Castor affairé

Un **castor affairé** est, en théorie de la calculabilité, une machine de Turing qui atteint une « activité opérationnelle » maximale (comme le nombre de pas effectués ou le nombre de symboles écrits avant son arrêt) parmi toutes les machines de Turing d'une certaine classe. Celles-ci doivent satisfaire certaines spécifications et doivent s'arrêter après être lancées sur un ruban vierge.

Une **fonction du castor affairé** quantifie cette activité maximale pour une machine de Turing à n états ; ce type de fonction n'est pas calculable. En fait, au bout d'un certain point, une fonction du castor affairé croît plus rapidement que n'importe quelle fonction calculable. Déterminer le castor affairé pour un ensemble de machines de Turing à n états donnés est un problème insoluble algorithmiquement ; en pratique, on ne peut même pas espérer le résoudre pour un nombre n au-delà de 10.

Le concept, introduit en 1962 par le mathématicien hongrois Tibor Radó, est l'un des premiers exemples connus de fonction non calculable.

Sommaire

Nom

Le terme « castor affairé » est la traduction littérale de l'expression anglaise « *busy beaver* », désignant familièrement une personne industrieuse et travailleuse. Le terme (et le concept) est introduit en 1962 par Tibor Radó sous le nom « *busy beaver game* » (« jeu du castor affairé ») dans son article de 1962 « *On Non-Computable Functions* » (« des fonctions non calculables »)[1].

Définition

Le jeu du castor affairé à n états, introduit par Tibor Radó, utilise une classe de machines de Turing dont chaque membre répond aux spécifications suivantes :

- La machine possède n états plus un état spécial d'arrêt, où n est un nombre entier positif ; l'un des n états est défini comme l'état de départ. Ils sont typiquement nommés 1, 2, …, n, 1 étant l'état de départ, ou A, B, C, …, A étant l'état de départ.
- Elle utilise un ruban unique, s'étendant à l'infini à droite et à gauche.
- L'alphabet du ruban est {0, 1}, 0 servant de symbole vierge.
- La fonction de transition de la machine prend en compte deux entrées :

1. l'état en cours ;
2. le symbole dans la cellule du ruban en cours de lecture ;

et retourne trois sorties :

1. le symbole à écrire par dessus celui de la cellule en cours (éventuellement le même) ;
2. la direction de déplacement, à droite ou à gauche (c'est-à-dire l'action de déplacer le ruban d'une unité à gauche ou à droite de la cellule en cours) ;
3. l'état vers lequel placer la machine (éventuellement l'état d'arrêt).

La machine démarre sur une cellule quelconque d'un ruban complètement vierge (c'est-à-dire ne comportant que des 0) ; elle procède ensuite par itération de sa fonction de transition jusqu'à atteindre éventuellement l'état d'arrêt. Si, et seulement si la machine s'arrête, le nombre de 1 alors présents sur le ruban est appelé le *score* de la machine.

Le jeu du castor affairé consiste à trouver, pour un nombre n donné, la machine de Turing possédant le score maximal. Celle-ci est le *castor affairé* à n états.

Fonction du castor affairé Σ

Définition

La fonction du castor affairé $\Sigma : \mathbf{N} \to \mathbf{N}$ est définie telle que $\Sigma(n)$ est le score maximal parmi toutes les machines de Turing à 2 symboles et n états répondant aux spécifications énoncées dans le paragraphe précédent, lorsqu'elles débutent sur un ruban vierge.

Σ est une fonction bien définie : pour chaque n, il existe un nombre fini de machines de Turing à n états définies ainsi, à un isomorphisme près, et donc un nombre fini de temps d'exécution possibles.

Le score d'une machine de Turing M étant noté $\sigma(M)$, toute machine de Turing à 2 symboles et n états pour lequel $\sigma(M) = \Sigma(n)$ est appelée un castor affairé. Pour un n donné, le castor affairé n'est pas unique : il en existe au moins deux ; si M est un castor affairé, il suffit de changer le sens de déplacement du ruban lors d'une transition vers l'état d'arrêt pour obtenir un autre castor affairé.

Incalculabilité

Dans son article de 1962, Tibor Radó prouve que si $f : \mathbf{N} \to \mathbf{N}$ est une fonction calculable, alors $\Sigma(n) > f(n)$ pour tout n suffisamment grand. Σ n'est donc pas une fonction calculable[1].

Ceci implique qu'il est indécidable par un algorithme général de déterminer si une machine de Turing arbitraire est un castor affairé : un tel algorithme ne peut pas exister puisque son existence permettrait à Σ d'être calculée, ce qui est impossible.

Bien que Σ soit une fonction incalculable, il est possible de déterminer sa valeur pour des petites valeurs de n. On peut montrer sans problème que $\Sigma(0) = 0$, $\Sigma(1) = 1$ et $\Sigma(2) = 4$, et avec plus de difficulté que $\Sigma(3) = 6$ et $\Sigma(4) = 13$[2]. $\Sigma(n)$ n'est pas connue pour $n > 4$, bien que des bornes inférieures soient établies.

Nombre maximal de pas

En plus de la fonction Σ, Tibor Radó introduit la fonction du nombre maximal de pas S. Pour une machine de Turing M de l'ensemble E_n des machines de Turing à 2 symboles et n états définies plus haut, $s(M)$ est le nombre de décalages de ruban que M exécute avant de s'arrêter. $S(n)$ est alors le nombre maximal de décalages pour E_n : $S : n \mapsto \max \{ s(M) \mid M \in E_n \}$. Ces machines de Turing décalant le ruban à chaque transition ou « pas » (y compris dans une transition vers l'état d'arrêt), ce nombre maximal de décalages est également le nombre maximal de pas.

Tibor Radó a montré que S n'est pas calculable pour la même raison que Σ ne l'est pas : elle croît plus rapidement que toute fonction calculable. Il remarque que pour tout n, $S(n) \geq \Sigma(n)$, puisqu'un décalage est nécessaire pour écrire un 1 sur le ruban. S croît donc au moins aussi rapidement que Σ, laquelle croît déjà plus rapidement que n'importe quelle fonction calculable.

Les inégalités suivantes sont valides pour tout $n \geq 1$:

- $S(n) \geq \Sigma(n)$
- $S(n) \leq (2n-1).\Sigma(3n+3)$
- $S(n) < \Sigma(3n+6)$
- Il existe une constante c telle que, pour tout $n \geq 2$,

$$S(n) \leq \Sigma\left(n + \left\lfloor \frac{8n}{\log(n)} \right\rfloor + c\right)$$ (« ⌊ ⌋ » étant la partie entière).

Exemples

1 état

Si la machine ne contient qu'un seul état (**A**) , le castor affairé correspond à la table de transition suivante :

État	Symbole 0	Symbole 1
A	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **STOP**	*Non utilisé*

À partir d'un ruban vierge, cette machine lit tout d'abord le symbole **0** : elle écrit donc le symbole **1**, déplace le ruban à droite et s'arrête. On obtient donc $S(1) = 1$ et $\Sigma(1) = 1$.

Le résultat serait identique si le ruban était déplacé à gauche plutôt qu'à droite. Si la machine restait à l'état **A** après le déplacement du ruban, elle recommencerait le même processus et ne s'arrêterait jamais. Dans tous les cas, la valeur de la table de transition pour le symbole **1** n'a aucune importance, une telle machine ne pouvant jamais atterrir sur une cellule contenant ce symbole.

2 états

Pour une machine à deux états (**A** et **B**), le castor affairé correspond à la table de transition suivante[3,4,1] :

État	Symbole 0	Symbole 1
A	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **B**	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **B**
B	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **A**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **STOP**

Cette machine s'arrête au bout de 6 pas, avec 4 **1** écrits sur le ruban : $S(2) = 6$ et $\Sigma(2) = 4$.

Le tableau suivant donne le détail de ses opérations, en partant d'une bande vierge et d'un état initial **A** :

Pas	État initial	Symbole lu	Symbole écrit	Déplacement	État suivant	Ruban
1	A	0	1	Droite	B	… 0 0 0 **1** <u>0</u> 0 0 …
2	B	0	1	Gauche	A	… 0 0 0 <u>1</u> **1** 0 0 …
3	A	1	1	Gauche	B	… 0 0 <u>0</u> **1** 1 0 0 …
4	B	0	1	Gauche	A	… 0 <u>0</u> **1** 1 1 0 0 …
5	A	0	1	Droite	B	… 0 **1** <u>1</u> 1 1 0 0 …
6	B	1	1	Droite	STOP	… 0 1 **1** <u>1</u> 1 0 0 …

La colonne « Ruban » donne l'état du ruban après une opération ; le caractère qui vient d'être écrit est en gras, celui sur lequel se trouve la tête de lecture de la machine est souligné.

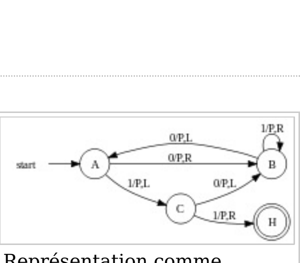
La direction du déplacement lors de la dernière opération n'a pas d'importance, la machine s'arrêtant de toute façon.

Si on inversait toutes les directions dans la table de transition (« droite » à la place de « gauche » et réciproquement), on obtiendrait également un castor affairé, la machine se comportant exactement en miroir de celle décrite ci-dessus.

Cette machine, très simple, est déjà décrite par Tibor Radó dans son article initial de 1962[1].

3 états

Pour une machine à trois états (**A**, **B** et **C**), le castor affairé produisant le plus de **1** correspond à la table de transition suivante[3,5,1] :



Représentation comme automate fini du castor affairé à trois états produisant le plus de « 1 ». Chaque cercle correspond à un état, chaque flèche à une transition. Le label de chaque flèche donne le symbole lu et le résultat ; par exemple, « 0/P,R » indique que le symbole « 0 » est lu, qu'on écrit sur le ruban (*print*, soit « P ») et qu'on se déplace à droite (*right*, « R »).

État	Symbole 0	Symbole 1
A	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **B**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **STOP**
B	• Écrire le symbole **0** • Déplacer le ruban à **droite** • Passer à l'état **C**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **B**
C	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **C**	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **A**

Cette machine s'arrête au bout de 14 pas avec 6 **1** sur le ruban.

Contrairement au cas avec 2 états, cette machine n'est pas celle qui s'arrête au bout du plus grand nombre de pas. Il en existe une autre qui est le castor affairé produisant le plus de pas, mais qui produit moins de **1**[3,6,1] :

État	Symbole 0	Symbole 1
A	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **B**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **STOP**
B	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **B**	• Écrire le symbole **0** • Déplacer le ruban à **droite** • Passer à l'état **C**
C	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **C**	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **A**

Cette machine s'arrête au bout de 21 pas avec 5 **1** sur le ruban.

On obtient donc $S(3) = 21$ et $\Sigma(3) = 6$, mais pour deux machines de Turing distinctes. Ce résultat est décrit par Tibor Radó dès 1962[1].

4 états

Pour une machine à quatre états, le castor affairé correspond à la table de transition suivante[3,7] :

État	Symbole 0	Symbole 1
A	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **B**	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **B**
B	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **A**	• Écrire le symbole **0** • Déplacer le ruban à **gauche** • Passer à l'état **C**
C	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **STOP**	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **D**
D	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **D**	• Écrire le symbole **0** • Déplacer le ruban à **droite** • Passer à l'état **A**

Cette machine s'arrête au bout de 107 pas avec 13 **1** sur le ruban. Ceux-ci ne sont pas consécutifs, l'état final du ruban étant le suivant :
… 0 0 1 0 1 1 1 1 1 1 1 1 <u>1</u> 1 1 1 0 0 …

5 états

À partir de 5 états, les castors affairés ne sont pas connus. Pour 5 états, la machine la plus active est la suivante[3,8] :

État	Symbole 0	Symbole 1
A	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **B**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **C**
B	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **C**	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **B**
C	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **D**	• Écrire le symbole **0** • Déplacer le ruban à **droite** • Passer à l'état **E**
D	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **A**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **D**
E	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **STOP**	• Écrire le symbole **0** • Déplacer le ruban à **droite** • Passer à l'état **A**

Cette machine produit 4 098 **1** en 47 176 870 pas. Les **1** ne sont pas consécutifs : 8 191 **0** sont intercalés. Découverte en 1989, on ignore s'il s'agit du castor affairé pour cette classe de machine de Turing : en 2003, il subsistait 43 machines de ce type dont la possible exécution perpétuelle n'était pas prouvée[9].

6 états

Pour 6 états, la machine la plus active est la suivante[3,10] :

État	Symbole 0	Symbole 1
A	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **B**	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **E**
B	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **C**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **F**
C	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **D**	• Écrire le symbole **0** • Déplacer le ruban à **droite** • Passer à l'état **B**
D	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **E**	• Écrire le symbole **0** • Déplacer le ruban à **gauche** • Passer à l'état **C**
E	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **A**	• Écrire le symbole **0** • Déplacer le ruban à **droite** • Passer à l'état **D**
F	• Écrire le symbole **1** • Déplacer le ruban à **gauche** • Passer à l'état **STOP**	• Écrire le symbole **1** • Déplacer le ruban à **droite** • Passer à l'état **C**

Cette machine produit environ $3{,}515 \times 10^{18\,267}$ **1** en environ $7{,}412 \times 10^{36\,534}$ pas. Elle est découverte en juin 2010.

Généralisation

Il est possible de généraliser le problème à des machines de Turing comportant n états et m symboles, conduisant aux fonctions généralisées suivantes :

- $\Sigma(n, m)$: le nombre maximal de symboles autres que 0 écrits par une machine à n états et m symboles ;
- $S(n, m)$: le nombre maximal de pas effectués par une machine à n états et m symboles.

Avec 2 états et 3 symboles, le castor affairé est la machine suivante, qui s'arrête au bout de 38 pas, son ruban comportant 9 symboles « 2 » (et 1 « 1 »)[3,11] :

Avec 3 états et 3 symboles, la machine la plus active connue s'arrête au bout de 119 112 334 170 342 540 pas, son ruban contenant 374 676 383 exemplaires du même symbole. On ignore s'il s'agit du castor affairé pour cette combinaison d'états et de symboles[3,12].

Valeurs connues

Les valeurs de $\Sigma(n)$ et $S(n)$ ne sont connues que pour $n < 5$. Pour toutes les autres ne sont connues, au mieux, que des bornes inférieures.

En 1964, Milton Green construit un ensemble de machines de Turning montrant que pour k ≥ 2 :

$$\Sigma(2k) > 3 \uparrow^{k-2} 3 > A(k-2, k-2)$$

où ↑ est la notation des flèches de Knuth et A est la fonction d'Ackermann[13].

Ainsi :

$$\Sigma(10) > 3 \uparrow\uparrow\uparrow 3 = 3 \uparrow\uparrow 3^{3^3} = 3^{3^{3^{\cdot^{\cdot^{\cdot^{3}}}}}}$$

(où le dernier terme est une tour de 3^{27} = 7 625 597 484 987 exposants), et :

$$\Sigma(12) > 3 \uparrow\uparrow\uparrow\uparrow 3 = g_1$$

où g_1 est l'énorme valeur de départ de la suite qui définit le nombre de Graham.

Les tableaux suivants recensent les valeurs exactes et les bornes inférieures de $S(n, m)$ et $\Sigma(n, m)$ pour n et $m \leq 6$[14,3]. Les entrées « ? » sont plus grandes que le maximum des entrées à leur gauche et au-dessus : elles n'ont pas été investiguées ou ont été surpassées par des machines plus petites.

S(n,m)

	2 états	3 états	4 états	5 états	6 états
2 symboles	6	21	107	≥ 47 176 870	$\geq 7{,}4 \times 10^{36\,534}$
3 symboles	38	≥ 119 112 334 170 342 540	$\geq 1{,}0 \times 10^{14\,072}$	?	?
4 symboles	≥ 3 932 964	$\geq 5{,}2 \times 10^{13\,036}$	?	?	?
5 symboles	$\geq 1{,}9 \times 10^{704}$	?	?	?	?
6 symboles	$\geq 2{,}4 \times 10^{9\,866}$	?	?	?	?

Σ(*n*,*m*)

	2 états	3 états	4 états	5 états	6 états
2 symboles	4	6	13	≥ 4 098	$\geq 3{,}5 \times 10^{18\,267}$
3 symboles	9	≥ 374 676 383	$\geq 1{,}3 \times 10^{7\,036}$	?	?
4 symboles	≥ 2 050	$\geq 3{,}7 \times 10^{6\,518}$	?	?	?
5 symboles	$\geq 1{,}7 \times 10^{352}$	?	?	?	?
6 symboles	$\geq 1{,}9 \times 10^{4\,933}$	?	?	?	?

Annexes

Articles connexes

- Machine de Turing
- Complexité de Kolmogorov
- Turmite

Sur les autres projets Wikimedia :

Castors affairés (https://commons.wikimedia.org/wiki/Category:Busy_beavers?uselang=fr), sur Wikimedia Commons

Liens externes

- (en) Busy Beaver (http://mathworld.wolfram.com/BusyBeaver.html) (MathWorld)
- (en) Busy Beaver - Currently Known Results (http://www.drb.insel.de/~heiner/BB/) (Heiner Marxen)
- (en) Historical survey of Busy Beavers (http://www.logique.jussieu.fr/~michel/ha.html) (Pascal Michel)

Références

1. (en) Tibor Radó, « On Non-Computable Functions », *Bell Systems Technology Journal*, vol. 41, nº 3, mai 1962, p. 877-884 (lire en ligne (http://www.alcatel-lucent.com/bstj/vol41-1962/articles/bstj41-3-877.pdf))
2. suite A028444 de l'OEIS
3. (en) Pascal Michel, « Historical survey of Busy Beavers » (http://www.logique.jussieu.fr/~michel/ha.html), juin 2012
4. (en) Heiner Marxen, « 2-state Busy Beaver (by T.Rado) » (http://www.drb.insel.de/~heiner/BB/simTM22_bb.html), 6 juillet 2010
5. (en) Heiner Marxen, « 3-state Busy Beaver (most ones) (by T.Rado) » (http://www.drb.insel.de/~heiner/BB/simTM32_bbO.html), 6 juillet 2010
6. (en) Heiner Marxen, « 3-state Busy Beaver (most steps) (by T.Rado) » (http://www.drb.insel.de/~heiner/BB/simTM32_bbS.html), 6 juillet 2010
7. (en) Heiner Marxen, « 4-state Busy Beaver (by A.Brady) » (http://www.drb.insel.de/~heiner/BB/simTM42_bb.html), 6 juillet 2010
8. (en) Heiner Marxen, « 5-state TM #1 from MaBu-List » (http://www.drb.insel.de/~heiner/BB/simmbL5_1.html), 6 juillet 2010
9. (en) Georgi Georgiev (Skelet), « Busy Beaver nonregular machines for class TM(5) » (http://skelet.ludost.net/bb/nreg.html), 16 mai 2003
10. (en) Heiner Marxen, « 6-state 2-symbol #b (Pavel Kropitz) » (http://www.drb.insel.de/~heiner/BB/simKro62_b.html), 6 juillet 2010
11. (en) Heiner Marxen, « 2-state 3-symbol currently best (by Brady / Michel) » (http://www.drb.insel.de/~heiner/BB/simTM23_cb.html), 6 juillet 2010
12. (en) Heiner Marxen, « 3-state 3-symbol #b (T.J. & S. Ligocki) » (http://www.drb.insel.de/~heiner/BB/simLig33_b.html), 6 juillet 2010
13. (en) Milton W. Green, « A Lower Bound on Rado's Sigma Function for Binary Turing Machines », *1964 Proceedings of the Fifth Annual Symposium on Switching Circuit Theory and Logical Design*, 1964, p. 91-94 (DOI 10.1109/SWCT.1964.3 (http://dx.doi.org/10.1109%2FSWCT.1964.3))
14. (en) Marxen Heiner, « Busy Beaver » (http://www.drb.insel.de/~heiner/BB/), 7 novembre 2011

- Lin, Shen and Radó, Tibor (1965), Computer Studies of Turing Machine Problems (http://portal.acm.org/citation.cfm?doid=321264.321270), Journal of the Association for Computing Machinery, Vol. 12, No. 2 (avril 1965), p. 196-212. Ceci inclut la thèse de Lin, qui a montré que $\Sigma(3) = 6$ en prouvant que toutes les machines à trois états et deux symboles ne s'arrêtaient jamais si elles ne s'arrêtent pas après 21 étapes.

- Brady, Allen H. (1983), The determination of the value of Rado's noncomputable function Sigma(k) for four états Turing machines, Mathematics of Computation, vol. 40, nº 162 (avril 1983), p. 647-665. Une preuve de $\Sigma(4) = 13$