets. Une face d'un R -graphoïde peut être vue comme une face sur une surface virtuelle parcourue en empruntant une arête puis en tournant autour du sommet pour prendre l'arête suivante dans la rotation.

Compte tenu du contexte, nous choisissons de conserver la terminologie de R -graphoïde. Cependant ces remarques suggèrent l'étude de la puissance d'automates finis se déplaçant sur différentes surfaces.

Considérons maintenant des graphoïdes "étiquetés" ou encore graphoïdes "à boussoles".

3.3.3 C -graphoïde

Soit D un ensemble fini non vide. Par *boussole* ou *ensemble de directions* D du graphoïde G , on entend une application $c : H \rightarrow D$ telle que la restriction $c|_{H(v)}$ est injective pour tout v . Il en résulte que $\deg(v) \leq \text{card}(D)$.

Un C -graphoïde est alors la donnée d'un graphoïde et d'une boussole, c'est donc un quadruplet (V, H, I, c) . Un C -graphe est un C -graphoïde sans demi-arête libre.

Tout C -graphoïde peut évidemment être considéré comme un R -graphoïde particulier. Certains C -graphoïdes, les $2D$ - et $3D$ -graphoïdes vont jouer naturellement un rôle privilégié. Nous allons les définir précisément, et pour cela revenir sur les notions de plongements.

3.3.4 Plongements de graphes et de graphoïdes

Rappelons d'abord quelques définitions et résultats à propos des graphes. \hat{E}

- Un *plongement de dimension* n , $n \in \{2, 3\}$ (ou n -plongement) d'un graphe $G = (V, E)$ est une application P qui assigne à tout sommet v de G un point $P(v)$ de \mathcal{R}^n , à toute arête e une courbe $P(e)$ dans \mathcal{R}^n , de telle sorte que :

- Si $v \neq v'$, $P(v) \neq P(v')$,

- Pour tout v de V , e de H , $P(v)$ n'est pas un point intérieur de $P(e)$, et $P(v) \in P(e)$ si et seulement si v est une extrémité de e .
- Tout graphe fini ou infini possède un 3-plongement,
- Un graphe qui possède un 2-plongement est dit planaire.

Passons maintenant aux graphoïdes.

- Un plongement de dimension n , $n \in \{2, 3\}$ (ou n -plongement) d'un graphoïde se définit quasiment comme un plongement de graphe, mais il faut cependant arrêter le sort des demi-arêtes.

Précisons donc qu'il s'agit encore d'une application P qui assigne à tout sommet v de G un point $P(v)$ de \mathcal{R}^n , à toute demi-arête h une courbe $P(h)$ dans \mathcal{R}^n , de telle sorte que :

- Si $v \neq v'$, $P(v) \neq P(v')$,
- Pour tout v de V , h de H , $P(v)$ n'est pas un point intérieur de $P(h)$, et $P(v) \in P(h)$ si et seulement si $v = ver(h)$,
- Si $h, h' \in H$ et si $h \neq h'$, alors, $P(h)$ et $P(h')$ n'ont pas de point intérieur commun, et $(P(h) \cap P(h')) \setminus P(V) \neq \emptyset$ si et seulement si $\{h, h'\} \in I$.

Un graphoïde est dit *planaire* s'il possède un 2-plongement. Notons que si G_0 est le graphe obtenu en supprimant toutes les demi-arêtes libres d'un graphoïde G , alors G est planaire si et seulement si G_0 l'est.

Un *plongement plan* d'un R -graphoïde est un 2-plongement du graphoïde sous-jacent avec la propriété supplémentaire que la rotation correspond à l'orientation du plan suivant le sens des aiguilles d'une montre.

Un R -graphoïde est dit *planaire* lorsqu'il possède un plongement plan. On a aussi que G est plan si et seulement si G_0 l'est.

3.3.5 nD -graphoïdes, $n \in \{2, 3\}$

Par $2D$ ou $3D$ -graphoïde, nous entendons des C -graphoïdes avec $D = \{n, e, s, w\}$ et $D \cup \{\text{haut, bas}\}$ comme ensembles de directions respectifs, et possédant respectivement des 2- et 3-plongements dont les arcs $P(h)$ sont des segments de droites parallèles ou perpendiculaires aux axes.

Un nD -plongement est dit *normé* lorsque les arcs $P(h)$ sont des segments de droites de longueur $1/2$ parallèles ou perpendiculaires aux axes. Et il est dit *fortement normé* lorsque $|P(v) - P(v')| = 1$ équivaut au fait que v et v' sont adjacents.

Il est intéressant de rappeler que tout graphe simple planaire fini ou infini a un plongement dans lequel les arêtes sont représentées par des segments de lignes droites. Ce qui se prolonge aux R -graphes.

De plus, un graphe planaire a, au plus, un ensemble dénombrable de sommets de degré 3, ce qui, joint au fait que les triangles pavent le plan, justifie, en Informatique, le recours aux triangulations.

Un chemin rectilinéaire polygonal de \mathcal{R}^2 , RPP , est un ensemble fini ordonné de points p_1, \dots, p_m tels que :

- p_i précède p_{i+1} pour tout $0 \leq i \leq m$,
- les segments $p_i p_{i+1}$ sont parallèles ou perpendiculaires aux axes.

L'index de rotation d'un RPP w est $rin(w)$ défini par :

- $rin(p_0, p_1) = 0$, et
- $rin(p_0, \dots, p_m) = \sum_{i=0}^{m-2} rin(p_i, p_{i+1}, p_{i+2})$ où, pour $i \geq 2$,

$$rin(p_i, p_{i+1}, p_{i+2}) = \begin{cases} 1 & : \quad dir(p_{i+1}p_{i+2}) = r^{-1}(dir(p_i p_{i+1})) \\ 0 & : \quad dir(p_{i+1}p_{i+2}) = dir(p_i p_{i+1}) \\ -1 & : \quad dir(p_{i+1}p_{i+2}) = r(dir(p_i p_{i+1})) \\ -2 & : \quad dir(p_{i+1}p_{i+2}) = r^2(dir(p_i p_{i+1})) \end{cases}$$

où r désigne la bijection définie sur $D = \{e, s, n, w\}$ par $r(e) = s, r(s) = w, r(w) = n, r(n) = e$ et où $dir(pp')$ signifie direction du segment du point p au point p' .

Nous pouvons à partir de ce qui précède décrire une hiérarchie de labyrinthes, et lui faire correspondre parallèlement une hiérarchie de machines. Nous porterons en fait notre attention à la hiérarchie fondamentale suivante :

$$\begin{aligned} R\text{-graphes} &\longrightarrow R\text{-graphes plans} \longrightarrow 2D\text{-graphes} \dots \\ &\dots \longrightarrow 2D\text{-graphes fortement normés.} \end{aligned}$$

Notons le parallèle avec la hiérarchie de Chomsky pour les langages.

4 Des automates sur des motifs, dans des labyrinthes

Un automate fini "en soi" n'est rien sinon un "mécanisme" qui dispose d'un nombre fini d'états (que l'on peut interpréter comme sa mémoire) susceptible d'évoluer dans un système avec lequel il interagit. Cela conduit donc à différentes spécifications plus ou moins formelles, dont un caractère essentiel est, généralement, qu'elles soient finies. Le degré de rigueur des preuves que l'on veut (ou "peut raisonnablement") présenter implique le degré de formalisation consenti.

4.1 Les 4-automates

Rappelons d'abord la définition des 4-automates se déplaçant dans des motifs finis du plan. Il s'agit de machines $\mathcal{A} = (Q, \{0, 1, B\}, \delta, q_0, F)$ où

- Q est l'ensemble fini des états de l'automate,
- 0 et 1 désignent les couleurs des points à coordonnées entières du plan qui déterminent le motif, et B est une troisième couleur permettant de discriminer les points de la frontière,
- δ est la fonction de transition de l'automate,
 $\delta : Q \times \{0, 1, B\} \rightarrow \mathcal{P}(Q) \times \{0, e, n, w, s\}$ dans le cas non déterministe,
 $\delta : Q \times \{0, 1, B\} \rightarrow Q \times \{0, e, n, w, s\}$ dans le cas déterministe, où e, n, w, s désignent les directions habituelles du plan,
- q_0 est l'état initial de l'automate,
- F est une partie de Q , l'ensemble des états d'acceptation de l'automate.

Le fonctionnement de l'automate sur un motif est le suivant : lorsque l'automate est sur un point du motif, dans un état p , et que la couleur du point est x , alors il passe en état q et se déplace, en admettant que la fonction de transition soit définie sur (p, x) , suivant la direction d telle que $(q, d) \in \mathcal{P}(Q) \times \{0, e, n, w, s\}$ si $d \neq 0$, reste sur la même position sinon.

Une *configuration de l'automate \mathcal{A} sur un motif M* est un couple (q, y) où q est un état et y un point du motif.

Un motif M est accepté par \mathcal{A} si et seulement si \mathcal{A} partant de la configuration initiale (q_0, y_0) (où y_0 est le point initial du motif) entre dans un état d'acceptation après un nombre fini de pas de calcul ou mouvements.

Notons que, d'après sa définition, l'automate ne peut se déplacer en dehors de sa frontière.

Nous reviendrons à la reconnaissance de figures du plan, dans un paragraphe propre, en conclusion de ce rapport, afin d'introduire [DM97].

4.2 Automates sur dédales et divers graphoïdes

Nous donnerons maintenant les définitions dans le cadre général des R - et C -graphoïdes car elles pourront alors être assez facilement interprétées pour les cas particuliers habituels. D'abord formellement des machines, puis de leurs tâches : exploration et reconnaissance de graphoïdes.

On considère des automates qui sont, fondamentalement, des automates de Mealy, c'est-à-dire des automates avec sorties, $\mathcal{A} = (Q, X, Y, \delta, \lambda, q_0)$, où Q est l'ensemble fini des états de l'automate, q_0 son état initial, δ sa fonction de transition, qui gouverne ses états, et λ sa fonction de déplacement, qui détermine son mouvement dans le graphoïde.

δ est donc une fonction de $Q \times X$ dans Q , pas nécessairement partout définie, et λ une fonction de $Q \times X$ dans Y , pas nécessairement partout définie non plus. Ces automates n'ont pas d'états d'acceptation ou de rejet, ce qui est cohérent avec l'objectif qu'on leur donne a priori: explorer des labyrinthes. Nous n'introduirons de tels états que lorsqu'il s'agira de reconnaître ou de décider des classes d'objets.

1. *R*-automates ou automates sur un *R*-graphoïde .

Nous ne nous préoccupons que de *R*-graphoïdes dont les sommets sont de degré fini et borné par un entier b , $b \geq 3$. En effet, les cas où cette borne est inférieure à 2 sont triviaux. Ici, $X = \{1, \dots, b\}$, $Y = \{\varsigma, 0, \dots, b-1\}$, où ς est un symbole qui indique en quelque sorte les limites de la structure. L'automate, dans un certain état, est sur une demi-arête h et "voit" les demi-arêtes adjacentes au sommet $ver(h)$.

Une configuration de l'automate est donc un couple (q, h) d'un état et d'une demi-arête.

L'évolution de l'automate est décrite par une suite de configurations qui se succèdent via le relation de dérivation \vdash , de la façon suivante :

Si $\delta(q, deg(ver(h))) = q'$ et $\lambda(q, deg(ver(h))) = j$, alors,

$(q, h) \vdash (q', h')$ où $h' = h$ si $j = \varsigma$, $h' = hal(r^j(h))$ sinon.

Cela signifie donc que l'automate, dans un certain état, sur une demi-arête, soit reste en place, tout en changeant d'état, soit se déplace, en changeant d'état, sur une demi-arête d'un sommet voisin si la demi-arête $r^j(h)$ est bornée, est dit *quitter* le graphoïde sinon.

On admettra que le système se bloque si l'une ou l'autre des deux fonctions n'est pas définie pour $(q, ver(h))$.

2. *C*-automates ou automates sur un *C*-graphoïde

Dans ce cas, on a un ensemble de directions D . Alors,

$X = \{(S, d)/S \subseteq D, d \in S\}$, $Y = D \cup \{\varsigma\}$ et $\lambda(q, (S, d)) \in S \cup \{\varsigma\}$ pour tout $(q, (S, d)) \in Q \times X$.

Si $\delta(q, (c(H(ver(h)), c(h)))) = q'$ et si $\lambda(q, (c(H(ver(h)), c(h)))) = d'$, ou bien $d' = \varsigma$ et alors l'automate reste sur la même demi-arête, ou bien $d' \neq \varsigma$ et alors il existe une demi-arête $h_1 \in H(ver(h))$ telle que $d' = c(h_1)$; deux possibilités encore, ou bien h_1 est libre, alors l'automate est dit *sortir* du graphoïde par cette demi-arête, ou bien elle est bornée et alors l'automate va se placer sur $h' = hal(h_1)$.

3. Comportement d'un automate à partir d'une demi-arête

Les positions de l'automate sont des demi-arêtes du graphoïde. L'automate ne se déplace pas nécessairement le long d'arêtes. Son comportement se représente par la suite d'états et de positions $(q_i, h_i)_{i \geq 0}$ qu'il a à partir d'une configuration initiale (q_0, h_0) , où q_0 est précisément l'état initial de l'automate.

4. Expressions du comportement dans les cas particuliers classiques

– Automates dans les “dédalles à la Blum” \hat{E}

Dans cette sorte de dédale, l’automate se déplace de cellule blanche en cellule blanche (les autres, noires, sont interdites). On se contente d’une description très informelle des chemins suivis, avec, parfois un peu d’équilibre sur les limites! Les résultats ne sont pas en cause, d’ailleurs ils sont rigoureusement prouvés par Hemmerling, par exemple!

– Automates dans les “dédalles à la Budach”

En interprétant le labyrinthe comme un automate lui-même, Budach donne une expression de ce comportement en termes de jeu, puis, dans un cas particulier, en termes de système dynamique. Voyons comment.

– Automates

On désigne par $P'(D)$ l’ensemble des parties non vides de D . Ici, $X = Y = P'(D)$,
 $\delta : Q \times P'(D) \longrightarrow Q$ et $\lambda : Q \times P'(D) \longrightarrow D$, avec $\lambda(q, S) \in S$ pour tout q , tout S .

On peut interpréter tout labyrinthe comme un automate avec Z comme ensemble d’états (possiblement infini donc), D comme ensemble X d’entrées, $P'(D)$ comme ensemble Y de sorties, et des fonctions de transitions définies respectivement par $\delta(P, r) = Pr$, le sommet adjacent au sommet P dans la direction r , et $\lambda(P, r) = val(P)$. Notons que cette dernière fonction ne dépend pas de la direction mais seulement du point sur lequel se trouve l’automate.

– Mouvement d’un automate dans un labyrinthe

Le comportement de l’automate \mathcal{A} dans le labyrinthe \mathcal{L} peut être vu comme un jeu du premier automate dans le second, et interprété comme un système dynamique. Précisons.

– Soit \mathcal{A} . On signifiera par $X \xrightarrow{\mathcal{A}} Y$ que l’ensemble des entrées de \mathcal{A} est X et son ensemble de sorties Y .

Soit \mathcal{A} et \mathcal{A}' deux automates, $X \xrightarrow{\mathcal{A}} Y$ et $Y \xrightarrow{\mathcal{A}'} Y'$.

On appelle jeu de \mathcal{A} dans \mathcal{A}' l’automate $X \xrightarrow{\mathcal{A} \times \mathcal{A}'} Y'$ dont l’ensemble des états est $Q \times Q'$ et les fonctions de transitions sont définies par :

$$\begin{aligned} \bar{\delta}((q, q'), x) &= (\delta(q, x), \delta'(q', \lambda(q, x))) \\ \bar{\lambda}((q, q'), x) &= \lambda'(q', \lambda(q, x)). \end{aligned}$$

– Dans le cas où $Y' = X$ et où λ' ne dépend que de l’état de l’automate, ce jeu entre les deux automates peut être vu comme un système dynamique $\mathcal{S} = (Q \times Q', g)$, g étant déterminée comme suit :

on peut écrire λ' comme une fonction μ' de Q' dans X : $\mu'(q') = \lambda'(q', y)$. Ceci donne la fonction $\bar{\mu}$ de $Q \times Q'$ dans X définie par : $\bar{\mu}((q, q')) = \mu'(q')$, d'où $\bar{\lambda}((q, q'), x) = \bar{\mu}((q, q'))$. Alors, on pose :

$$g((q, q')) = (\bar{\delta}(q, q'), \bar{\mu}((q, q'))), \text{ soit, en explicitant :}$$

$$g((q, q')) = (\delta(q, \mu'(q')), \delta'(q', \lambda(q, \mu'(q')))).$$

- Explicitons ce que cela donne dans le cas du jeu de \mathcal{A} dans le labyrinthe \mathcal{L} qui permet effectivement cette interprétation comme système dynamique $\mathcal{S}(\mathcal{A}, \mathcal{L}) = (Q \times Z, g)$:

$$g((q, P)) = (\delta(q, valP), \delta'(P, \lambda(q, valP))), \text{ soit,}$$

$$g((q, P)) = (\delta(q, valP), P\lambda(q, valP)).$$

ce que l'on peut représenter par :

$$q \xrightarrow{valP} \delta((q, valP)) \text{ dans } \mathcal{A},$$

$$P \xrightarrow{\lambda((q, valP))} P\lambda((q, valP)) \text{ dans } \mathcal{L}.$$

On s'intéresse aux chemins de l'automate dans le labyrinthe à partir d'un point initial O , l'automate étant au départ dans son état initial, c'est-à-dire aux suites $(g^t((q_0, O)))_{t \geq 0}$.

Ces chemins peuvent être représentés par :

$$P_0 = O \xrightarrow{r_0} P_1 \xrightarrow{r_1} P_1 \xrightarrow{r_1} \dots P_n \xrightarrow{r_n} \dots$$

La suite $(q_t, val(P_t), \lambda((q_t, val(P_t))))_t$ est appelée le \mathcal{A} -modèle de \mathcal{L} lorsque \mathcal{A} part de (q, O) . Si (\mathcal{L}, O) est un labyrinthe pointé (ou labyrinthe dans lequel on a distingué un point), le \mathcal{A} -modèle de (\mathcal{L}, O) est le \mathcal{A} -modèle de \mathcal{L} lorsque \mathcal{A} part de (q_0, O) .

5. Exploration, fuite, maîtrise et reconnaissance d'un labyrinthe par un automate fini

- \mathcal{A} est dit *explorer* un R - ou C -graphoïde L lorsque pour toute position de départ h et tout sommet v le comportement correspondant contient une position incidente à v .
- \mathcal{A} est dit *s'échapper de* L lorsque son comportement est fini quelle que soit sa position de départ, c'est-à-dire que quelle que soit cette position initiale, l'automate arrive sur une demi-arête libre.
- \mathcal{A} est dit *maîtriser* L lorsque, quelle que soit la configuration initiale, le comportement de l'automate est une suite infinie.
- *Un automate* pour lequel on a, de plus, spécifié un ensemble d'états d'acceptation est dit *reconnaître un labyrinthe pointé* lorsque partant, dans son état initial, du point désigné, il entre au bout d'un temps fini dans un état d'acceptation.

On dit qu'un tel *automate reconnaît une classe de labyrinthes pointés*

s'il en reconnaît tout objet, et qu'il *décide une telle classe* lorsqu'il reconnaît tout objet de la classe et rejette tout objet qui n'est pas dans cette classe en entrant dans un état d'un ensemble d'états de rejet qui aura été spécifié.

Un problème d'automates et de labyrinthes nécessite que soient précisées une classe d'automates, une classe de labyrinthes et une tâche à accomplir par un tel automate dans un tel labyrinthe.

6. Piège pour un automate

Soient une classe \mathcal{L} de labyrinthes et une classe \mathcal{A} d'automates dont la tâche est d'explorer les labyrinthes de \mathcal{L} . Un *piège* pour un automate A de \mathcal{A} est un couple (L, h) où L est un élément de \mathcal{L} , h une demi-arête de L telle que A ne puisse explorer L à partir de la position que détermine h .

Nous ne faisons que mentionner ici le problème de l'arrêt des machines, que nous examinerons au cours du texte chaque fois qu'il se posera, et dont il faut noter qu'il est essentiel dans les stratégies de décision.

Quelle est la puissance de tels automates dans les structures évoquées jusque-là? Nous allons voir que c'est à la fois beaucoup et peu !

4.3 Sur des performances d'automates finis dans des labyrinthes

Dans ce paragraphe, nous énonçons et prouvons quelques résultats illustrant ce que peuvent et ne peuvent pas réaliser des automates finis dans des labyrinthes, en mettant en évidence des méthodes de preuves classiques dans le domaine.

4.3.1 Reconnaissance et ou décision

1. Considérons d'abord les motifs de Blum et Hewitt (modèles BH) : ce sont des carrés finis de carrés-unité (ou cellules) du plan, dont la frontière est constituée de cellules marquées par un caractère particulier, B par exemple, les cellules du motif proprement dit étant colorées en blanc et noir, ou 0 et 1, comme le montre la figure 6. La cellule initiale d'un tel motif est la cellule correspondant à son coin nord ouest.

Un 4-automate est capable de décider les motifs qui contiennent un seul rectangle d'une couleur dont les bords sont parallèles à ceux du carré. Même

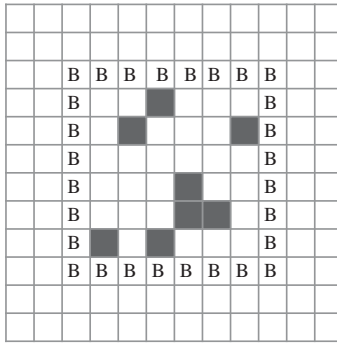


FIG. 6 - Motif de Blum et Hewitt

chose en remplaçant rectangle par carré. Mais, [BH67],

Théorème 1 *Un 4-automate déterministe ne peut pas, en général, décider si un motif fini carré de côté de longueur impaire contient un 1 sur sa cellule centrale.*

La preuve de ce théorème repose sur une idée assez efficace pour ce genre de résultats : la mise en évidence de motifs particuliers, équivalents pour un automate (c'est-à-dire indistingables par lui), qui permettent d'exhiber des situations dans lequel l'automate est en défaut.

Dans ce cas, il s'agit de motifs carrés, de même dimension, sans cellules frontières c'est-à-dire dont les cellules sont colorées par 0 ou 1, disons-les *sous-motifs distingués*. Deux tels motifs sont indistingables par \mathcal{A} si chaque fois que \mathcal{A} entre dans l'un par la cellule x_i , en état q_i et en ressort par la cellule x_j en l'état q_j , il fait de même sur l'autre, c'est-à-dire que s'il y entre par x_i , en état q_i , il en ressort par x_j en l'état q_j (les cellules du bord étant numérotées de la même façon). On définit ainsi une relation d'équivalence sur ce type de motifs.

Venons-en à la preuve du théorème.

Preuve Supposons qu'il existe un 4-automate \mathcal{A} qui décide la classe des motifs BH de longueur de côté impaire dont la cellule centrale est colorée par 1. On peut supposer que l'automate revient sur sa cellule initiale pour s'arrêter.

Un sous-motif distingué de longueur de côté x a $4x$ cellules le long de son périmètre. Par conséquent, \mathcal{A} , à n états, a $4nx$ possibilités de sortir d'un tel sous-motif, plus une de ne pas en sortir. Mais comme il a aussi $4nx$ possibilités d'entrer dans un tel motif, on en déduit que le nombre de classes d'équivalence de sous-motifs distingués de longueur de côté x est au plus $(4xn + 1)^{4xn}$. Cependant le nombre de ces motifs est 2^{x^2} . Donc,

puisque $2^{x^2} > (4xn + 1)^{4x^n}$ pour x suffisamment plus grand que n , il existe certainement une classe qui contient deux éléments distingués m_1 et m_2 distincts, c'est-à-dire qui diffèrent au moins par la couleur d'une de leurs cellules, disons c .

Considérons alors deux motifs de longueur de côté impaire suffisamment grande par rapport à n , ne différant que par leurs sous-motifs m_1 et m_2 disposés de telle sorte que c soit la cellule centrale de chacun d'eux. \mathcal{A} devrait accepter l'un et rejeter l'autre, mais il ne peut pas les distinguer ! Absurde. \square

Cependant il est assez clair qu'un 4-automate non déterministe décide la classe en question : il suffit qu'il se déplace sur la première diagonale jusqu'à entrer dans une cellule colorée par 1. L'automate teste alors s'il aboutit à la cellule du coin inférieur gauche du motif en descendant le long de la diagonale convenable. On en déduit immédiatement :

Corollaire 1 *Les 4-automates non déterministes sur les motifs BH sont plus puissants que les 4-automates déterministes.*

Cependant ils ne sont pas encore si puissants, comme nous allons le voir maintenant.

2. Nous considérons dans ce paragraphe des motifs qui sont plus généraux que ceux de Blum et Hewitt. Il s'agit d'ensembles finis de points de \mathcal{Z}^2 , colorés en blanc (1) et noir (0), entourés de frontières formées de points colorés en B qui sont précisément les points qui ne sont ni blancs, ni noirs, mais qui ont un voisin blanc ou noir suivant les quatre directions usuelles ou une diagonale (8-voisin). Ce sont des motifs qui peuvent donc avoir des trous, comme les rectangles "creux" représentés dans la figure ?? . On dénote Rcr la classe de ces rectangles dont le point initial est le point noir du coin en haut, à gauche.

Théorème 2 *Il n'existe pas de 4-automate non déterministe reconnaissant la classe Rcr .*

Une preuve de ce théorème donnée dans [Myl72] utilise encore l'idée de sous-motifs distingués équivalents dans lequel l'automate est mis en défaut.

Preuve Supposons qu'un 4-automate $\mathcal{A} = (Q, \{0, 1, B\}, \delta, q_0, F)$ non déterministe reconnaît Rcr . Considérons des sous-motifs d'objets de Rcr constitués d'une ligne de points 0, horizontale ou verticale, bordée de deux lignes de même longueur de points B , comme le montre la figure 8, dont les seuls points d'entrée ou de sortie sont les points des extrémités numérotées 1, 2, 3, 4, 5, 6.

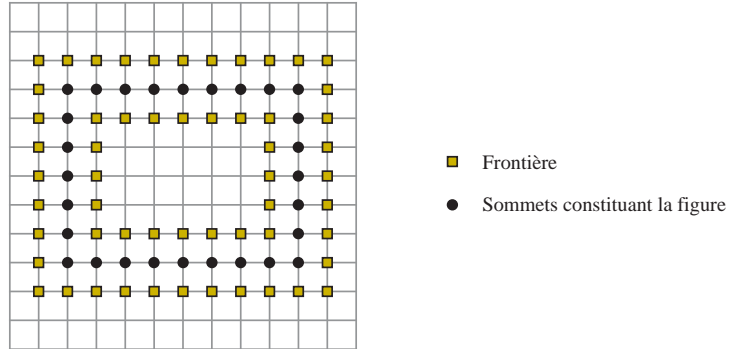


FIG. 7 - *Rectangles creux*

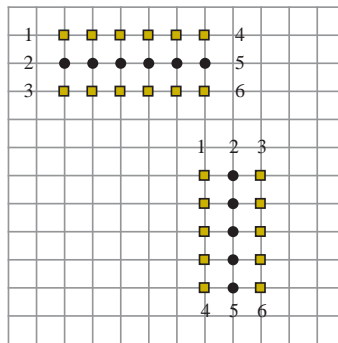


FIG. 8 - *Motifs distingués*

A chacun de ces motifs w est associée une fonction $\Delta_{w, \mathcal{A}}$ de $Q \times \{1, 2, 3, 4, 5, 6\}$ dans $\mathcal{P}(Q \times \{1, 2, 3, 4, 5, 6\}) \cup \{0\}$, déterminée comme suit : si \mathcal{A} entre par le point i et en état q dans w , ou bien il ne sort pas de w , alors $\Delta_{w, \mathcal{A}}(q, i) = 0$, ou bien il en sort par un point j , en état p , alors $\Delta_{w, \mathcal{A}}(q, i)$ est l'ensemble des couples (p, j) possibles (suivant δ et w). La relation E définie sur la classe des motifs distingués par $w_1 E w_2$ si et seulement si $\Delta_{w_1, \mathcal{A}} = \Delta_{w_2, \mathcal{A}}$ est une relation d'équivalence. D'après la définition de E , le nombre de ses classes est évidemment fini, par conséquent il existe au moins une classe ayant une infinité d'éléments, soit w un de ses éléments.

On peut supposer que w est un motif horizontal. Soit w' le motif distingué vertical de même longueur qui apparaît dans le carré c de Rcr représenté sur la figure 9.

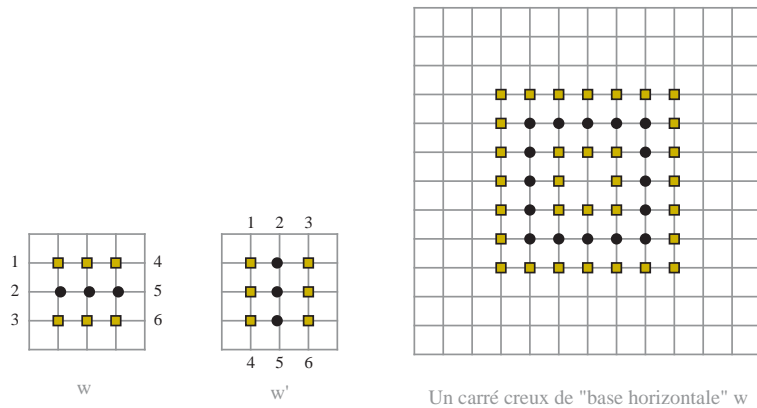


FIG. 9 - Les motifs distingués w et w' , le carré c

Soit w et w' sont dans la même classe infinie C , soit non. Mais alors w' appartient aussi à une classe infinie. En effet, comme le carré creux est reconnu par \mathcal{A} , l'automate, au cours de son test, doit entrer et sortir de chaque sous-motif w et w' . Comme l'automate est spécifié de façon finie, au bout d'un certain nombre de pas, son comportement devient périodique, par conséquent, pour peu que la longueur de w' soit suffisante (ce qui est possible car elle ne dépend que du choix de w qui lui-même appartient à une classe contenant une infinité de motifs de longueurs distinctes) il existe une infinité de motifs distingués verticaux équivalents à w' , et même plus précisément un motif de longueur n pour tout $n \geq |w'|$. Nous sommes maintenant en mesure de montrer que nous aboutissons à une situation absurde en exhibant un motif qui n'est pas dans Rcr et qui est reconnu par \mathcal{A} .

Supposons que \mathcal{A} accomplisse au moins m traversées du carré pour le reconnaître. Et considérons la "spirale" obtenue en remplaçant convena-

blement chaque occurrence de w et w' par un élément équivalent strictement plus long convenable, de telle sorte que l'on obtienne $2m + 3$ coins inférieurs gauche. On prend comme point initial du motif le $m + 1$ -ième coin inférieur gauche. Il est clair que l'automate reconnaît ce motif qui n'est pas dans Rc , de même que tout motif obtenu en complétant de façon absolument quelconque les extrémités de notre "spirale". \square

Notons que la technique qui consiste à construire une spirale qu'un automate ne peut explorer est souvent utilisée. Elle permet par exemple de prouver le résultat suivant ([Hem89]):

Théorème 3 *Il n'existe pas de C -automate qui explore tout 2D-graphe fini connexe fortement normé ayant au plus deux faces, et qui s'arrête toujours sur ce type de labyrinthes.*

4.3.2 Plaçons-nous maintenant dans le cadre de l'exploration de dédales à la Blum

Proposition 2 *Un 4-automate déterministe peut explorer tout dédale fini simplement connexe, c'est-à-dire dont le seul obstacle est la frontière. Mais l'automate ne s'arrête pas.*

L'idée de l'algorithme est de suivre l'obstacle à main droite, tout en explorant, avant chaque nouveau pas le long de la frontière, toutes les cellules blanches possibles vers le nord. Il est représenté sur la figure 10.

Notons que l'automate réalisant cet algorithme, que nous n'écrivons pas, a un nombre d'états assez important !

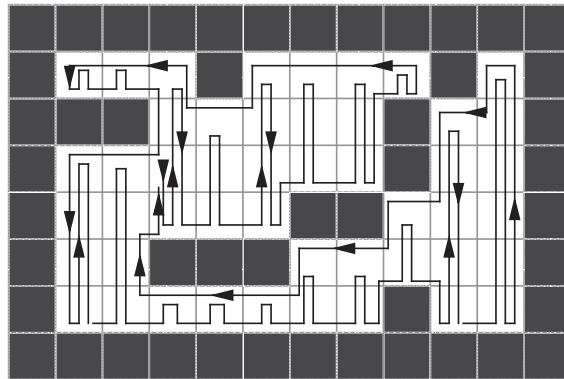


FIG. 10 - 4-automate explorant un dédale simplement connexe à la Blum

4.3.3 Dans le cadre des R -graphoïdes

Hemmerling obtient un résultat analogue et plus général :

Proposition 3 *Il existe un R -automate déterministe qui explore tout R -ficographe de degré borné à une seule face. Mais le R -automate ne s'arrête pas.*

L'algorithme ici est exactement suivre le graphe à main droite. Ce que l'on peut décrire plus précisément : l'automate est placé initialement sur n'importe quelle demi-arête h du ficographe, il se déplace dès lors systématiquement vers la position $hal(r(h))$ de sa position courante h , comme le montre la 11. On remarque que l'on en déduit le résultat pour les dédales finis M à la Blum en appliquant l'algorithme précédent au graphe réduit $red(G(M))$ qui a bien une seule face. On remarque aussi que ces algorithmes peuvent être vus comme des versions particulières de l'algorithme de Terry [RFH72].

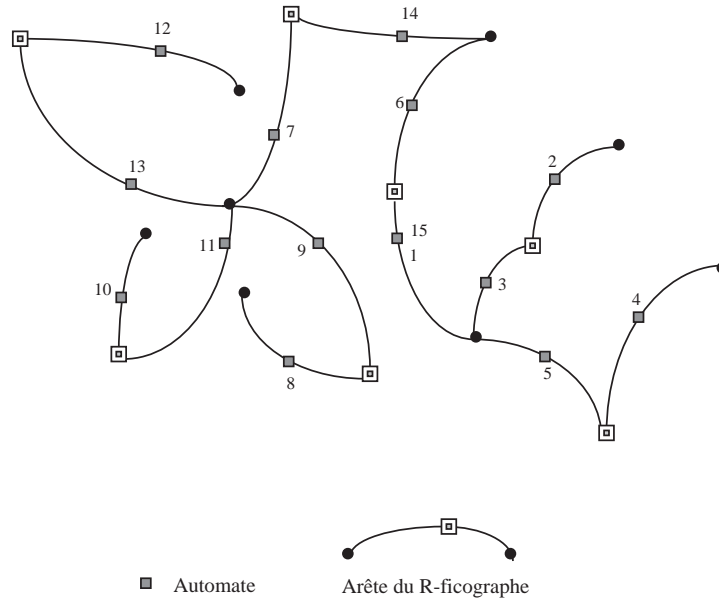


FIG. 11 - Exploration d'un R -ficographe à une seule face

Citons encore ici un résultat de Hemmerling [Hem87a], un des rares qui donnent une évaluation du nombre d'états de l'automate dont il affirme l'existence. Il nécessite d'introduire la notion d'index de rotation pour un $2D$ -ficographe \mathcal{L} : $rot(\mathcal{L}) = \max\{|rin(w)| / w = (v_0, \dots, v_m), 1 \leq m, \text{ est un chemin suivant une face, avec } (v_i, v_{i+1}) \neq (v_j, v_{j+1}), 0 \leq i < j < m\}$, et réalise un algorithme d'exploration des $2D$ -ficographe quelconques fondé sur la considération assez

sophistiquée de chemins particuliers (chemins balancés réguliers ou “regular swings”).

Proposition 4 *Pour tout entier $k \in \mathcal{N}$, Il existe un automate fini avec seulement $O(k)$ états, qui explore tout 2D-ficographe \mathcal{L} dont l'index de rotation satisfait $\text{rot}(\mathcal{L}) \leq k$ et qui s'arrête après $O(n^2 \cdot k)$ pas.*

4.3.4 Exploration du monde: cas particulier d'un graphe infini

Proposition 5 *Un 4-automate ne saurait explorer le plan \mathbb{Z}^2 monocolore, dit monde vide.*

Preuve Nous considérons donc comme graphe la grille carrée classique, ou encore le C -graphe infini, $C = \{n, e, s, w\}$, fortement normé dont les sommets sont les points à coordonnées entières du plan euclidien, les arêtes les segments horizontaux et verticaux qui les joignent.

L'idée de la preuve est simple : comme l'automate a un nombre fini d'états, qu'il ne peut prendre qu'un nombre fini de directions, qu'il n'y a pas d'information particulière sur les sommets, son cheminement devient nécessairement périodique après un nombre fini de pas qui dépend précisément de son nombre d'états. En conséquence il va se déplacer dans une bande de largeur finie du plan, laissant une infinité de points à l'extérieur.

Soyons un peu plus précis. Le comportement de l'automate est en fait caractérisé par la suite infinie $(q_i)_{i \geq 0}$ de ses états successifs. Comme $|Q|$ est fini, il existe deux entiers distincts i_1 et i_2 tels que $q_{i_1} = q_{i_2}$ et $q_{i_j} \neq q_{i_1}$ pour tout $i_1 \leq j < i_2$.

Par ailleurs, à partir d'une position, l'automate ne peut se déplacer, en prenant des états distincts, qu'à une distance au plus $|Q|$ le long d'une direction. Il en résulte que l'automate ou bien boucle dans un carré de côté au plus $2|Q|$, ou bien se déplace dans une bande infinie de largeur au plus $2|Q|$, dont la direction est donnée par le vecteur joignant les positions points de départ de deux périodes consécutives. Dans chacun des cas l'automate laisse une infinité de points hors de sa trajectoire. Voir la figure 12. \square

Il résulte de cette preuve que pour tout automate fini, il existe une partie finie du plan discret qui le piège. L'intérêt est donc de chercher s'il existe des pièges ayant des propriétés intéressantes. Citons pour exemples quelques résultats dont des preuves peuvent être trouvées, par exemple, dans [Hem89].

Théorème 4 *Pour tout R -automate, il existe un piège (L, h) où L est un R -ficographe plan cubique, dont chaque face consiste en au plus six angles.*

Théorème 5 *Il n'existe pas de C -automate qui explore tout 2D-ficographe fortement normé et qui s'arrête toujours sur de tels labyrinthes.*

Théorème 6 *Pour tout C -automate fini dont l'ensemble de directions D contient $\{e, s, w, n\}$ il existe un piège (L, h) tel que L est un 2D-ficographe ayant*

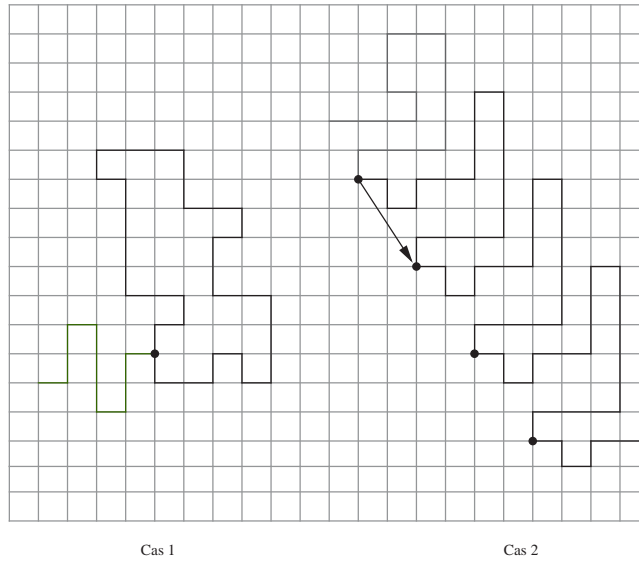


FIG. 12 - Cheminement d'un automate fini dans le monde vide

au plus trois faces et $2^{O(\sqrt{n \cdot \ln(n)})}$ sommets, où n est le nombre d'états de l'automate.

Ces résultats sont essentiellement dus à Budach qui donna la première preuve (très sophistiquée) [Bud78] de l'existence de "dédalles à la Budach" cofinis piégeant un automate fini. Les améliorations de son résultat sont essentiellement dues à Müller [Mül79] pour le nombre de faces et Antelmann [Ant80] pour le nombre de sommets. De ces résultats, dont les preuves sont plus ou moins compliquées, on peut déduire que les automates finis sur des structures de dimension supérieure ou égale à 2 ne sont pas très puissants.

Pourtant d'autres résultats semblent mettre en lumière le contraire. Par exemple le suivant, évoqué dans [BS77], que l'on pourrait traduire brutalement par le fait qu'un automate fini dans un tel plan est universel pour le calcul !

Théorème 7 *Un automate fini dans le plan de cellules blanches, structuré par deux lignes infinies rationnelles de cellules noires qui se coupent (ou système affine de coordonnées) peut simuler une machine de Turing universelle.*

En effet, on sait que toute machine de Turing peut être simulée par une machine à deux compteurs [Fis66], [HU79], [Min67]. Les distances de l'automate aux axes représentent les compteurs. L'évolution de l'automate dans le quart de plan permet de diminuer ou d'augmenter les compteurs, et les axes permettent de décider si un compteur est vide. □

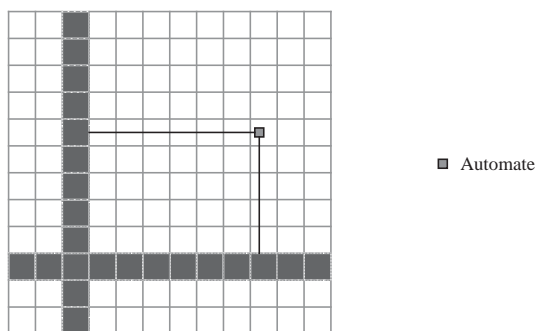


FIG. 13 - *Un automate sur le plan coloré de deux axes*

En fait, ces quelques remarques mettent en évidence la prédominance de la structure sur l'automate. Cependant, on peut améliorer les performances de ces derniers sur des plans colorés plus ou moins complexes ou dans des structures plus générales, en leur "ajoutant de la mémoire" (leur mémoire initiale consistant en leurs seuls états). C'est ce qui conduit, entre autres, aux automates à galets.

5 Automates à galets et quelques autres modèles

Ajouter de la mémoire à un automate fini se déplaçant sur une structure dans l'espace peut se réaliser de bien des manières. Cela peut être la possibilité de poser, reconnaître et déplacer des points de repères, des marqueurs sur la structure en question, mais aussi celle de diriger l'automate dans la structure suivant un calcul qui est effectué par ailleurs, par exemple sur le ruban d'une machine de Turing, ce qui diffère totalement, dans ce dernier cas des machines de Turing sur un plan. Nous ne nous intéresserons vraiment qu'à la première manière, les marqueurs introduits étant soit les galets dans les cellules des labyrinthes ou plus généralement sur les sommets des graphes, soit des pointeurs (terminologie de Hemmerling qui introduit ce modèle) sur les demi-arêtes des graphoïdes.

1. Automate à marqueurs :
galets sur les sommets, pointeurs sur les demi-arêtes du graphoïde sous-jacent. L'automate sait reconnaître s'il porte un marqueur (soit sur sa position, soit sur le sommet incident à cette position). En principe un élément du graphoïde ne porte qu'un seul marqueur éventuel. L'automate peut déposer ou reprendre un marqueur. En configuration initiale le graphoïde ne porte pas de marqueurs.

Le comportement d'un tel automate est décrit par une suite de triplets constitués de l'état, de la position et de l'existence ou non d'un marqueur à cette position.

Remarque 2 *Tout automate à k galets peut être simulé par un automate à k pointeurs. Mais la réciproque n'est pas vraie !*

2. D'autres modèles :

Nous signalons pour finir quelques autres types de machines sans références à leurs performances autres que le tableau final, issu de [Hem89] auquel le lecteur intéressé peut se reporter pour plus de connaissances.

– Systèmes de k automates coopérants

Ils peuvent être considérés comme une généralisation des automates à pointeurs. Ce sont k automates finis qui travaillent simultanément sur le même graphoïde.

Chaque automate du système sait quels sont les automates qui ont la même position que lui. En configuration initiale, tous les automates sont à la même position.

Un tel système explore un graphoïde si tout sommet est visité par au moins un automate, ceci quelle que soit la position de départ. Le système s'arrête dès qu'un automate s'échappe.

– Automates multi-têtes

Une coopération plus riche peut être assurée par des automates à k -têtes, qui peuvent être vus comme des systèmes de k -automates (les têtes) coopérant avec la possibilité d'échanger leurs informations d'entrée via un centre de contrôle.

Un type d'automates encore plus puissants est obtenu en donnant la possibilité à chaque automate d'un tel système de rejoindre (en une transition) - "sauter" à - la position de n'importe quel autre tête.

Une autre philosophie conduit à des modèles d'un type différent, dont nous donnons quelques représentants ci-dessous.

– Automate à un ruban de Turing : C'est un automate fini auquel on adjoint un alphabet de travail X_T contenant un symbole spécial, le "blanc", les fonctions δ et λ sont alors décrites par :

$$\delta : X \times X_T \times Q \longrightarrow X_T \times \{1, -1, \varsigma\}, \text{ et,}$$

$$\lambda : X \times X_T \times Q \longrightarrow Y \text{ avec } \lambda((S, d), x_T, q) \in S \cup \{\varsigma\}.$$

Une configuration d'un tel automate à ruban de Turing est une suite de quadruplets (h, q, w_1, w_2) où w_1 et w_2 sont les mots du ruban de Turing, de part et d'autre de la tête (avec la convention que cette dernière pointe la première lettre de w_2).

- Automate à pile: c'est un automate à ruban de Turing, dont le ruban est une pile.
- Automate à "pile visitable": c'est un automate à pile, avec cette propriété supplémentaire que l'on peut aller lire (sans écrire) dans la pile.
- Automate à compteur: c'est un automate à pile avec exactement trois symboles dans son alphabet de travail: le blanc, le symbole de fond de pile et un seul autre symbole 1. La pile d'un tel automate représente un nombre naturel. L'automate sait reconnaître si sa pile est vide ou non.

3. Des liens entre ces différents types d'automates

Dans le schéma de la figure 14, les flèches $\mathcal{A} \rightarrow \mathcal{B}$ signifient que les automates de la classe \mathcal{A} sont au moins aussi puissants que ceux de la classe \mathcal{B} .

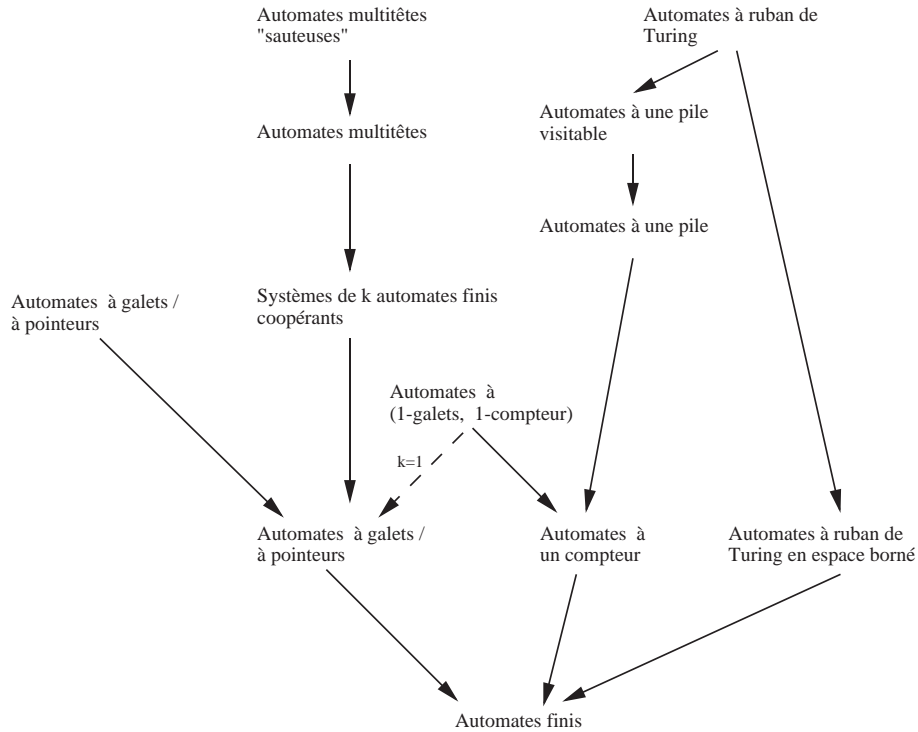


FIG. 14 - Puissance comparée des différents types d'automates évoqués

Voyons maintenant quelques performances et contre-performances des automates à marqueurs, spécialement des automates à galets.

6 Des automates à galets ou à pointeurs dans des labyrinthes

6.1 Le monde vide

Commençons par le labyrinthe très particulier qu'est le monde vide.

Théorème 8 *Il existe un automate à trois galets qui explore le monde vide.*

Preuve Un algorithme sur lequel est fondé ce résultat est représenté par la figure 15. Les trois galets que possède initialement l'automate sont successivement déposés suivant un triangle, les cheminements de l'automate sont représentés par les flèches. A chaque tour, chaque fois qu'il rencontre un galet, l'automate le déplace vers une nouvelle position suivant la direction (retenue dans ses états) dans laquelle il arrive sur le galet. \square

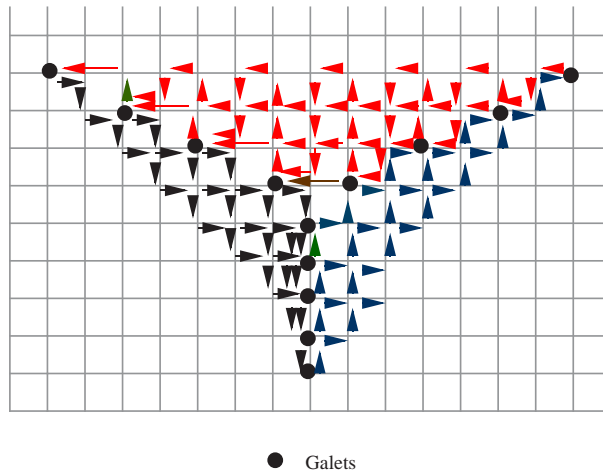


FIG. 15 - *Exploration du monde vide par un automate à trois galets*

Remarque 3 *Il est intéressant de noter immédiatement que cette exploration du monde, qui, bien sûr, n'est pas unique, définit une bijection de \mathbb{Z}^2 sur \mathcal{N} , et par conséquent un ordre sur les éléments de \mathbb{Z}^2 pour peu que l'on aie choisi une origine. Nous utilisons très fortement une telle bijection dans [DM97].*

Remarque 4 *Jusqu'à présent, les galets considérés sont indifférenciés. On peut justement se demander si leur donner une étiquette ajoute de la puissance à l'automate ainsi obtenu. On démontre que, suivant les structures sur lesquelles se*

déplacent les automates, ces automates à galets étiquetés ont la même puissance que les automates à galets non étiquetés [BH67], [Hem89], [DM97]. Mais c'est faux ou encore inconnu sur d'autres.

Dans le cas du monde vide, nous pouvons encore montrer que :

Proposition 6 *Un automate à un galet, ni un automate à deux galets ne sauraient explorer le monde vide.*

Preuve Considérons en effet un automate à un galet g . Il part d'une position de la grille avec son galet. Il doit le déposer avant un certain temps fini t_0 sous peine de se comporter comme un automate sans galet, dont on a déjà vu qu'il ne reconnaît pas le monde. Mais il doit ensuite revenir chercher ce galet avant un temps fini t_1 pour la même raison. Et, comme l'automate et son galet sont spécifiés de façon finie, la suite qui décrit son comportement (chaque élément de cette suite représentant un état du système, c'est-à-dire la position de l'automate et l'information qu'il porte ou non son galet) est nécessairement périodique. Par conséquent si la région du plan qu'il sillonne n'est pas bornée, c'est encore une fois, à partir d'un certain point une bande du plan de largeur finie et fonction du nombre d'états de l'automate. Il en résulte que l'automate laisse une infinité de points hors de sa trajectoire. La figure 16 illustre ce raisonnement. \hat{E}

Supposons maintenant que l'automate a deux galets g_1 et g_2 . Il part d'un point de \mathcal{Z}^2 , en possession de ses deux galets. Son comportement est encore caractérisé par une suite infinie d'états qui peut être décomposée en sous-suites particulières elles-mêmes définies par le fait que les deux galets sont hors graphe, que l'un des galets est déposé et pas l'autre, que les deux galets sont déposés. On peut donc la formaliser comme suit :

$$I^{\{i,j\}} S_i^j S_{\{i,j\}} S_i^j S_{\{i,j\}} S_i^j S^{\{i,j\}} S_i^j \dots,$$

où $i, j \in \{1, 2\}$, $I^{\{i,j\}}$ est la sous-suite initiale de l'automate, S_i^j une sous-suite correspondant au fait que l'un des galets est déposé, l'autre repris, $S^{\{i,j\}}$ une sous-suite (non initiale) correspondant au fait que l'automate se déplace avec ses deux galets c'est-à-dire qu'aucun des galets n'est sur le graphe, et $S_{\{i,j\}}$ une sous-suite correspondant au fait que l'automate se déplace seul, alors que ses deux galets ont déjà été déposés.

Alors, de deux choses l'une : ou bien l'automate se trouve en situation de porter avec lui ses deux galets une infinité de fois, ou bien non.

Dans le premier cas, après une suite finie que nous dirons d'initialisation, la trajectoire de l'automate s'écrit : $S_1^2 S^{\{1,2\}} S_2^1 S^{\{1,2\}} S_1^2 S^{\{1,2\}} S_2^1 S^{\{1,2\}} S_1^2 S^{\{1,2\}} S_2^1 S^{\{1,2\}} \dots$, où S_j^i désigne une suite finie de mouvements au cours de laquelle le galet j est posé, le galet i emporté par l'automate, $S^{\{1,2\}}$ une suite de mouvements au cours de laquelle les deux galets sont emportés par l'automate (le fait que la suite en question commence par S_1^2 n'a aucune importance, il suffit éventuellement de

faire un changement de nom).

D'après les résultats dans les cas des automates à zéro ou un galet, on sait que ces différentes suites représentent des comportements bornés de l'automate, dans le temps et l'espace, et comme la suite globale devient elle-même périodique, il est clair que l'automate n'explore pas \mathcal{Z}^2 . Voir la figure 16.

Dans le second cas, à partir d'une certaine position atteinte en temps fini, l'automate ne porte jamais plus qu'un galet. Sa trajectoire est alors de la forme $S_1^2 S_{\{1,2\}} S_2^1 S_{\{1,2\}} S_1^2 S_{\{1,2\}} S_2^1 S_{\{1,2\}} S_1^2 S_{\{1,2\}} S_2^1 S_{\{1,2\}} \dots$. Alors, de deux choses l'une : ou bien la distance entre les deux galets est bornée, ou bien cette distance "croît" avec le temps (c'est-à-dire que la suite des distances des galets contient une sous-suite strictement croissante). Dans le premier cas, il est clair que l'automate laisse des points à l'extérieur de sa trajectoire, car à partir d'une certaine position, ou bien il boucle dans un carré de côté $8 \lfloor Q \rfloor$, ou bien il se déplace dans une bande infinie de cette largeur. Dans le second cas, à partir d'une certaine position, la longueur du trajet que fait l'automate ayant déposé le galet g_1 pour aller relever le galet g_2 , dans le monde vide, est telle qu'il se déplace dans ce monde comme un automate sans galet, par conséquent dans une bande de largeur bornée. Par ailleurs, lorsque l'automate porte un de ses galets, comme il doit le déposer avant d'aller à la recherche de l'autre, il ne se déplace que dans un carré borné du plan dont le côté est au plus $2.8 \lfloor Q \rfloor$. Voir la figure 17.

Notons que, dans le cas de trois galets, les distances entre les différents couples de galets ne cessent de croître avec le temps. \square

6.2 Le monde "demi-vide"

Nous pouvons encore citer le résultat suivant [Sze83a], exprimé sous la "forme Blum" :

Proposition 7 *Tout demi-plan peut être exploré par un automate à un galet, mais pas par un automate sans galet.*

Preuve L'algorithme se lit encore une fois sur un dessin de la figure 18.

L'automate, portant son galet, part d'une cellule quelconque, se dirige vers la frontière. Lorsqu'il la rencontre en la cellule f , il va déposer son galet sur la cellule nord adjacente à la cellule f . Il entreprend alors systématiquement son exploration du monde en alternant les phases suivantes. (1) Emprunter le chemin $(ws)^n$, $n \geq 1$, jusqu'à la cellule-frontière, suivre alors le chemin $(en)^n$ pour retrouver le galet. (2) Déplacer le galet d'une cellule au nord. (3) Emprunter le chemin $(se)^n$, $n \geq 1$, jusqu'à la cellule-frontière, suivre alors le chemin $(nw)^n$ pour retrouver le galet. (4) Déplacer le galet d'une cellule au nord, et reprendre à (1).

Considérons maintenant un automate sans galet. Ou bien il est un nombre fini de fois sur une cellule du bord, auquel cas il n'explore pas ce bord tout entier. Ou bien il en visite une infinité, auquel cas son comportement devient

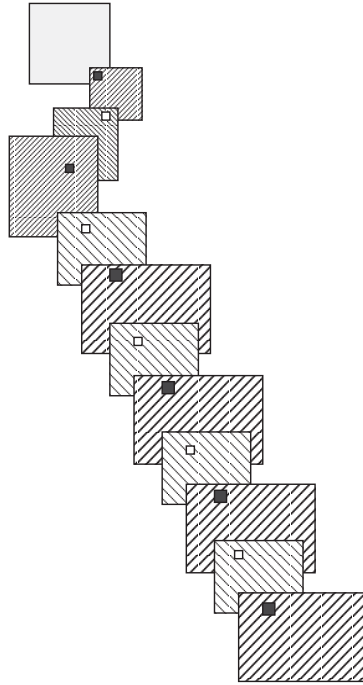


FIG. 16 - *Deux galets ne suffisent pas pour explorer le monde (1)*

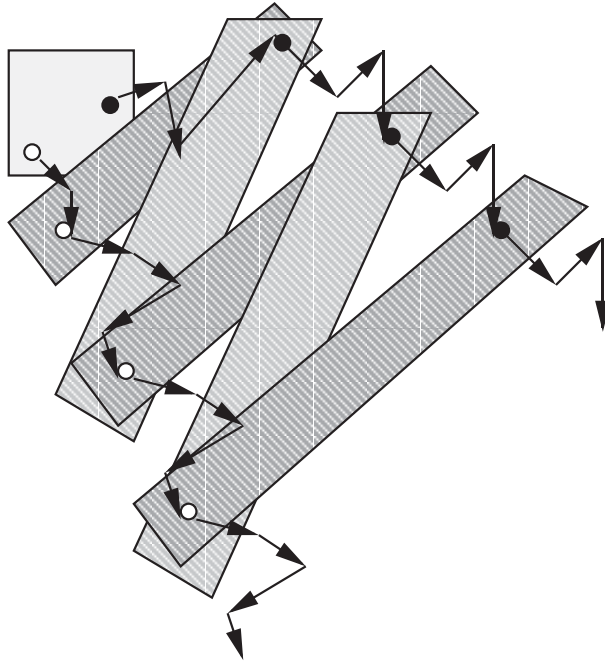


FIG. 17 - Deux galets ne suffisent pas pour explorer le monde (2)

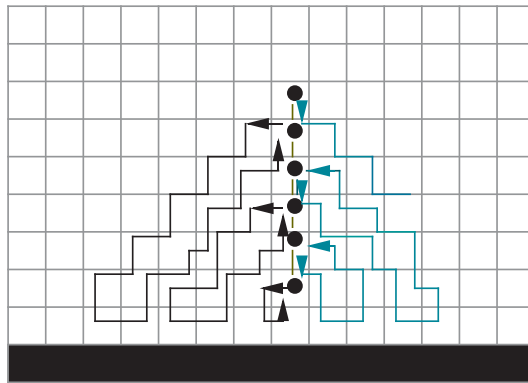


FIG. 18 - Un automate à un galet explore tout demi-plan

périodique le long du bord, dans une bande de largeur bornée. Dans les deux cas, il laisse des cellules du demi-plan à l'écart de son chemin. \square

6.3 Exploration des dédales à la Blum, finis, ou 2D-ficographe

Il est facile de voir qu'un automate à un galet peut explorer tout dédale à la Blum fini à un seul obstacle, de même que tout R -ficographe à une seule face en s'arrêtant. Il suffit de déposer le galet à la position initiale, de réaliser l'algorithme (voir proposition 1 et 2) et d'arrêter l'automate lorsqu'il retrouve son galet pour la première fois.

Qu'en est-il des ficographes finis en général?

Nous donnons d'abord le résultat de [BS77], énoncé sous la forme la plus générale qu'en donne [Hem89], qui le complète d'ailleurs par une évaluation de la complexité de l'algorithme sous-jacent.

Théorème 9 *Il existe un automate à deux galets qui explore tout 2D-ficographe normé \mathcal{L} et s'arrête après $O(n^3)$ pas, si n désigne le nombre de sommets du graphe sous-jacent à \mathcal{L} .*

L'idée de la preuve est de se ramener, par un artifice convenable, à un ficographe à une seule face ou un dédale à un seul obstacle si nous nous plaçons dans le cas de Blum. Ceci repose sur la possibilité de mettre un ordre sur les faces ou les obstacles qui permette de montrer que l'automate visite effectivement tous les points ou cellules blanches du labyrinthe. Ici, par souci de rigueur et d'homogénéité, nous choisissons de rester dans le cadre des ficographes, et nous raisonnerons donc en termes de faces et non plus d'obstacles.

1. Un ordre sur les arêtes d'un C -ficographe

Soit \mathcal{L} un C -ficographe et P un 2D-plongement normé de \mathcal{L} . Pour toute arête $e = \{h, h'\}$ de \mathcal{L} , on définit ses coordonnées comme les coordonnées du point $coo(e) = \frac{1}{2}(P(ver(h)) + P(ver(h')))$, c'est-à-dire du centre du segment de droite liant les images par P des extrémités de e .

Il en résulte un ordre entre arêtes de \mathcal{L} défini par : $e_1 \leq e_2$ si et seulement si, avec $coo(e_i) = (x_i, y_i), i \in \{1, 2\}, x_1 < x_2$ ou $x_1 = x_2$ et $y_1 \leq y_2$.

2. Une arête $e = \{h, h'\}$ appartient à une face F si $ang(h) = (h, ver(h), r(h))$ ou $ang(h')$ appartient à F .

Une arête e est dite *arête minimum d'une face F* lorsqu'elle appartient à F et satisfait $e \leq e'$ pour toute arête e' de F . Toute face a donc exactement une arête minimum propre.

Lemme 1 *Soit e l'arête minimum d'une face intérieure d'un 2D-ficographe normé \mathcal{L} . Alors c'est une arête verticale et elle appartient à deux faces distinctes de \mathcal{L} . Si elle appartient à deux faces intérieures, elle n'est l'arête minimale que de l'une d'elles.*

Preuve Supposons que e , minimum pour F , soit horizontale. Comme l'arête $e = \{h, h'\}$ est minimum (pour la face), son extrémité gauche est un sommet de degré 1. Comme la face est intérieure, $ang(h)$ et $ang(h')$ lui appartiennent. Par conséquent, la demi-droite $\{coo(e) + (x, -1/2)/x \leq 0\}$ coupe, pour la première fois, la frontière de la face au milieu d'une arête e' verticale mais telle que $e' < e$, ce qui contredit la minimalité de e . Voir la figure 19.

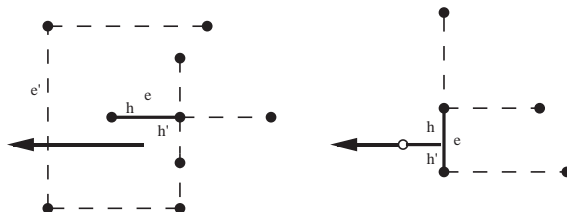


FIG. 19 - Arête minimum d'une face intérieure

Soit e , minimum pour F . Elle est donc verticale. Le point $coo(e) - (1/2, 0)$ ne peut pas être un point de la région déterminée par F , car, comme F est une face intérieure, cela impliquerait l'existence d'une arête de la face strictement inférieure à e .

De ce qui précède, il résulte que $ang(h)$ ne peut être un angle de F . S'il appartient à une face intérieure, cette face ne peut admettre e comme arête minimum, car la demi-droite $\{coo(e) + (x, 0)/x \leq -1/2\}$ doit rencontrer la frontière de cette face sur une arête strictement inférieure à e . \square

L'arête minimum d'une face intérieure est donc "l'arête verticale la plus au sud des arêtes les plus à l'ouest". Mais l'arête de la face extérieure peut évidemment être horizontale.

3. Nous sommes maintenant en mesure d'associer canoniquement à un $2D$ -ficographe normé \mathcal{L} ayant strictement plus d'une face un $2D$ -ficographe normé \mathcal{L}' n'ayant qu'une seule face.

En effet, si \mathcal{L} a plus d'une face, il a au moins une face intérieure, et il n'en n'a qu'un nombre fini m . Chacune de ces faces a une arête minimum et une seule, mais chacune d'elle appartient à deux faces (ou encore a pour ordre 2). Elles forment un ensemble $\{e_1, \dots, e_m\}$.

Si on supprime e_1 , \mathcal{L} devient un $2D$ -ficographe normé avec $m - 1$ faces intérieures. En répétant ce processus, lorsqu'on a supprimé toutes les arêtes minimum de \mathcal{L} , on obtient \mathcal{L}' , qui n'a plus qu'une face (extérieure). Il s'agit d'ailleurs d'un arbre.

La question qui se pose maintenant est de savoir si l'on peut déterminer effectivement l'arête minimale d'une face, de quel type de face, et, si oui,

à quel prix !

4. Nous allons devoir “suivre” le bord de régions déterminées par les faces du graphe, donc des courbes fermées simples sur lesquelles nous pourrions nous repérer en comptant le nombre de changements de direction, et utiliser une version discrète du théorème de Riemann-Hopf de géométrie différentielle.

Proposition 8 *Pour tout chemin rectilinéaire simplement clos w ,*

$$\bar{rin}(w) = \begin{cases} 4 & : \text{ si la région touchant } w \text{ à main gauche est bornée} \\ -4 & : \text{ si cette région est non bornée,} \end{cases}$$

où $\bar{rin}(w)$ est l'index de rotation clos de w .

Preuve La définition d'un *RPP* et celle de son index de rotation ont été données au paragraphe 2.3.4.. Dire qu'un tel chemin $w = (p_0, \dots, p_m)$ est clos signifie que $p_0 = p_m$ et son index de rotation, alors dit clos, est noté $rin(w)$ et vaut $rin(w) + rin(p_{m-1}p_m p_1)$. Enfin w est dit simplement clos lorsque deux segments distincts du chemin ne se coupent, éventuellement, qu'en une de leurs extrémités, ou plus formellement lorsque pour $0 \leq i < j < m$, $p_i p_{i+1} \cap p_j p_{j+1} \neq \emptyset$ implique soit $j = i+1$ et $p_i p_{i+1} \cap p_j p_{j+1} = \{p_j\}$, soit $j = m-1, i = 0$ et $p_i p_{i+1} \cap p_j p_{j+1} = \{p_0\} = \{p_m\}$. Il est alors clair qu'un chemin simplement clos sépare le plan en deux régions, l'une le touche à main gauche, l'autre à main droite.

Passons à la preuve proprement dite. On raisonnera modulo m sur les indices des points du chemin que l'on suppose donc simplement clos.

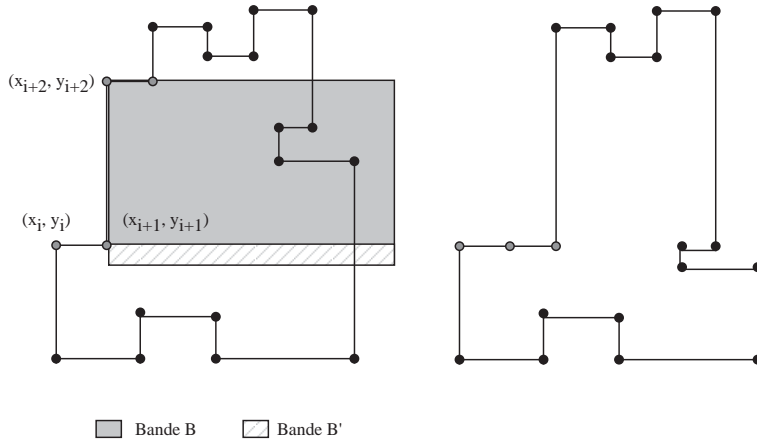
- Rappelons que $rin(p_i p_{i+1} p_{i+2})$ vaut $0, 1, -1$ ou -2 . Seuls les cas où $rin(p_i p_{i+1} p_{i+2})$ vaut 1 ou -1 sont à prendre en compte.

En effet, si $rin(p_i p_{i+1} p_{i+2}) = 0$ pour un indice i d'un point de w , l'assertion sera vraie pour le chemin simplement clos $p_0, \dots, p_{i-1}, p_{i+1}, \dots, p_m$. De plus, $rin(p_i p_{i+1} p_{i+2}) = -2$ ne se présente pas, car cela impliquerait que les segments $p_i p_{i+1}$ et $p_{i+1} p_{i+2}$ ont des points intérieurs communs.

- Les chemins $w = p_i p_{i+1} p_{i+2} p_{i+3}$ dont l'indice de rotation $rin(w)$ est nul peuvent être supprimés dans un chemin simplement clos, pour donner des chemins simplement clos ayant le même indice de rotation clos que le chemin clos originel. En effet, supposons que l'on aie un tel chemin, et que, par exemple, $rin(p_i p_{i+1} p_{i+2}) = -rin(p_{i+1} p_{i+2} p_{i+3})$, avec $p_i p_{i+1}$ ayant la direction est. Il suffit en quelque sorte de projeter $p_{i+2} p_{i+3}$ sur la droite qui supporte $p_i p_{i+1}$.

Soit $p_j = (x_j, y_j)$ un point du chemin clos originel, et ϵ un nombre réel tel que $0 < \epsilon < 1/2 \min\{|y_j - y_k| \mid 0 \leq j, k < m\}$.

Considérons les deux ensembles $B = \{(x, y) \mid x_{i+1} \leq x, y_{i+1} - \epsilon \leq y \leq y_{i+2}\}$ et $B' = \{(x, y) \mid x_{i+1} \leq x, y_{i+1} - \epsilon \leq y \leq y_{i+1}\}$. On



Chemins simplement clos, avant et après "effacement" d'un sous-chemin d'index de rotation nulle

FIG. 20 - Transformation d'un chemin clos par élimination d'un sous-chemin d'indice de rotation nul

définit une transformation τ de B sur B' par : $\tau(x, y_{i+1} - \epsilon + z) = (x, y_{i+1} - \epsilon + z \frac{\epsilon}{y_{i+2} - y_{i+1} + \epsilon})$, $x_{i+1} \leq x$ et $0 \leq z \leq y_{i+2} - y_{i+1} + \epsilon$. Notons que $\tau(p_{i+2}) = p_{i+1}$ et que $\tau(p_{i+3}) = (x_{i+3}, y_{i+1})$, ce qui réalise la projection voulue. Par ailleurs, si l'on pose :

$$q_j = \begin{cases} p_j & : \text{ si } p_j \notin B \\ \tau(p_j) & : \text{ sinon} \end{cases}$$
, on obtient un chemin simplement clos tel que si la région touchant le chemin simplement clos originel à main gauche est bornée si et seulement si celle qui touche le chemin obtenu à main gauche est bornée. Voir la figure 20.

- Il résulte du point précédent qu'il suffit de considérer les chemins simplement clos tels que $rin(p_i p_{i+1} p_{i+2})$ vaut 1 (-1) pour tout i . Dans le cas général, un chemin infini ayant cette propriété décrit une spirale : voir la figure 21.
- Mais dans le cas d'un chemin simplement clos de ce type, $p_0 p_1 \dots p_m$, on a nécessairement $m = 4$.

En effet, il est facile de vérifier le lemme suivant :

Lemme 2 Pour un chemin quelconque dans un 2D-graphe,

- $w = (p_0, p_1, \dots, p_m)$,
- $rin(w) = rin(p_0, p_1 \dots p_{k+1}) + rin(p_k, p_{k+1}, \dots, p_m)$ pour $0 \leq k < m$,
- $rin(w) = -rin(p_m, p_{m-1}, \dots, p_0)$, et

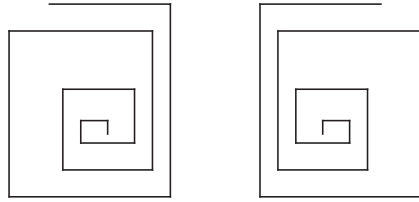


FIG. 21 - *Chemin pour lesquels les indices de rotation de trois sommets successifs sont respectivement 1 et -1*

- $rin(w) = -k \text{ modulo } 4$ si et seulement si $dir(p_{m-1}p_m) = r^k(dir(p_0p_1))$ pour $k = 0, 1, 2, 3$.

Cela entraîne que, dans notre cas particulier, m est congru à 0 modulo 4.

Considérons $d_i = |p_i - p_{i-1}|$. Si pour tout i , $d_{i+2} = d_i$, alors, le chemin étant simplement clos, $m = 4$. S'il existe un entier i tel que $d_{i+2} \neq d_i$, considérons les points $p_{i-1}, p_i, p_{i+1}, p_{i+2}$ et p_{i+3} .

Ou bien $d_{i+1} = d_{i+3}$, et alors $p_{i+3}p_{i+4}$ a des points intérieurs communs avec p_{i-1}, p_i , ou bien un seul point commun p_{i-1} , ce qui est impossible car si $i \neq 1$ on a une contradiction avec le fait que le chemin est simple, si $i = 1$, $m = 5$ et on a une contradiction avec le fait que m est congru à 0 modulo 4.

Ou bien $d_{i+1} \neq d_{i+3}$, et alors le chemin étant simple ne peut plus être qu'une spirale infinie, ce qui est encore contradictoire. Voir la figure 22. Ce qui achève la preuve de la proposition.

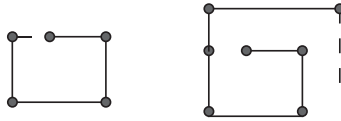


FIG. 22 - *Deux types de contradictions dans la preuve de la proposition*

□

5. Nous sommes maintenant en mesure de prouver le théorème.

Lemme 3 *Il existe un automate à un compteur qui, dans tout 2D-ficographe normé \mathcal{L} , décide en $O(n_{\mathcal{L}})$ pas si sa position de départ est sur l'arête minimum d'une face, intérieure ou extérieure, de \mathcal{L} , et qui revient à sa position initiale.*

Preuve Remarquons tout d'abord que, à toute face finie F d'un tel 2D-ficographe, on sait associer un chemin simplement clos w_F qui a le

même index de rotation que le chemin $w(F)$ qui la borde suivant le même sens de parcours. Voir la figure 23.

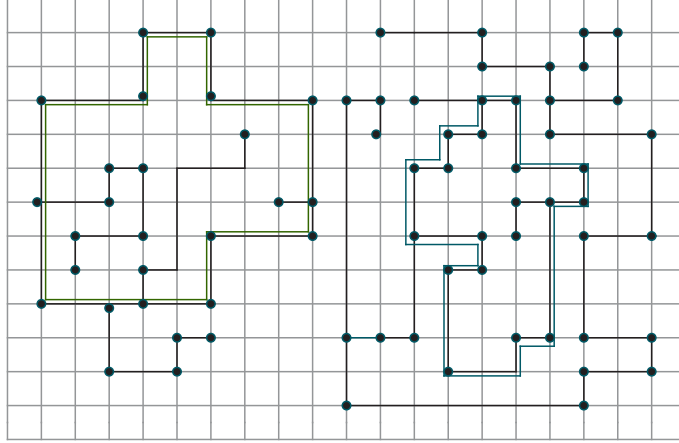


FIG. 23 - Chemins simplement clos associés à une face finie

Soit e_0 une arête du 2D-graphe normé considéré. Elle appartient à une face F . Posons $coo(e_0) = (x_0, y_0)$, et $coo(e) = (x, y)$ pour l'arête courante e , $e_0 = \{h_0, h'_0\}$, $e = \{h, h'\}$. Et supposons qu'un automate se déplace à partir de la position h'_0 . On peut aussi supposer que l'automate part vers l'est, car s'il part vers le sud ou vers l'ouest, e_0 n'est évidemment pas l'arête minimum de la face.

Tant que $x - x_0 > 0$, e n'est pas en concurrence avec e_0 . Ce n'est plus le cas lorsque $x - x_0 = 0$.

Le graphe étant normé, e ne peut être qu'une arête verticale, et quatre cas peuvent se présenter, illustrés sur la figure 24, où p_0, p_1 représentent les sommets incidents à e_0 , p_{n-1}, p_n les sommets incidents à l'arête courante e et où w désigne la partie parcourue de w_F à partir de p_2 .

On peut compléter les chemins de l'automate par les chemins indiqués en pointillé sur les figures pour obtenir des chemins simplement clos du graphe. On en déduit alors facilement que $rin(p_0 p_1 w p_{n-1} p_n)$ vaut 1 modulo 4 dans le cas (1), -1 modulo 4 dans le cas (2), 0 modulo 4 dans le cas (3) et -2 modulo 4 dans le cas 4.

Dans les deux premiers cas, e ne peut être qu'en dessous de e_0 , par conséquent e_0 ne peut pas être minimum. Dans le quatrième cas, e ne peut être qu'au dessus de e_0 , donc elle n'entre pas en compétition avec e_0 . Il ne reste que le cas (3) à examiner. Dans ce cas il se peut qu'il existe des arêtes $e'_0 = e, e'_1, \dots, e'_m$ du graphe, verticales, telles que $x'_j = x = x_0$ et qui soient situées au dessus de e_0 ou que la m -ième soit e_0 . Voir la figure 25

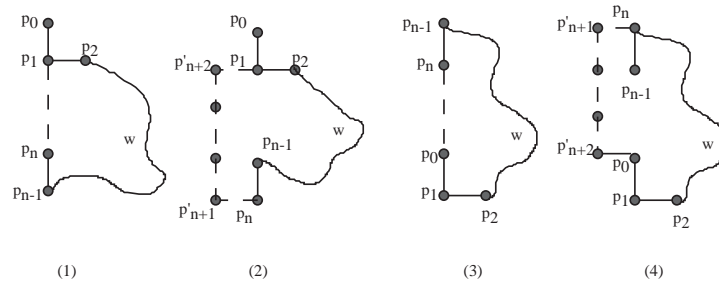


FIG. 24 - Recherche en cours de l'arête minimum

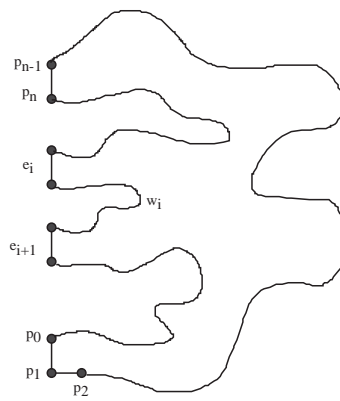


FIG. 25 - Cas (3)

Supposons qu'il existe deux arêtes e'_i, e'_{i+1} , distinctes et distinctes de e_0 dans cette suite. Alors, il existe un chemin w_i qui mène l'automate de l'une à l'autre, et il est facile de vérifier que $\text{rin}(w_i)$ vaut 2 modulo 4.

Nous sommes maintenant en mesure de conclure. Il suffit de considérer un automate à un compteur et qui, grâce à ses états compte modulo 4. En position initiale, il est sur la partie inférieure d'une arête verticale, compteur à zéro. Si son compteur s'annule, l'automate sait, par son état s'il est sur une arête possiblement minimum. Il prend alors un nombre fini de positions, et entre chacune d'elles, il évalue modulo 4 l'indice du chemin qui les sépare. Dès que cet indice vaut 0 modulo 4, c'est que sa position précédente était sa position d'origine.

Il est clair que l'on raisonne de manière analogue en partant d'une arête horizontale (qui ne peut être arête minimum que d'une face extérieure, et alors son sommet incident gauche est de degré 1).

□

6. Il ne reste plus qu'à voir qu'un automate à un compteur peut être simulé sur un tel graphe par un automate à deux galets. Ceci n'est pas difficile : l'automate, en se déplaçant de l'un à l'autre, gère ses deux galets de telle sorte que leur distance soit la valeur du compteur.
Le calcul de la complexité s'en déduit.

Le résultat le plus général est celui de Hemmerling [Hem87a], que nous énonçons ci-dessous. Il est obtenu à partir d'algorithmes d'exploration qui s'appliquent à des $2D$ -ficographes arbitraires, et utilisent un type particulier de chemins, les chemins régulièrement balancés ("regular swinging").

Théorème 10 *Il existe un automate à deux galets qui explore tout $2D$ -ficographe et s'arrête après $O(n^4)$ pas de calcul.*

Ce résultat est complété par le suivant affirmé par Hoffmann [Hof81] :

Théorème 11 *Il n'existe pas d'automate à un galet explorant tout $2D$ -labyrinthe fortement normé.*

Avant de passer au cas des graphes simples infinis dénombrables connexes ou incographes, mentionnons des résultats pour d'autres modèles de machines, sur ces mêmes ficographes, dont les preuves se déduisent aisément de celles qui viennent d'être produites.

Théorème 12 *Il existe un automate à un pointeur et un galet, et deux automates coopérants, qui explorent tout $2D$ -ficographe et s'arrêtent en respectivement $O(n^2)$ et $O(n^3)$ pas de calcul.*

7 Labyrinthes infinis, incographe

Revenons à cette idée que tout automate fini sur le plan discret dans lequel deux obstacles forment un système de coordonnées rationnelles simule toute machine de Turing. Cela conduit à se demander quelles peuvent être les performances d'un tel automate (éventuellement avec galets) dans un plan obstrué de telle sorte que le complémentaire des obstacles reste connexe? Nous avons déjà vu ce qu'il en est pour le plan non obstrué. Qu'en est-il en présence d'obstacles?

7.1 Principaux résultats

Théorème 13 *Il existe un automate à sept galets distingués qui explore tout $2D$ -incographe et tout $2D$ -ficographe, et qui s'arrête sur tout $2D$ -ficographe à la fin de son exploration.*

Ce résultat, dû à Blum et Sakoda [BS77], a été amélioré plus tard par Sze-pietowski [Sze82]. Dans la terminologie de Hemmerling, il s'énonce :

Théorème 14 *Il existe un automate à cinq galets distingués qui explore tout $2D$ -incographe normé et tout $2D$ -ficographe normé, et qui s'arrête sur tout $2D$ -ficographe à la fin de son exploration.*

7.2 Eléments de preuves

Les preuves de ces deux théorèmes s'appuient sur les faits suivants : Il existe un automate à ruban de Turing qui explore tout $2D$ -incographe, toute machine de Turing peut être simulée par un automate à deux compteurs et il est possible de simuler deux tels compteurs par un automate à galets dans un incographe convenable.

Le premier fait provient de ce qu'il suffit de mettre sur l'ensemble des directions $D = \{e, s, w, n\}$ un ordre total qui détermine un ordre total (par exemple l'ordre hiérarchique) sur l'ensemble des mots sur D , puis de faire énumérer ces mots par une machine de Turing. Si u est un tel mot, l'automate se déplacera suivant le chemin indiqué par u et reviendra à sa position de départ en suivant le renversé \bar{w} de w dans lequel si d_i est une direction interdite, \bar{d}_i aura été remplacé par un symbole neutre pour l'automate.

Construire un compteur dans un incographe revient à trouver un chemin infini, le long de certaines faces, sur lequel on pourra retrouver et manipuler tout entier naturel sans ambiguïté. Si l'incographe L considéré ne contient que des faces finies, on peut imaginer construire un tel chemin dans l'incographe L' obtenu en supprimant les arêtes minimum des faces. Mais un tel incographe peut contenir des faces infinies (contenues dans une région bornée ou non) et aucun automate à galets ne peut décider si une face est finie.

Cependant, si l'on se place dans un $2D$ -incographe normé le problème peut être surmonté grâce au résultat suivant :

Proposition 9 *Soit L un $2D$ -incographe normé, L' le $2D$ -graphe normé obtenu à partir de L en supprimant toutes les arêtes minimum des faces finies de L . Alors il n'y a pas de points isolés dans L' et toute face de L' est infinie.*

Ainsi, sous l'hypothèse que L est un $2D$ -incographe normé, à partir d'une position donnée, marquée par un galet O , l'automate peut déterminer un chemin infini de la façon suivante : ou bien le sommet sur lequel il arrive n'est situé que sur des faces finies ou bien il existe une face infinie qui le contient. Dans ce dernier cas, l'automate suivra cette face. Dans le premier cas, l'automate parcourra toutes les faces suivant un certain ordre, puis s'échappera en s'interdisant l'arête minimale de l'une canonique de ces faces, qu'il saura reconnaître grâce à deux galets supplémentaires, comme on l'a vu dans le cas fini.

Grâce à deux autres galets et à ses états internes, l'automate est en mesure de simuler deux compteurs suivant une procédure qui ne sera pas détaillée ici.

7.3 Mais ...

D'après Hoffmann [Hof85],

Théorème 15 *Il n'existe pas d'automate à quatre galets qui explore tout $2D$ -incographe fortement normé.*

8 Quelques problèmes ouverts

Ce panorama de résultats sur les automates à galets que nous avons esquissé laisse entrevoir de nombreuses perspectives de travail, comme par exemple l'étude des automates à galets sur des surfaces, la comparaison de leur puissance de reconnaissance avec celles d'outils classiques comme les automates mosaïques, leur efficacité dans d'autres domaines (automates cellulaires). Mais il reste aussi de nombreux problèmes propres, dont certains résistent depuis plusieurs années. Explicitons-en quelques uns qui, à notre connaissance, restent ouverts et suggèrent d'autres recherches.

1. Automates sans marqueur sur des motifs ou dans des labyrinthes finis
 - Un automate peut-il décider si un motif (dans le sens de Blum et Hewitt) forme une région connexe du plan?
 - Existe-t-il un C -automate fini qui explore tout $2D$ -incographe ayant seulement deux faces?

2. Automates avec marqueurs

- Peut-on améliorer les bornes inf ou sup de la complexité en espace de l'exploration de tout d -dédale fini.
- Existe-t-il un automate à un compteur qui explore tout $2D$ -ficographe?
- Existe-t-il un automate à un compteur qui explore tout R -arbre fini ou $2D$ -arbre et s'arrête toujours?
- Nous avons vu qu'il existe un automate à galets qui explore tout R -ficographe, s'arrête en $O(n^2)$ pas et utilise $O(n)$ galets non distingués. Problème : Existe-t-il un automate à galets explorant tout R -ficographe, s'arrêtant en temps linéaire et utilisant seulement $O(f)$ galets sur les R -graphes ayant f faces?
- Existe-t-il un automate à (1 pointeur, 1 galet) qui explore tout R -ficographe plan cubique?

3. Automates à marqueurs et systèmes d'automates coopérants

- Les $2D$ - et $3D$ -ficographe finis peuvent-ils être explorés par des automates de Turing en espace borné?
- Soit A un C -automate avec k galets distingués, pour la direction $D = \{e, s, w, n\}$. Il est prouvé qu'il existe un automate à k galets non marqués qui est équivalent au premier dans ce sens que, partant avec tous ses galets d'une position sur un $2D$ -graphe normé arbitraire, il simule le premier partant avec tous ses galets de la même position [Hem89].
Ce problème est ouvert pour les automates à k marqueurs sur des C -ficographe non normés.

Remarque 5 *Nous n'avons pas explicité dans ce rapport la comparaison entre automates à galets distingués et non distingués, car cette comparaison conduit nécessairement à discuter la question des simulations, que nous abordons plus précisément dans [DM97] où nous prouvons l'existence d'automates à galets intrinsèquement universels pour certains types de simulations.*

La plupart des simulations sont ad hoc et le plus généralement implicites. L'oublier peut conduire à des confusions sinon à des contradictions.

4. Automates dans les labyrinthes infinis

- Les R -incographe plans cubiques peuvent-ils être explorés par un R -automates avec un nombre fini de marqueurs (distingués ou non)?

- On a vu qu'il n'existe aucun automate à 4 galets qui explore tout $2D$ -incographe strictement normé.
Problème : Les 2-dédales co-finis peuvent-ils être explorés par trois automates coopérants?

Les quelques problèmes énumérés concernent les automates à galets dans certaines structures. Il en existe encore de nombreux pour les autres machines évoquées dans le tableau du paragraphe 5.

Références

- [Ant80] H. Antelmann. *Ober den Platzbedarf von Fallen für endliche Automaten*. PhD thesis, Humboldt-Universität, Berlin, 1980.
- [Ass77] G. Asser. Bemerkungen zum labyrinth problem. *Elektr. Inform. und Kybern. (EIK)*, 13 (3-4):203–216, 1977.
- [BH67] M. Blum and C. Hewitt. Automata on a 2-dimensional tape. In *8-th IEEE Conference on Swat.*, pages 155–160, 1967.
- [BK78] M. Blum and D. Kozen. On the power of the compass. In *19-th IEEE FOCS Conference*, pages 132–142, 1978.
- [BS77] M. Blum and W. J. Sakoda. On the capability of finite automata in 2 and 3 dimensional space. In *17-th IEEE FOCS Conference*, pages 147–161, 1977.
- [Bud75] L. Budach. On the solution of the labyrinth problem for finite automata. *Elektr. Inform. und Kybern. (EIK)*, 11:661–672, 1975.
- [Bud77] L. Budach. Environments, labyrinths and automata. In *Proceedings of Foundations of Computation Theory (FCT'77)*, 1977. LNCS 56, 54–64.
- [Bud78] L. Budach. Automata and labyrinths. *Math. Nachr.*, 86:195–282, 1978.
- [Bud81] L. Budach. Two pebbles don't suffice. In *MFCS'81*, 1981. LNCS 118, 578–589.
- [CIP86] J. H. Chang, O. H. Ibarra, and M. A. Palis. On pebble automata. *Theoretical Computer Science*, 44:111–121, 1986.
- [Con70] J. H. Conway. Mathematical games. *Scientific American*, October:120–123, 1970.
- [Con71] J. H. Conway. Mathematical games. *Scientific American*, February:112–117, 1971.
- [Cor84] R. Cori. Cartes, hypercartes et leurs groupes d'automorphismes. Séminaire Lotharingien, Université de Bordeaux 1, Avril 1984.
- [Coy78] W. Coy. On mice and maze. *Elektr. Inform. und Kybern. (EIK)*, 14:227–232, 1978.
- [CR80] Stephen A. Cook and Charles W. Rackoff. Space lower bounds for maze threadability on restricted machines. *Siam J. Comput.*, 9:637–652, 1980.

- [DM97] M. Delorme and J. Mazoyer. Deux théorèmes d'universalité pour des automates à galets. Research report, 1997.
- [Döp71] K. Döpp. Automaten in labyrinthen. *Elektr. Inform. und Kybern. (EIK)*, 7:79–94 and 167–190, 1971.
- [Fis66] Patrick C. Fischer. Turing machines with restricted memory access. *Information and Control*, 9:364–379, 1966.
- [Gar87] Max Garzon. Cyclic automata. *Theoretical Computer Science*, 53:307–317, 1987.
- [Gar93] Max Garzon. Cayley automata. *Theoretical Computer Science*, 103:83–102, 1993.
- [GH94] P. Globberman and D. Harel. Complexity results for multi-pebble automata and their logics. In *International Conference on Algorithms, Logic and Programming (ICALP94)*, 1994.
- [Hem86a] A. Hemmerling. 1-pointer automata searching finite plane graphs. *Zeitsch. f. Math., Logik und Grundlagen d. Math.*, 32:245–256, 1986.
- [Hem86b] A. Hemmerling. Remark on the power of compass. In *Proceedings of Mathematical Foundation of Computer Science (MFCS'86)*, 1986. LNCS 233, 405–413.
- [Hem87a] A. Hemmerling. Normed two-plane traps for finite systems of cooperating automata. *Elektr. Inform. und Kybern. (EIK)*, 23:453–470, 1987.
- [Hem87b] A. Hemmerling. Three dimensional traps and barrages for cooperating automata. In *Proceedings of Foundations of Computation Theory (FCT'87)*, 1987. LNCS 278.
- [Hem89] A. Hemmerling. *Labyrinth Theory*. BSB B. G. Teubner, 1989.
- [Hem91] A. Hemmerling. Pebble automata in labyrinths with rotation systems. *Zeitsch. f. Math., Logik und Grundlagen d. Math.*, pages 453–466, 1991.
- [HK84] A. Hemmerling and K. Kriegel. On searching of special classes of mazes and finite embedded graphs. In *Proceedings of Mathematical Foundation of Computer Science (MFCS'84)*, 1984. LNCS 176, 291–300.
- [Hof81] F. Hoffmann. One pebble does not suffice to search plane labyrinths. In *Proceedings of Foundations of Computation Theory (FCT'81)*, 1981. LNCS 117, 433–444.

- [Hof85] F. Hoffmann. Four pebbles don't suffice to search planar infinite labyrinths. In L. Lovasz and E. Szemerédi, editors, *Theory of algorithms*, pages 191–206. North-Holland, 1985.
- [HU79] J. E. Hopcroft and J. D. Ullmann. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.
- [Jac70] A. Jacques. Constellations et graphes topologiques. *Combinatorial Theory and its Applications*, Colloq. Math. Soc. Janos Bolyai, 1970.
- [Koz79] D. Kozen. Automata and planar graphs. In *Proceedings of Foundations of Computation Theory (FCT'79)*, pages 243–254, 1979. Akademie Verlag Berlin.
- [Kri84] K. Kriegel. *Universelle-1-Kiesel-Automaten für k-komponentige Labyrinth*. PhD thesis, Berlin, 1984.
- [Lie89] P. Lienhardt. Subdivisions de surfaces et cartes bidimensionnelles, 1989.
- [LS81] B. Leong and J. Seiferas. New real-time simulation of multihead tape units. *Journal of Association for Computing Machinery (JACM)*, 28:166–180, 1981.
- [MF71] A. R. Meyer and M. J. Fischer. Economy of description by automata, grammars and formal systems. In *21st Symp. on Switching and Automata Theory*, pages 188–191, 1971.
- [Min67] M. L. Minsky. *Finite and Infinite Machines*. Prentice Hall, 1967.
- [Mül79] H. Müller. Automata catching labyrinths with at most three components. *Elektr. Inform. und Kybern. (EIK)*, 15(1-2):3–9, 1979.
- [Myl72] J. Mylopoulos. On the recognition of topological invariants by 4-way finite automata. *Computer Graphics and Image Processing*, 1:308–316, 1972.
- [PGK91] A. Goralcikova P. Goralcik and Koubek. Alternation with a pebble. *Information Processing Letters*, 38:7–13, 1991.
- [Rem92] E. Remila. *Pavages de figures par des barres et reconnaissance de graphes sous-jacents à des réseaux d'automates*. PhD thesis, Ecole Normale Supérieure Lyon, France, 1992.
- [RFH72] P. Rosenstiehl, J. R. Fiksel, and A. Holliger. Intelligent graphs: Networks of finite automata capable of solving graph problems. In Ronald C. Read, editor, *Graph Theory and Computing*, pages 219–265, 1972.

- [Rol80] H. A. Rollik. Automaten in planaren graphen. *Acta Informatica*, 13:287–298, 1980.
- [Ros70] A. Rosenfeld. Connectivity in digital pictures. *Journal of the Association for Computing Machinery*, 17:146–160, 1970.
- [Ros79] A. Rosenfeld. *Picture Languages*. Academic Press, 1979.
- [RS72] R. Ritchie and F. Springsteel. Language recognition by marking automata. *Infor. and Control*, 20:313–330, 1972.
- [Sav70] W. J. Savitch. Relations between nondeterministic and deterministic tape complexities. *Journal of Computer and System Sciences*, 4:177–192, 1970.
- [Sav73] W. J. Savitch. Maze recognizing automata and non-deterministic tape complexity. *Journal of Computer and System Sciences*, 7:389–403, 1973.
- [Sha51] C. E. Shannon. Presentation of a maze-solving machine. In *Cybernetics; Trans. of the 8th Conf. of the Josiah Macy Jr Foand*, pages 173–180, 1951. Editor : H. V. Foerster.
- [Sha74] A. N. Shah. Pebble automata on arrays. *Computer Graphics and Image Processing*, 3:236–246, 1974.
- [Sze82] A. Szepietowski. A finite-5-pebble automaton can search every maze. *Information Processing Letter*, 15(5):199–204, 1982.
- [Sze83a] A. Szepietowski. On searching plane labyrinths. *Elektr. Inform. und Kybern. (EIK)*, 19:79–84, 1983.
- [Sze83b] A. Szepietowski. Remarks on searching labyrinths by automata. In *Proceedings of Foundations of Computation Theory (FCT'83)*. Springer Verlag, 1983.
- [Sze85] A. Szepietowski. On paterson’s problem. *Elektr. Inform. und Kybern. (EIK)*, 21 (6):313–314, 1985.
- [Tou97] L. Tougne. *Construction de familles de cercles discrets par automates cellulaires*. PhD thesis, Ecole Normale Supérieure Lyon, France, 1997.
- [VA85] G. Vijayan and Wigderson A. Rectilinear graphs and their embeddings. *SIAM Journal of Computing*, 14:355–372, 1985.