Dominos, apériodicité et quasicritaux (3/4)

Thomas Fernique

Dubna, 19-29 juillet 2012

1 Machines de Turing

1.1 Définition

Une machine de Turing est modèle abstrait du fonctionnement des appareils mécaniques ou électronique de calcul, introduit par Alan Turing en 1937 et toujours utilisé comme modèle des ordinateurs actuels. [1] Intuitivement (aujourd'hui !), une machine de Turing est un programme informatique. Formellement :

Définition 1 *Une* machine de Turing *est définie par un triplet* $(Q, \mathcal{A}, \delta)$ *où*

- Q *est un ensemble fini d'*états, *dont un* départ q_0 *et un* arrêt $\square$ *;*
- $\mathcal{A}$ *est un ensemble fini de* lettres *dont le* vide $\#$ *;*
- *la* transtion δ *est la fonction de* $(Q \backslash \square) \times \mathcal{A}$ *dans* $\mathcal{A} \times \{\leftarrow, \rightarrow\} \times Q$.

Étant donné un ruban $R \in \mathcal{A}^{\mathbb{Z}}$*, la machine en l'état* q *et en position* $i \in \mathbb{Z}$

- *lit* $s = R[i]$ *puis calcule* $\delta(q, s) = (s', \leftrightarrows, q')$ *;*
- *écrit* $R[i] := s'$ *puis passe en position* $i \pm 1$ *selon* $\leftrightarrows$ *et en l'état* q'.

La machine part en q_0 *sur un ruban presque vide et ne s'arrête qu'en* $\square$.

Exemple : incrément binaire du nombre codé sur le ruban :

	lit 0/#			lit 1		
état	écrit	déplace	va en	écrit	déplace	va en
q_0	1	$\rightarrow$	$\square$	0	$\rightarrow$	q_0

Question : que fait la machine suivante sur un ruban vide ? (écrit 1^5 en 21 pas).

1. Le premier "ordinateur" – au sens d'appareil capable de simuler une machine de Turing – fut la "machine analytique" conçue par l'anglais Charles Babbage en 1830. Il ne termina jamais sa construction et il faudra attendre le Z3 de l'Allemand Konrad Suze, en 1941.

	lit 0/#			lit 1		
état	écrit	déplace	va en	écrit	déplace	va en
q_0	1	$\rightarrow$	q_1	1	$\rightarrow$	$\square$
q_1	1	$\leftarrow$	q_1	0	$\rightarrow$	q_2
q_2	1	$\leftarrow$	q_2	1	$\leftarrow$	q_0

1.2 Problème de l'arrêt

Définition 2 *Un* problème de décision *se code par un nombre fini de lettres sur le ruban d'une machine de Turing et appelle une réponse oui ou non. Il est* décidable *s'il existe une machine de Turing qui, lancée sur ce ruban, s'arrête en temps fini après avoir écrit sur le ruban la bonne réponse.*

Les problèmes des classes P ou NP sont tous décidables : on est au delà de l'algorithmique. Pourtant, on ne peut pas décider en toute généralité si une machine de Turing M s'arrête sur un ruban w (*problème de l'arrêt*) :

Théorème 1 (Turing, 1937) *Le problème de l'arrêt est indécidable.*

Preuve.
- supposons qu'une machine M_H décide l'arrêt sur toute entrée $M; w$;
- soit D la machine qui s'arrête sur l'entrée N ssi $M_H(N; N) = \text{non}$;
- Que fait D sur l'entrée D ? Elle s'arrête ssi elle ne s'arrête pas…

$\square$

On montre par *réduction* que l'arrêt reste indécidable sur un ruban vide :

Corollaire 1 *Le problème de l'arrêt sur ruban vide est indécidable.*

Preuve.
- supposons qu'une machine M_E décide l'arrêt sur toute entrée M ;
- soit M_w la machine qui écrit w sur le ruban (vide) puis simule M sur w ;
- soit T la machine qui lit $M; w$ en entrée et écrit M_w en sortie ;
- alors la machine qui simule T pus M_E décide le problème de l'arrêt.

$\square$

Il y a un parallèle étroit avec les théorèmes d'incomplétude de Gödel (correspondance preuve/programme de Curry-Howard, 1958/69), datant des mêmes années (1931) et reposant aussi sur l'*auto-référencement*. On peut montrer que non seulement l'arrêt mais toute propriété non triviale est indécidable (théorème de Rice, 1953).[2] Bien que les machines utilisées dans les preuves soient quelque peu artificielles, les résultats sont assez révélateur de la difficulté de déterminer ce que fait un programme.

Question : que fait la machine de Turing suivante lancée sur un ruban vide ? (elle écrit 374×10^6 lettres en 119×10^{15} pas, c'est un candidat "castor affairé").

	lit 0/#			lit 1			lit 2		
état	écrit	déplace	va en	écrit	déplace	va en	écrit	déplace	va en
q_0	1	$\rightarrow$	q_1	2	$\leftarrow$	q_0	1	$\leftarrow$	q_2
q_1	0	$\leftarrow$	q_0	2	$\rightarrow$	q_1	1	$\leftarrow$	q_1
q_2	1	$\rightarrow$	$\square$	1	$\rightarrow$	q_0	1	$\rightarrow$	q_2

2 Retour aux dominos

2.1 Tuiles de Turing

On peut naturellement associer à toute machine de Turing un jeu fini de tuiles (des dominos de Wang) tel qu'à tout calcul de cette machine corresponde un pavage du plan (Fig. 1–3).

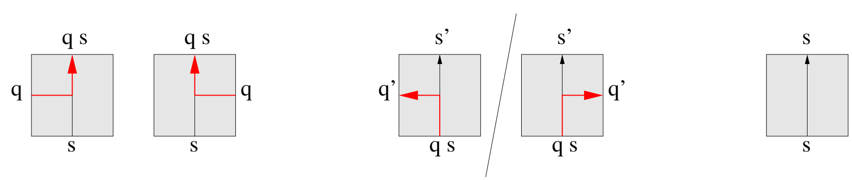

FIGURE 1 – Tuiles pour $\delta(q, s) = (s', \leftrightharpoons, q')$ et pour la propagation du ruban.

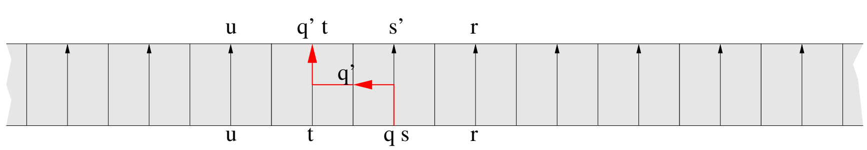

FIGURE 2 – Un pas de calcul.

2. Une propriété P est non triviale s'il existe une machine de Turing M_P qui la satisfait, et une $M_{\overline{P}}$ non (exemple : s'arrêter, calculer la fonction racine carrée...). Supposons que M décide si un programme passé en entrée vérifie une propriété P On définit alors le programme $M'(N; w)$ qui lance M sur la sous-fonction $T : N(w); M_P$. Alors M décide l'arrêt de $N(w)$.

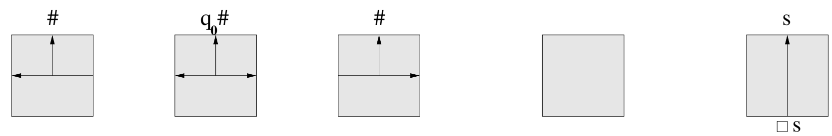

FIGURE 3 – Initialisation sur ruban vide et arrêt d'un calcul.

2.2 Indécidabilité de la complétion

Le *problème de la complétion*, également introduit par Wang, est le suivant : étant donné un jeu fini de tuiles et un motif valide fini , peut-on étendre ce motif en un pavage du plan ?

Proposition 1 (Wang, 1961) *Le problème de la complétion est indécidable.*

Preuve.
- Associons à une machine de Turing le jeu précédent ;
- retirons les tuiles d'arrêt ;
- considérons comme motif initial la seule tuile d'initialisation.

La complétion est possible ssi la machine de Turing ne s'arrête pas. □

2.3 Indécidabilité du pavage

Problème : les tuiles de Turing pavent le plan même si la machine s'arrête, car on peut toujours paver par les tuiles transportant verticalement une lettre.

Solution : "hybrider" les tuiles avec un jeu de tuiles τ dont une tuile T_0 apparaît dans tout pavage et permet d'initialiser le calcul de la machine de Turing.

Problème : T_0 doit apparaître un peu partout dans tout pavage [3] sinon, par compacité, il y aurait un pavage sans T_0. Donc τ doit être (a)périodique.

Problème : si τ ne pave que périodiquement, il a une plus grande période ; elle borne la longueur du calcul, ce qui ne permet pas toujours de détecter l'arrêt.

Solution : prendre le jeu apériodique de Robinson pour τ et l'hybrider avec les tuiles de Turing de sorte à ce que le calcul puisse être arbitrairement long. Précisément :

3. En particulier, le même calcul est donc initié un peu partout !

1. transformer les tuiles de Robinson en dominos de Wang (il y en a 56) ;
2. distinguer une parité des carrés : la même parité interdit un croisement ;
3. le côté interne sud de chaque carré pair initialise un calcul (au milieu) ;
4. une tuile du bord d'un carré pair envoie vers l'extérieur un signal ;
5. un carré pair a une lettre ou un signal sur son bord interne nord ;
6. un carré pair a un état ou un signal sur ses bords verticaux internes ;
7. une tuile avec un signal horizontal transporte verticalement une lettre ;
8. une tuile avec un signal vertical peut transporter horizontalement un état ;
9. les autres tuiles calculent, *i.e.* sont multipliées par des tuiles de Turing.

Le côté d'un carré d'ordre $2n$ comporte $C_n = 4^n + 1$ tuiles (récurrence). Soit F_n le nombre de lignes de ce carré d'ordre $2n$ qui n'intersectent aucun de ses sous-carrés. On vérifie : $F_n = F_n + 2\frac{F_n - 1}{2} = 2F_n - 1$. Comme $F_1 = 3$, on en déduit $F_n = 2^n + 1$. La zone de calcul d'un carré d'ordre $2n$ est donc de taille F_n^2. Il y a donc des zones de calcul arbitrairement grandes, d'où

Théorème 2 (Berger, 1964) *Le problème des dominos est indécidable.*

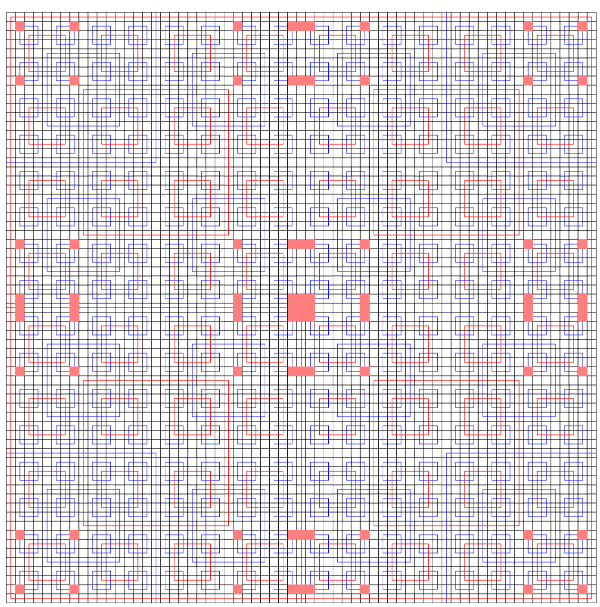

FIGURE 4 – Zone de calcul d'un carré (en rouge).

2.4 Problèmes connexes

2.4.1 Problème périodique des dominos

L'indécidabilité semblant venir de la quasipériodicité, il est naturel de se demander s'il est possible de décider si un jeu fini de dominos pave périodiquement le plan ou non – c'est le *problème périodique des dominos*, prouvé indécidable par Gurevich et Koryakov (1972). Preuve de Jeandel (2010) :

- partir d'un jeu de tuile apériodique τ ;
- croiser les dominos de τ avec des dominos d'une machine de Turing M ;
- étendre en un jeu τ' qui permet de former des grilles carrées de pas arbitraire où chaque case lance un calcul de M sur le bord sud et permet un arrêt n'importe où sur le bord nord.

Le jeu admet alors un pavage périodique ssi il existe un pas de grille égal au nombre de pas de calcul de M, *i.e.*, ssi M s'arrête.

2.4.2 Problème des k dominos

Une autre question naturelle est de se demander si le problème des dominos devient décidable si on borne le nombre de dominos considérés – c'est le *problème des k dominos*. Notamment, pour quelle valeur de k un jeu de k dominos admet-il toujours un pavage périodique (et donc le problème du pavage est décidable) ?

Le jeu apériodique de Robinson (1971) comporte 56 dominos. Kari et Čulik (1996) ont trouvé un jeu apériodique de 13 dominos (dont un conjecturé inutile). Inversement, Jeandel a montré par recherche exhaustive (2010) que tout jeu de 9 dominos admettait un pavage périodique (calcul pour 10 toujours en cours. . .).

2.4.3 Problème des k tuiles

Extension naturelle aux tuiles. Avec toujours translations seules :

- indécidable si $k \geq 11$ (Ollinger, 2008) [4] ;
- décidable si $k = 1$ car si on pave on le peut périodiquement (Wijshoff-van Leeuwen, 1984) par des pseudo-hexagones (Beauquier-Nivat, 1991).

Avec les autres isométries :

- indécidable si $k \geq 5$ (Ollinger, 2008, même preuve) ;
- ouvert pour $k = 1$ – on ne sait pas si une tuile qui pave peut toujours le faire périodiquement (Einstein problem).

Donner tuiles de nombre de Heesch 3 (définir, parallèle avec le castor affairé).

4. Idée : une unique tuile code tout un jeu de dominos, les autres "neutralisent" tous ces codage sauf un, de sorte à ce que les tuiles simulent exactement les dominos donnés