Unconventional computation 1 / 2

Introduction aux et tour d'horizon des modèles non conventionnels de calcul

Jérôme Durand-Lose

LABORATOIRE D'INFORMATIQUE LIFO FONDAMENTALE D'ORLEANS

Laboratoire d'Informatique Fondamentale d'Orléans
Université d'Orléans, Orléans, FRANCE

ÉNS Cachan — 2 septembre 2014

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

L'Informatique *est* une science

Théorie, prédiction, réfutation

Mathématisation
Modèle, prédiction...

Expérimentation
Prototypage, mesure de performances...

Application
Ordinateur, internet, intelliphone...

Paradigmes ?

Modèles

- Machine de Turing, machine RAM...
- Fonctions récursives, λ-calcul...
- Modèles du parallélisme, du distribué, des communications...
- Réseaux de Pétri, Abstract state machine... (vérification...)

($\leadsto$ logique et mathématiques discrètes)

Paradigme implicite

- Action atomique ($\leadsto$ temps discret)
- Valeurs discrètes

Paradigme conventionnel

« correspond » à nos ordinateurs et y est « implantable »

- Machines de Turing (ou équivalent) :
  Calcul des données au résultat

- Espace, valeurs, états discrets
- Temps discret

- Classes de calculabilité
- Classes de complexité

Autant de façons de ne pas être conventionnel !

Sortir des bornes

Comportement et non production de *résultats*

Décider la Halte

$\rightsquigarrow$ *Hyper-computing*

Classes de complexité incomparables

ce que l'on peut encore faire en limitant une ressource

Valeurs continues

$\rightsquigarrow$ *Analog computation*

Temps continu

$\rightsquigarrow$ *Continuous computation*

Déplacement du paradigme, monde ouvert

Sans arrêt

- système d'exploitation
- serveur

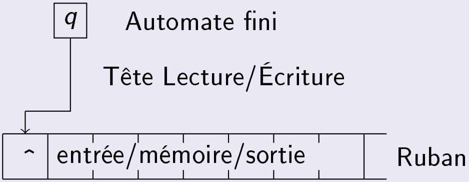
Interaction

- agents
- dialogue

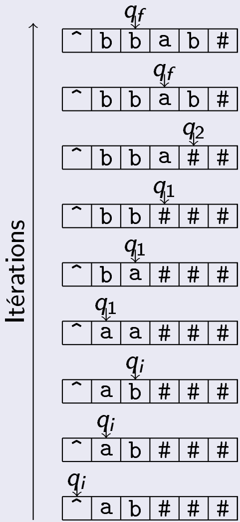
Communication

- transmission, acheminement de l'information
- π-calcul

Machines de Turing



- Entrée écrite sur le ruban
- Résultat écrit sur le ruban… quand la machine s'arrête



LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Variations sur les machines de Turing

Quasi-conventionnelles

- aléatoires
- non-déterministes
- alternantes

Sans arrêt

- *à essais et erreurs finis* (change d'avis un nombre fini de fois)
  On peut déjà décider la Halte !

À l'infini, et après

À temps infini

- définition d'une limite pour chaque case

Accélérante (vers la « réalisation »)

- chaque transition est deux fois plus rapide que la précédente
- infinité d'itérations mais durée totale finie connue

Transfinie / ordinale (Hamkins, 2007)

- on repart de la limite...
- limite pour l'état et la position de la tête
- état initial, transition suivante, transition limite
  $\rightsquigarrow$ échelle de temps ordinale
- le ruban (espace) peut aussi être ordinal !

Réalisable ?

Variations et « démonstrations » de la thèse de Church-Turing à partir de

- causalité, localité
- densité finie d'information, granularité espace-temps
- « discrétisation » à une certaine échelle
  ($\rightsquigarrow$ automates cellulaires)
- (espace euclidien)

Calcul quantique

- basé sur la superposition quantique
  (+ opérateur hermitiens + observations)
- classes de complexité différentes
- « téléportation quantique »

Relativité restreinte, générale, cosmologie. . .

À grande échelle, notre espace-temps n'est pas euclidien
Il vérifie(rait) les équations de la RG
Il existe beaucoup de solutions « intéressantes »

Accélération relative

- Deux *time-like curve*, l'une accélérée par rapport à l'autre

Premier exemple

- célérité $\nearrow$ écoulement du temps $\searrow$
- atteindre la vitesse de la lumière
- atteindre la fin des temps en un temps fini
- récupère le résultat (mais plus grand chose à faire)

Utilisation de singularités

Trou noir où on plongerait (Etesi and Németi, 2002)

- et on y serait freiné

Trou noir où on lancerait une machine

- qui y serait accélérée
- on resterait l'oreille collée à l'horizon

Structures emboîtées (Hogarth, 2004)

- monter la hiérarchie arithmétique

et pourquoi pas... (Stannett, 2013)

- revenir mais en inversant temps et espace
- modifier le passé (p.e. la valeur d'une variable)

« Vrais » réels

Implantés

- `double 3.54e41`
  nombre fini de valeurs / Approximation
- Symbolique $\sqrt{2}.\pi$
  nombre dénombrable de valeurs / Exact

General Purpose Analog Computer (Shannon, Pour-El)

- Système linéaire d'équations différentielles

Analyse récursive (Weihrauch, 2000)

- Représentation infinie, approximation convergente
  $\rho(\$q_0\$q_1\$\ldots\$q_i\$\ldots) = x$ *ssi* $q_i \in \mathbb{Q}$ et $|q_i - x| < 2^{-i}$
- Machine qui lit et écrit (sans raturer) dans ce format
- Temps infini pour une réponse complète / exacte

Modèle Blum, Shub et Smale (Blum et al., 1998)

Valeurs

- exactes
- pas de question de représentations

Manipulation

- évaluation de polynôme en les variables
- branchement en fonction du signe
- en temps constant

- « extension » des automates à compteurs
- généralisation :
  remplacer $\mathbb{R}$ par n'importe quelle structure (semi-)algébrique

Fonction $\mathbb{R}$-Récursive (Moore, 1996)

Ensemble de fonctions. . .

contenant constantes 0 et 1 (et projections)

clos par composition

clos par récursion différentielle :

$$h(\vec{x}, y) = f(\vec{x}) + \int_0^y g(\vec{x}, t, h(\vec{x}, t))dt$$

clos par recherche de zéro :

$$h(x) = \inf_{y \text{ selon } |.| \text{ puis signe}} h(\vec{x}, y) = 0$$

Physique et Biologie

Inspiration pour l'informatique (p.e. méta-heuristiques)

- recuit simulé
- algorithmes génétiques / évolutionnistes
- colonies de fourmis

L'utiliser pour calculer, p.e. optique

- communication, fibre optique, multiplexeurs optiques...
- déformation d'images, transformée de Fourrier...
  (Naughton and Woods, 2001)
- prismes, caches... (Goliaei and Jalili, 2012)

Reaction Systems (Ehrenfeucht and Rozenberg, 2010)

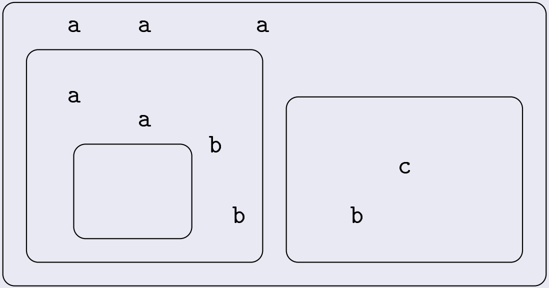
- soupe chimique
- réactions

Population protocols (Angluin et al., 2007)

- nuée d'automates simples
- rencontres au hasard
- mise à jour locale

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Membrane computing / P-Systems (Păun, 2002)



- Membranes incluses les unes dans les autres
- Objets (symboles) sont dans les espaces délimités
- Règles pour ajouter / enlever des objets / membranes

DNA computing (Păun et al., 1998)

Soupe / action / sélection

- Grand nombre de valeurs différentes engendrées
- Ne garder que celles représentant une solution

Modifications

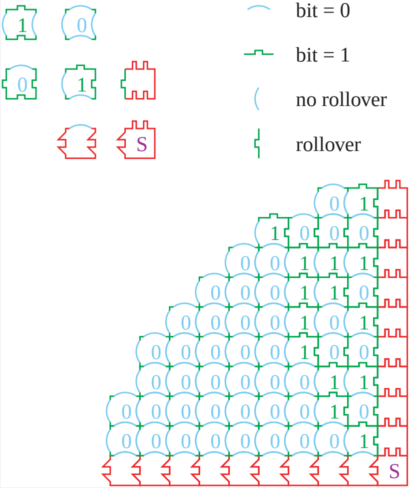
- Grosse molécule codant les données
- Modifications chimiques faisant un calcul

Construction de formes

- Repliement des protéines
- Interaction / assemblage
- (auto-assemblage des tuiles)

Algorithmic Self-Assembly of DNA (Winfree, 2000)

Fig .3



- Tuiles : grosses molécules
- S'attachent grâce à des brins d'ADN
- On part d'une *graine*
- Formes apparaissent par agrégation

Motivations théoriques

Expériences d'assemblages (tapis Serpinski)

Nanotechnologies

Calculer

- À partir de la graine,
  mise en place de l'entrée
- Ligne du dessus commence avec
  la représentation de la transition

Mise à jour *distribuée*
ni séquentielle ni parallèle synchrone

ˆ	b	b q_f	a	b	#
ˆ	b	b	a q_f	b	#
ˆ	b	b	a	# q_2	#
ˆ	b	b	# q_1	#	#
ˆ	b	a q_1	#	#	#
ˆ	a q_1	a	#	#	#
ˆ	a	b q_i	#	#	#
ˆ	a q_i	b	#	#	#
ˆ q_i	a	b	#	#	#

Construction de formes

- un carré ?
- tous les carrés ?
- en temps optimal ? (Becker et al., 2008)

Questions (Becker, Pattitz, Woods)

- formes/figures atteignables ?
- séparations des températures ?
- universalité intrinsèque : un jeu de tuiles *simulant* tous les autres

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

(von Neumann et Ulam 1952)

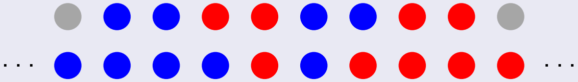
- *agencement* bi-infini de *cellules*
- à chaque itération, toutes les cellules changent d'état en fonction des voisines
- mode de fonctionnement des cellules identique (règle de transition unique)

Diagramme espace-temps

(von Neumann et Ulam 1952)

- *agencement* bi-infini de *cellules*
- à chaque itération, toutes les cellules changent d'état en fonction des voisines
- mode de fonctionnement des cellules identique (règle de transition unique)

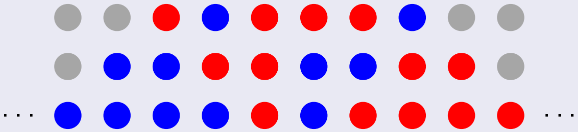
Diagramme espace-temps



(von Neumann et Ulam 1952)

- *agencement* bi-infini de *cellules*
- à chaque itération, toutes les cellules changent d'état en fonction des voisines
- mode de fonctionnement des cellules identique (règle de transition unique)

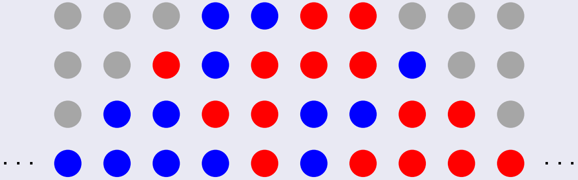
Diagramme espace-temps



(von Neumann et Ulam 1952)

- *agencement* bi-infini de *cellules*
- à chaque itération, toutes les cellules changent d'état en fonction des voisines
- mode de fonctionnement des cellules identique (règle de transition unique)

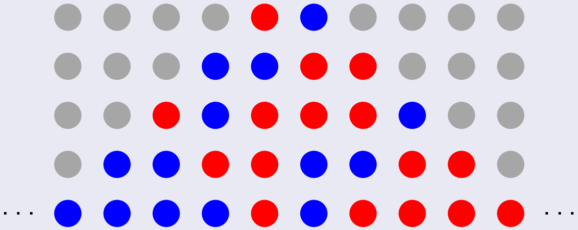
Diagramme espace-temps



(von Neumann et Ulam 1952)

- *agencement* bi-infini de *cellules*
- à chaque itération, toutes les cellules changent d'état en fonction des voisines
- mode de fonctionnement des cellules identique (règle de transition unique)

Diagramme espace-temps



Propriétés

- dynamique uniforme dans le temps et l'espace
- massivement parallèle
- synchronisation forte

Modélisation

- parallélisme à grain fin
- tout phénomène physique uniforme dans l'espace

Temps fini

- localement simulable en temps polynomial
- espace hyperbolique : résout SAT en temps polynomial (Margenstern and Morita, 2001)

Problématique propre

Synchronisation d'une ligne de fusiliers

- un seul général
- pas de communication globale
- interdiction de tirer avant

Approche récursive (Goto, 1966, Fig. 3+6)

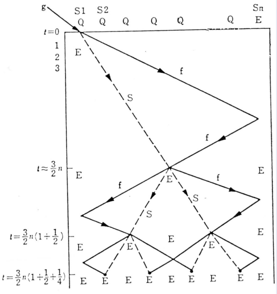


図 3·5 一斉射撃の問題（連続近似）

G	s_1	s_2	s_3	s_4	s_5	s_6
g	Q	Q	Q	Q	Q	E
$t=0$	f's'Efs	Q	Q	Q	Q	E
1	E	Q2f	Q	Q	Q	E
2	E	Q1	Qf	Q	Q	E
3	E	Q&	Q	Qf	Q	E
4	E	Q	Q2	Q	Qf	E
5	E	Q	Q1	Q	Q	f'Ef
6	E	Q	QS	Q	f'Q	E
7	E	Q	Q	a'Q*	Q	E
8	E	Q	f'S'ESf	f's'Esf	Q	E
9	E	f'2Q	E	E	Q2f	E
10	f'Ef	1Q	E	E	Q1	f'Ef
11	E	f'S'ESf	E	E	f's'Esf	E
12	a'Ea	E	a'Ea	a'Ea	E	a'Ea
13	F	F	F	F	F	F

図 3·6 一斉射撃解 ($n=6$)

Unconventional computation 2 / 2

géométrie euclidienne et machines à signaux

Jérôme Durand-Lose

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Laboratoire d'Informatique Fondamentale d'Orléans
Université d'Orléans, Orléans, FRANCE

L'UNIVERSITÉ N'EST PAS UNE ENTREPRISE
LE SAVOIR N'EST PAS UNE MARCHANDISE

ÉNS Cachan — 2 septembre 2014

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Règle et compas (Huckenbeck, 1989)

Objets

- points, droites, cercles

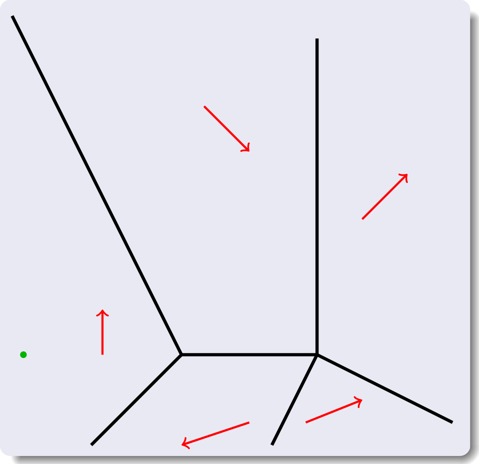
Primitives

- nouveau point (intersection cercles, droites)
- nouvelle droite
- nouveau cercle
- avoir une intersection ?
- appartenir à ?

Automates

- basé sur ces primitives

À dérivée constante par morceau (Asarin and Maler, 1995; Bournez, 1999)



- Régions polygonales
- Vitesse constante par région

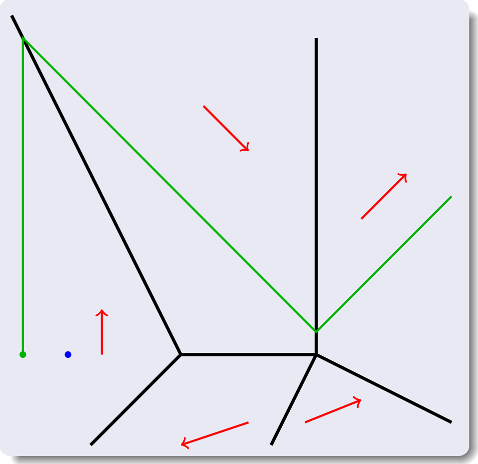
Calculer

- zone départ
- zone d'arrêt

Coefficients entiers

- évolution indécidable
- degré d'indicidabilité dépendant dimension

À dérivée constante par morceau
(Asarin and Maler, 1995; Bournez, 1999)



- Régions polygonales
- Vitesse constante par région

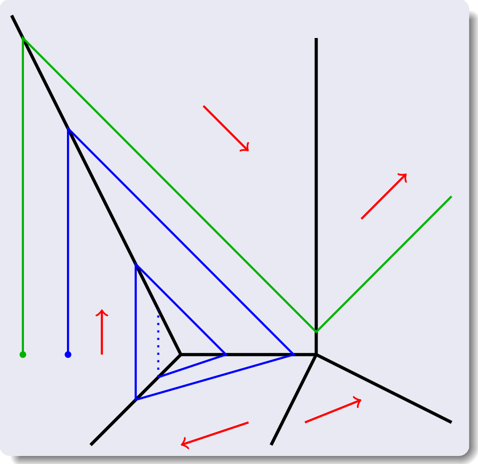
Calculer

- zone départ
- zone d'arrêt

Coefficients entiers

- évolution indécidable
- degré d'indicidabilité dépendant dimension

À dérivée constante par morceau (Asarin and Maler, 1995; Bournez, 1999)



- Régions polygonales
- Vitesse constante par région

Calculer

- zone départ
- zone d'arrêt

Coefficients entiers

- évolution indécidable
- degré d'indicidabilité dépendant dimension

Automate cellulaire et assemblage de tuiles

D. E.-T. définis par des contraintes

- locales
- discrètes
- (liens pavages)

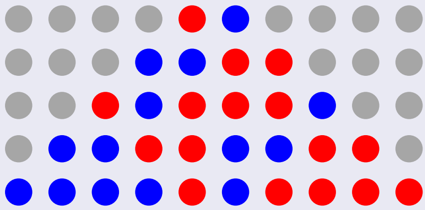
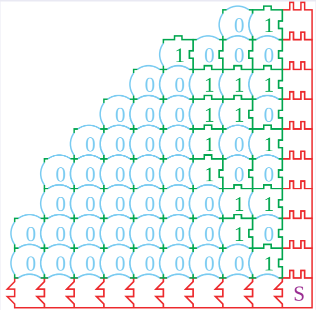
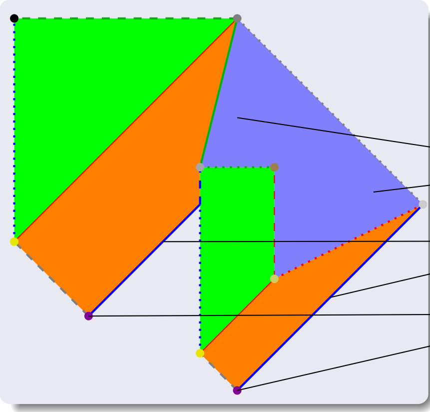


Fig .3



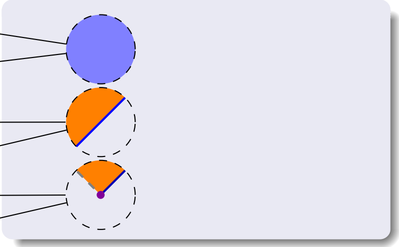
LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Automates de Mondrian (Jacopini and Sontacchi, 1990)



Contraintes

- locales
- continues



Une dimension pour le temps

Automate cellulaire : utilisation de signaux

Synchronisation d'une ligne de fusiliers (Goto, 1966)

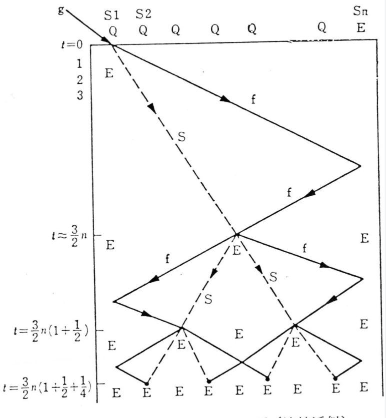


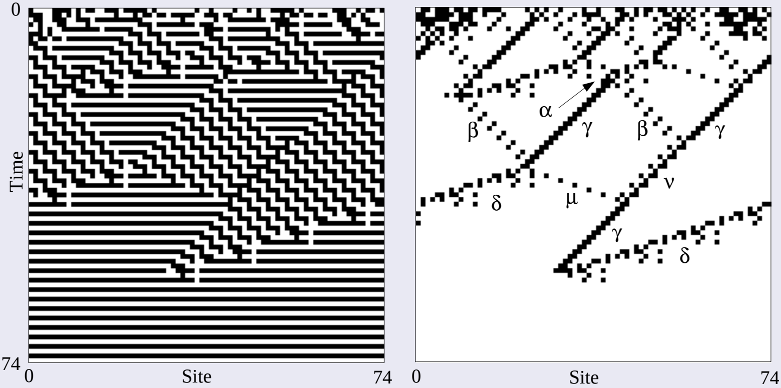
図 3·5 一斉射撃の問題（連続近似）

G	s_1	s_2	s_3	s_4	s_5	s_6
g	Q	Q	Q	Q	Q	E
$t=0$	f's'Efs	Q	Q	Q	Q	E
1	E	Q2f	Q	Q	Q	E
2	E	Q1	Qf	Q	Q	E
3	E	Q&	Q	Qf	Q	E
4	E	Q	Q2	Q	Qf	E
5	E	Q	Q1	Q	Q	f'Ef
6	E	Q	QS	Q	f'Q	E
7	E	Q	Q	a'Q'	Q	E
8	E	Q	f'S'ESf	f's'Esf	Q	E
9	E	f'2Q	E	E	Q2f	E
10	f'Ef	1Q	E	E	Q1	f'Ef
11	E	f'S'ESf	E	E	f's'Esf	E
12	a'Ea	E	a'Ea	a'Ea	E	a'Ea
13	F	F	F	F	F	F

図 3·6 一斉射撃解 ($n=6$)

Analyse en terme de signaux

Das et al. (1995)



(a) Space-time diagram. (b) Filtered space-time diagram.

Conception avec des signaux

Fischer (1965)

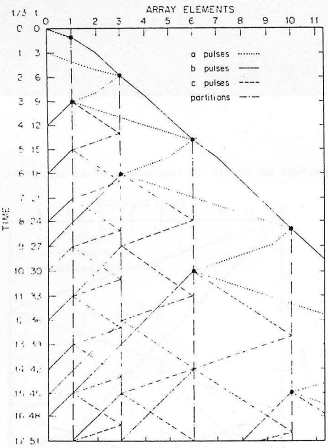
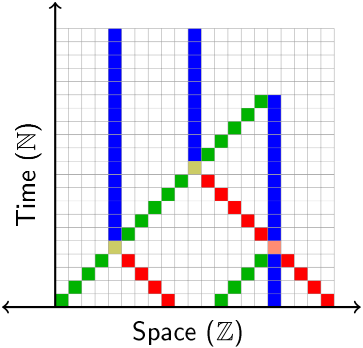
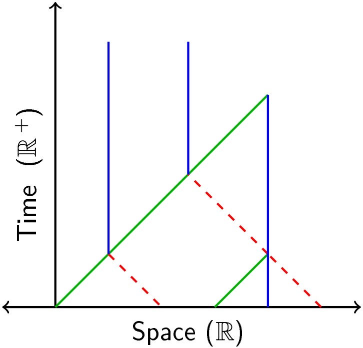


FIG. 2. Solution to the prime problem

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Signaux





- Signal (meta-signal)
- Collision (règle)

Vocabulaire et exemple : trouver le milieu

Meta-signaux (vitesse)

M (0)

Règles de collision

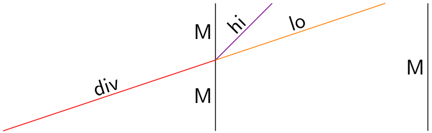
Vocabulaire et exemple : trouver le milieu

Meta-signaux (vitesse)

M	(0)
div	(3)

Règles de collision

Vocabulaire et exemple : trouver le milieu



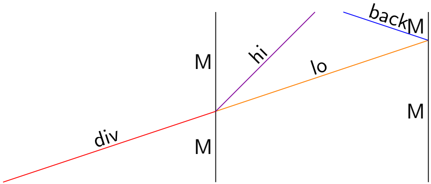
Meta-signaux (vitesse)

M	(0)
div	(3)
hi	(1)
lo	(3)

Règles de collision

$$\{ \text{div}, \text{M} \} \rightarrow \{ \text{M}, \text{hi}, \text{lo} \}$$

Vocabulaire et exemple : trouver le milieu



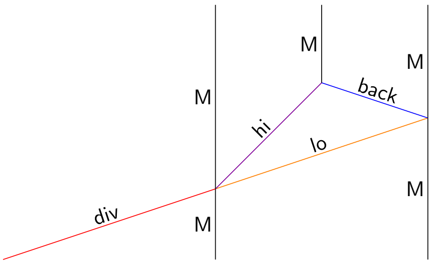
Meta-signaux (vitesse)

M	(0)
div	(3)
hi	(1)
lo	(3)
back	(-3)

Règles de collision

$$\{\ \text{div},\ \text{M}\ \} \rightarrow \{\ \text{M},\ \text{hi},\ \text{lo}\ \}$$
$$\{\ \text{lo},\ \text{M}\ \} \rightarrow \{\ \text{back},\ \text{M}\ \}$$

Vocabulaire et exemple : trouver le milieu



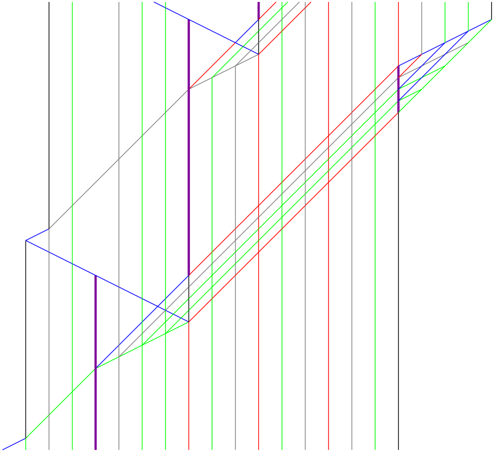
Meta-signaux (vitesse)

M	(0)
div	(3)
hi	(1)
lo	(3)
back	(-3)

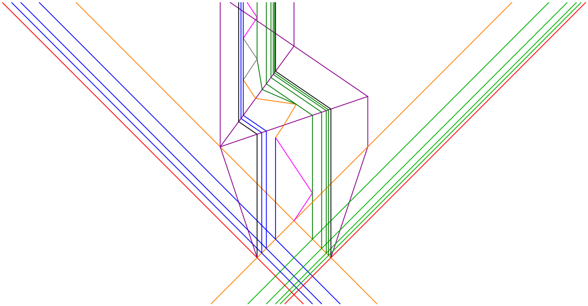
Règles de collision

$$\{\ \text{div},\ \text{M}\ \} \rightarrow \{\ \text{M},\ \text{hi},\ \text{lo}\ \}$$
$$\{\ \text{lo},\ \text{M}\ \} \rightarrow \{\ \text{back},\ \text{M}\ \}$$
$$\{\ \text{hi},\ \text{back}\ \} \rightarrow \{\ \text{M}\ \}$$

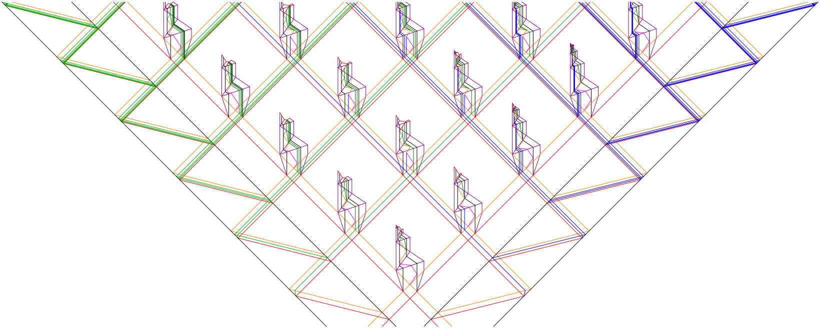
Dynamique complexe



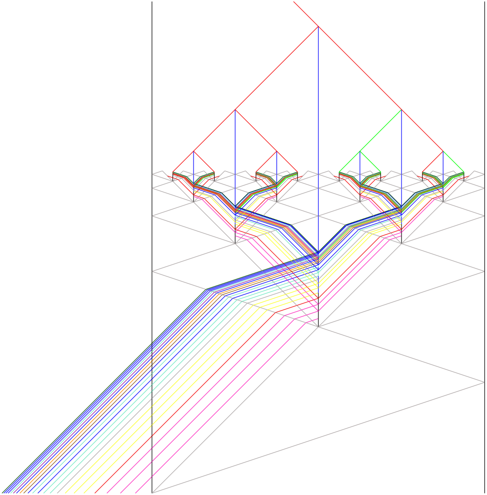
Dynamique complexe



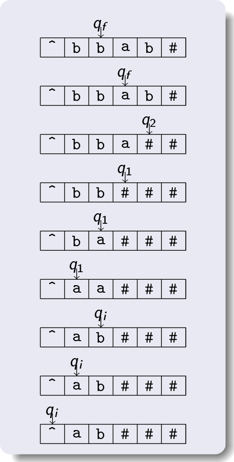
Dynamique complexe

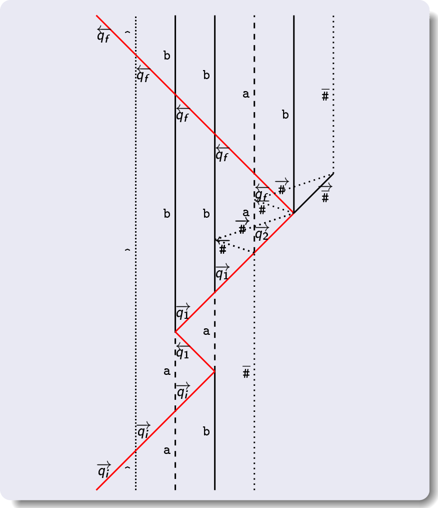


Dynamique complexe



Calcul par simulation d'une machine de Turing





Dynamique à trois vitesses sur $\mathbb{Q}$ (Becker et al., 2013)

- Vitesses $\in \mathbb{Q}$
- Positions initiales $\in \mathbb{Q}$

$\rightsquigarrow$ Collisions à coordonnées rationnelles
(solution d'un système linéaire à deux équations sur $\mathbb{Q}$)

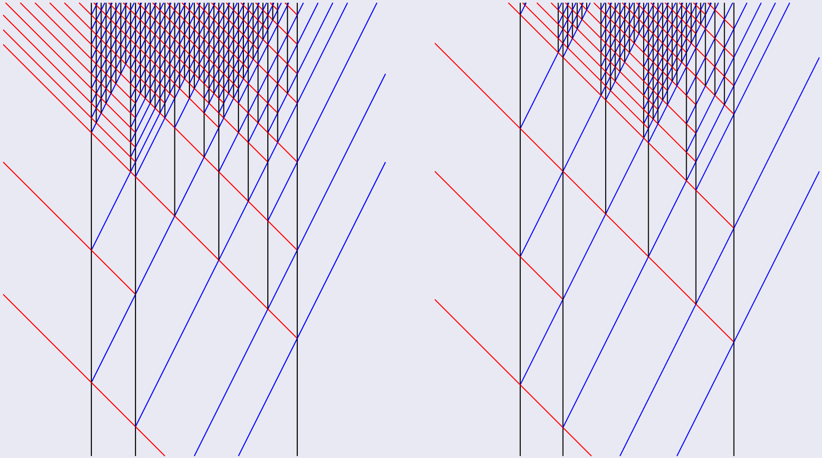
Implanté en Java

- précision exacte (sur $\mathbb{Q}$)
- (tonnes de diagrammes espace-temps)

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

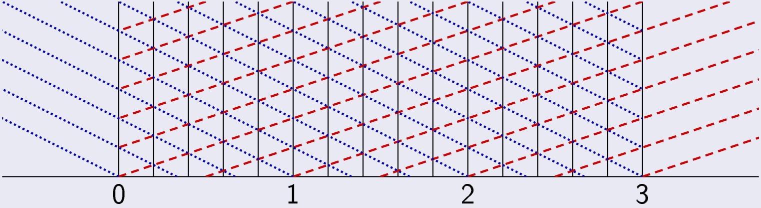
Dynamique à trois vitesses sur $\mathbb{Q}$ (Becker et al., 2013)

Engendre tout à chaque fois



Dynamique à trois vitesses sur $\mathbb{Q}$ (Becker et al., 2013)

Diagramme inclus dans une grille



- Pas d'accumulation
- Pas de calcul

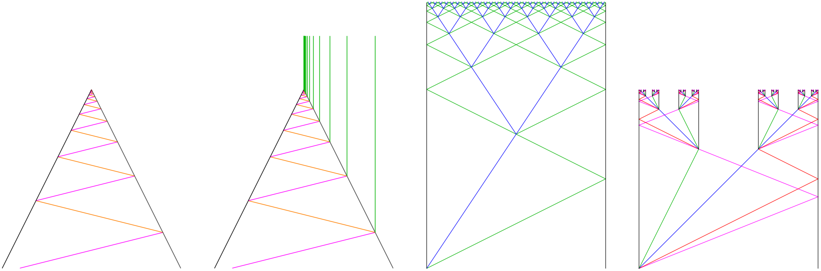
Algorithme d'Euclide

- calcul du modulo

- ré-itère en changeant les rôles
- pgcd

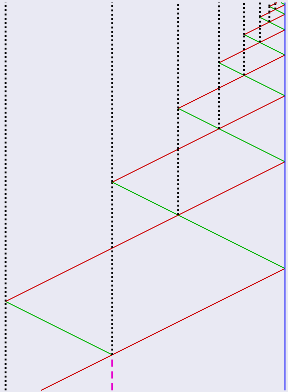
- pgcd converge (sur les entiers)

Génération de fractales

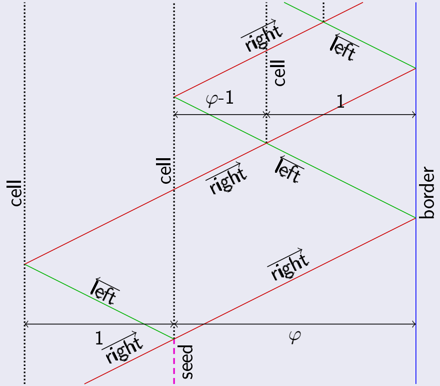


Génération de fractales

Fractale



Construction



φ est le *nombre d'or*

Calculer avec 3 vitesses ? (Durand-Lose, 2013)

- utiliser la fractale… mais sans l'engendrer

q_1
c c b c a # …

q_1
c c b c # …

q_1
c c b b …

q_1
c c c b …

q_0
c c c # …

q_0
c c b …

q_0
c b b …

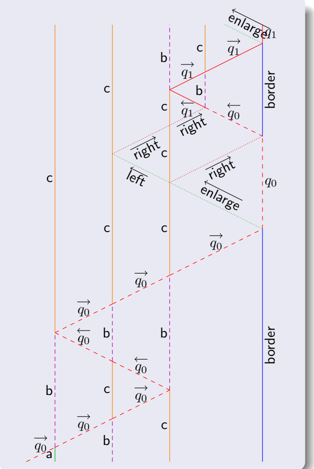
q_0
b b b …

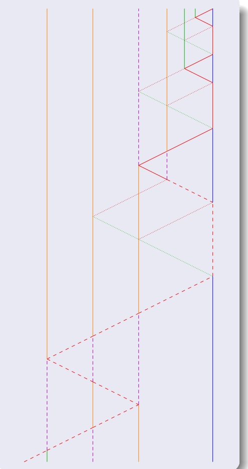
q_0
b c b …

q_0
b c c …

q_0
b b c …

q_0
a b c …





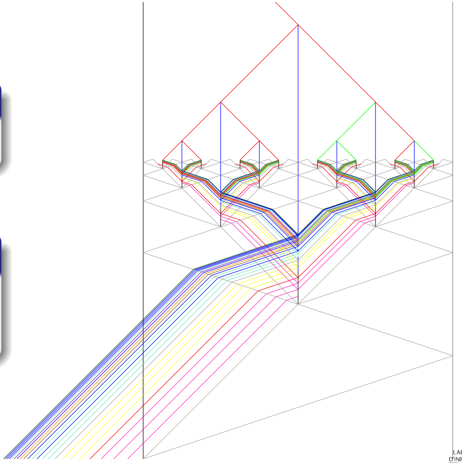
Satisfaction de formules booléennes quantifiées

QSAT

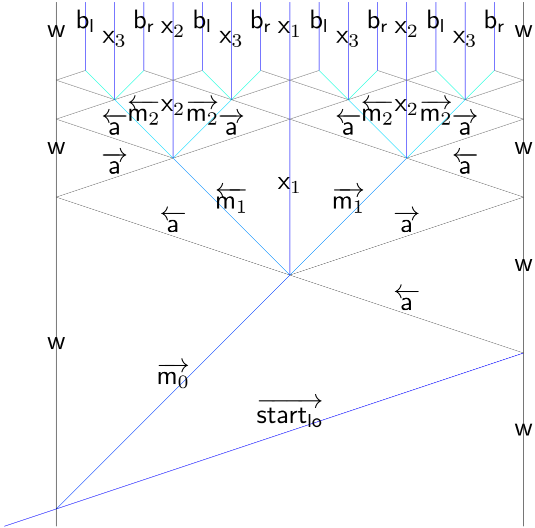
- $\exists x_1 \forall x_2 \forall x_3\, x_1 \wedge (\neg x_2 \vee x_3)$

Duchier et al. (2011)

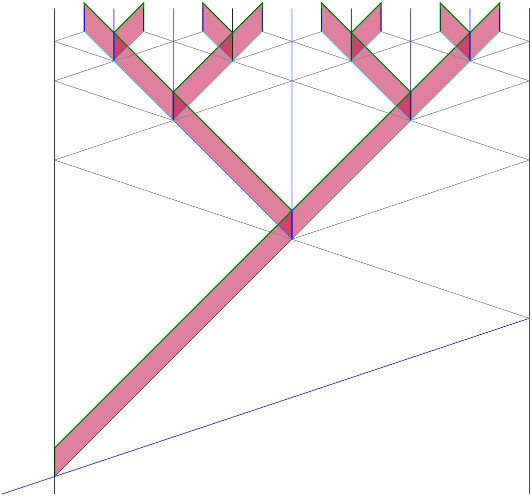
- une formule QSAT
  $\rightsquigarrow$ une machine à signaux



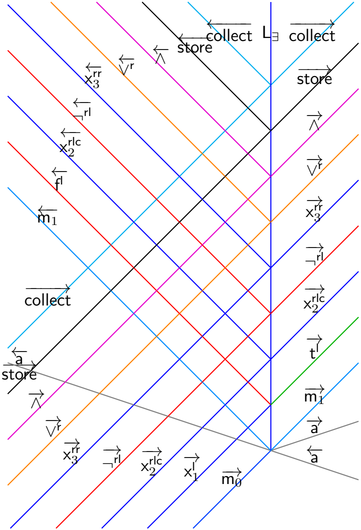
Création de l'arbre de tous les cas



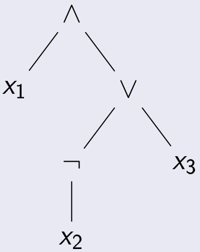
Propagation dans l'arbre



Duplication du faisceau

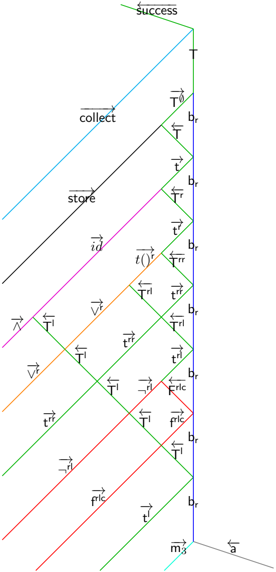


Évaluation de la formule

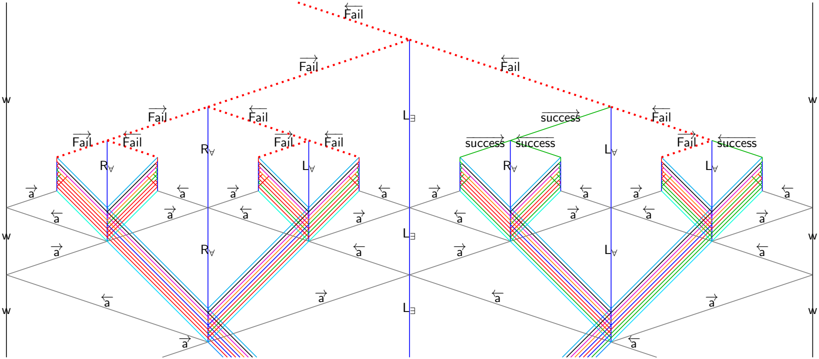


cas :

- x_1 vrai
- x_2 faux
- x_3 vrai



Agrégation du résultat



Complexité

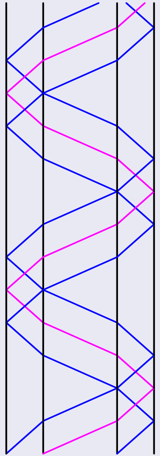
- durée constante
- profondeur de collisions quadratique
- largeur de collisions exponentielle

Machine générique pour QSAT (Duchier et al., 2012)

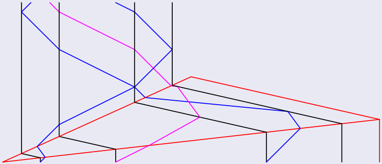
- formule codée uniquement dans la configuration initiale
- durée constante
- profondeur de collisions cubique
- largeur de collisions exponentielle

Primitives géométriques : accélération et repliement

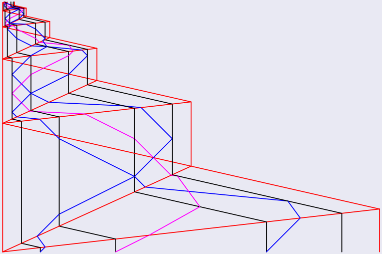
Normal



Réduit



Itéré ⇝ contracté



Hypercomputation

Contraction de l'espace et du temps

- automatique
- affine par morceau

Deux échelles de calcul différentes

- permet d'observer un calcul infini depuis l'extérieur
- on peut décider la Halte !

- Contractions emboîtées (Durand-Lose, 2009)

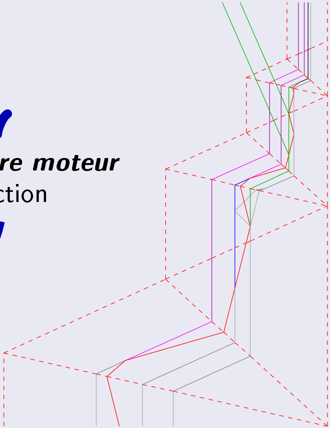
Dynamiques à deux échelles (Durand-Lose, 2012)

Ordre

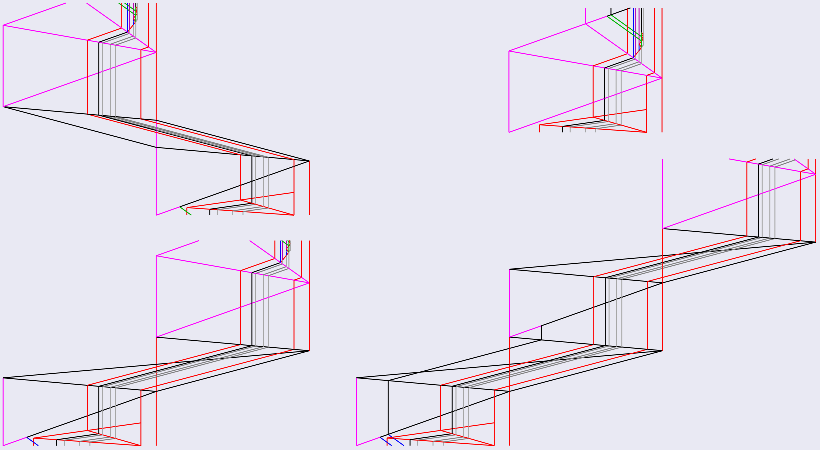
Structure de contrôle
calcul

Structure moteur
action

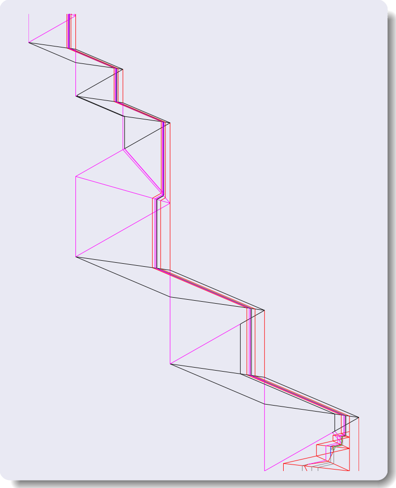
Redémarre

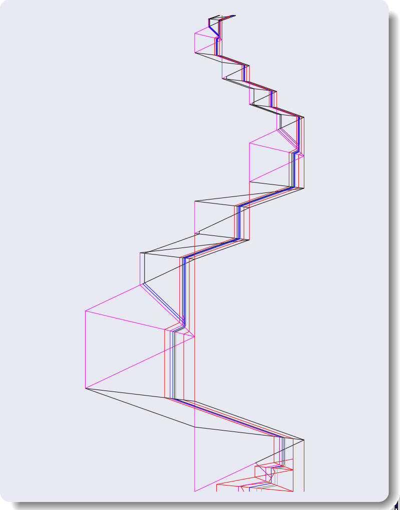


Déplacements spatiaux

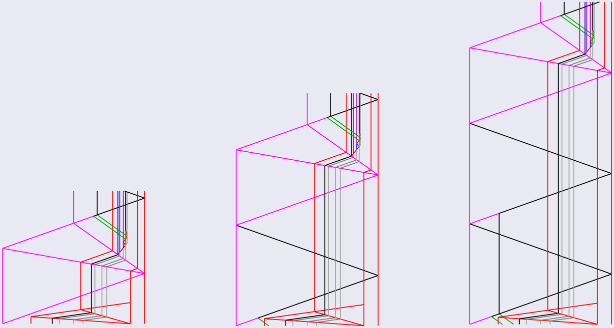


Choisir le lieu de l'accumulation





Plus ou moins de retard



- choisir la date de l'accumulation

Les accumulations sont exactement les points à coordonnées
- *c.e.* dans le temps
- *d-c.e.* dans l'espace

Références

Communautés

- Revues *Unconventional computation* et *Natural computation*
- Conférences *Unconventional computation and Natural computation* mais aussi DNA, Membrane Computing...
- *Physics and computation* workshop
- ...

Lecture

- Syropoulos (2010) livre où une grande partie des sujets présentés sont abordés
- Rozenberg et al. (2012) Handbook of Natural Computing

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Angluin, D., Aspnes, J., Eisenstat, D., and Ruppert, E. (2007). The computational power of population protocols. *Distributed Computing*, 20(4) :279–304.

Asarin, E. and Maler, O. (1995). Achilles and the Tortoise climbing up the arithmetical hierarchy. In *FSTTCS '95*, number 1026 in LNCS, pages 471–483.

Becker, F., Chapelle, M., Durand-Lose, J., Levorato, V., and Senot, M. (2013). Abstract geometrical computation 8: Small machines, accumulations & rationality. Submitted.

Becker, F., Rémila, E., and Schabanel, N. (2008). Time optimal self-assembly for 2d and 3d shapes : The case of squares and cubes. In Goel, A., Simmel, F. C., and Sosík, P., editors, *DNA Computing, 14th International Meeting on DNA Computing, DNA 14, Prague, Czech Republic, June 2-9, 2008. Revised Selected Papers*, volume 5347 of *LNCS*, pages 144–155. Springer.

Blum, L., Cucker, F., Shub, M., and Smale, S. (1998). *Complexity and real computation*. Springer, New York.

Bournez, O. (1999). Some bounds on the computational power of piecewise constant derivative systems. *Theory Comput. Syst.*, 32(1) :35–67.

Das, R., Crutchfield, J. P., Mitchell, M., and Hanson, J. E. (1995). Evolving globally synchronized cellular automata. In Eshelman, L. J., editor, *International Conference on Genetic Algorithms '95*, pages 336–343. Morgan Kaufmann.

Duchier, D., Durand-Lose, J., and Senot, M. (2011). Solving Q-SAT in bounded space and time by geometrical computation. In Ganchev, H., Löwe, B., Normann, D., Soskov, I., and Soskova, M., editors, *Models of computability in contecxt, 7th Int. Conf. Computability in Europe (CiE '11) (abstracts and handout booklet)*, pages 76–86. St. Kliment Ohridski University Press, Sofia University.

Duchier, D., Durand-Lose, J., and Senot, M. (2012). Computing in the fractal cloud: modular generic solvers for SAT and Q-SAT variants. In Agrawal, M., Cooper, B. S., and Li, A., editors,

LABORATOIRE D'INFORMATIQUE FONDAMENTALE D'ORLEANS

Theory and Applications of Models of Computations (TAMC '12), number 7287 in LNCS, pages 435–447. Springer.

Durand-Lose, J. (2009). Abstract geometrical computation 3: Black holes for classical and analog computing. *Nat. Comput.*, 8(3) :455–472.

Durand-Lose, J. (2012). Abstract geometrical computation 7: Geometrical accumulations and computably enumerable real numbers. *Nat. Comput.*, 11(4) :609–622. Special issue on Unconv. Comp. '11.

Durand-Lose, J. (2013). Irrationality is needed to compute with signal machines with only three speeds. In Bonizzoni, P., Brattka, V., and Löwe, B., editors, *CiE '13, The Nature of Computation*, number 7921 in LNCS, pages 108–119. Springer. Invited talk for special session *Computation in nature*.

Ehrenfeucht, A. and Rozenberg, G. (2010). Reaction systems : a formal framework for processes based on biochemical interactions. *ECEASST*, 26.

Etesi, G. and Németi, I. (2002). Non-Turing computations via Malament-Hogarth space-times. *Int. J. Theor. Phys.*, 41(2) :341–370.

Fischer, P. C. (1965). Generation of primes by a one-dimensional real-time iterative array. *J. ACM*, 12(3) :388–394.

Goliaei, S. and Jalili, S. (2012). An optical solution to the 3-sat problem using wavelength based selectors. *The Journal of Supercomputing*, 62(2) :663–672.

Goto, E. (1966). Ōtomaton ni kansuru pazuru [Puzzles on automata]. In Kitagawa, T., editor, *Jōhōkagaku eno michi [The Road to information science]*, pages 67–92. Kyoristu Shuppan Publishing Co., Tokyo.

Hamkins, J. D. (2007). A survey of infinite time Turing machines. In Durand-Lose, J. and Margenstern, M., editors, *Machines, Computations and Universality (MCU '07)*, number 4664 in LNCS, pages 62–71. Springer.

Hogarth, M. L. (2004). Deciding arithmetic using SAD computers. *Brit. J. Philos. Sci.*, 55 :681–691.

Huckenbeck, U. (1989). Euclidian geometry in terms of automata theory. *Theoret. Comp. Sci.*, 68(1) :71–87.

Jacopini, G. and Sontacchi, G. (1990). Reversible parallel computation: an evolving space-model. *Theoret. Comp. Sci.*, 73(1) :1–46.

Margenstern, M. and Morita, K. (2001). Np problems are tractable in the space of cellular automata in the hyperbolic plane. *Theoret. Comp. Sci.*, 259(1–2) :99–128.

Moore, C. (1996). Recursion theory on the reals and continuous-time computation. *Theoret. Comp. Sci.*, 162(1) :23–44.

Naughton, T. J. and Woods, D. (2001). On the computational power of a continuous-space optical model of computation. In Margenstern, M., editor, *Machines, Computations, and Universality (MCU '01)*, volume 2055 of *LNCS*, pages 288–299.

Păun, G. (2002). *Membrane Computing. An Introduction*. Springer.

Păun, G., Rozenberg, G., and Salomaa, A. (1998). *DNA Computing*. Springer.

Rozenberg, G., Bäck, T., and Kok, J. N., editors (2012). *Handbook of Natural Computing*. Springer-Verlag.

Stannett, M. (2013). Computation and spacetime structure. *IJUC*, 9(1-2) :173–184.

Syropoulos, A. (2010). *Hypercomputation*. Springer.

Weihrauch, K. (2000). *Introduction to computable analysis*. Texts in Theoretical computer science. Springer, Berlin.

Winfree, E. (2000). Algorithmic self-assembly of dna : Theoretical motivations and 2d assembly experiments. *Journal of Biomolecular Structure and Dynamics*, 2(11) :263–270.