

# Exact Discretization of 3-Speed Rational Signal Machines into Cellular Automata

Tom Besson<sup>(✉)</sup> and Jérôme Durand-Lose

University of Orléans, INSA Centre Val de Loire, LIFO EA 4022, Orléans, France  
`tom.besson@univ-orleans.fr`

**Abstract.** Cellular Automata (CA) operate in discrete time and space whereas Signal Machines (SM) have been developed as a continuous idealization of CA capturing the key concept of signals/particles and collisions. Inside a Euclidean space, dimensionless signals move freely; collisions are instantaneous. Today's issue is the automatic generation of a CA *mimicking* a given SM. On the one hand, many ad hoc manual conversions exist. On the other hand, some irrational or 4+-speed SM exhibit Zeno-like behaviors/space-time compression or rely on information being locally unbounded, both being incompatible with CA. This article provides a solution to automatically generate an exactly mimicking CA for a restricted class of SM: the ones that uses only three rational speeds, and rational initial positions. In these SM, signals are always contained inside a regular mesh. The discretization brings forth the corresponding discrete mesh. The simulation is valid on any infinite run and preserves the relative position of collisions.

**Keywords:** Abstract geometrical computation · Automatic discretization · Cellular automata · Signal machines · Unconventional models of computation

## 1 Introduction

Cellular automata (CA) are massively synchronous, uniform and local discrete dynamical systems. Since their introduction by J. Von Neumann in the forties [17], a lot of research, from dynamics to algorithmic, have been made. However, a recurrent problem is to create CA for a specific purpose, exhibiting a given behavior. Dedicated CA are crafted by experts, in the same way that programming a computer used to be reserved to specialists. As of now, coding became easier and easier with more and more abstract languages. What would these be such for CA?

The model chosen to serve this purpose is the signal machines (SM). Signal machines, originally thought as an idealization of CA, focus on continuous counterparts of (discrete) signals and collisions in CA. Signals and collisions are indeed key concepts of CA. They provide a lot of insight on the fabric of CA and

are the foundation of collision computing [1]. Some of their first usages stand in the generation of prime numbers by Fischer [13] and Goto's solution to the Firing Squad Synchronization Problem (FSSP) [14]. Signals are commonly used: to solve the FSSP [20], to compute with rule 110 [4] or with only four states [18], to understand one CA [5] and so on. In [19] one important question is the automatic positioning of discrete signals. Discrete signals are studied in [16].

In those references, as in many others, signals are represented by Euclidean lines to explain and reason as can be seen in Fischer's (Fig. 1(a)) and in Goto's (Fig. 1(b)) and Yunès's (Fig. 1(c)). Those constructions are understood in the continuum before being implemented in a discrete setting. Resulting states and transition functions are often left out because they are not used in the proof of constructions, obnoxious to establish, and cumbersome to read.

It is possible to *simulate* a CA with a SM [11]. However, the other way round is done only on particular ad hoc cases, often leaving the technical details out. The present paper provides an automatic conversion for a sub-class of signal machines: 3-speed rational. After a normalization phase, the states and the transition function are generated as well as initial configurations.

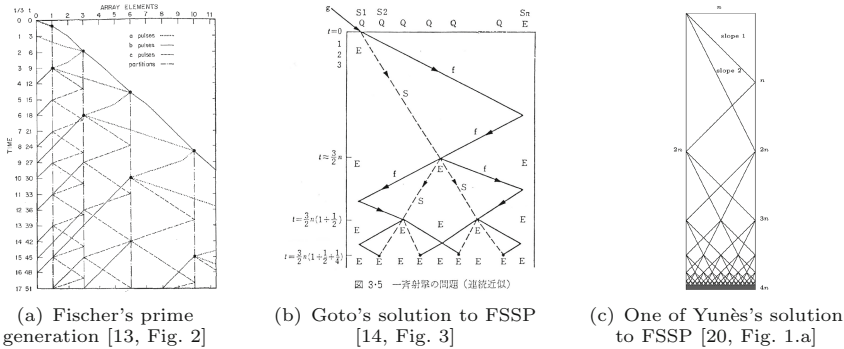
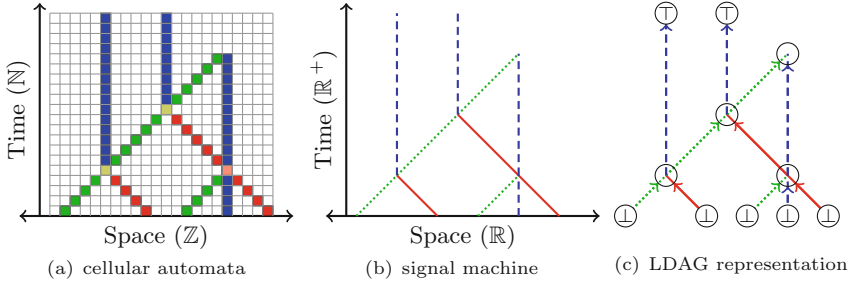


Fig. 1. Examples of signal use in designing cellular automata.

Signal machines are the formal tool for thinking about CA in the continuum and are inspired by the idealisation of discrete signals. They form an autonomous dynamical model where *signals* are dimensionless points moving with constant speed. They are completely described by their positions and natures called *meta-signals*. Starting with a finite number of signals on the real axis, when two or more meet, they are destroyed and new signals are emitted. A set of *collision rules* defines which signals are emitted according to the colliding meta-signals. The dynamics is represented on a space-time diagram where signals appear as segments as in Fig. 2(b). In previous articles, one of the authors researched on the possibilities of the model, from its computational power [9, 11], to its ability to simulate other models (like the BSS model [3] in [10]).

In signal machines, collisions are discrete steps related by signals: a collision is right before another if a signal generated by the first one ends in the second



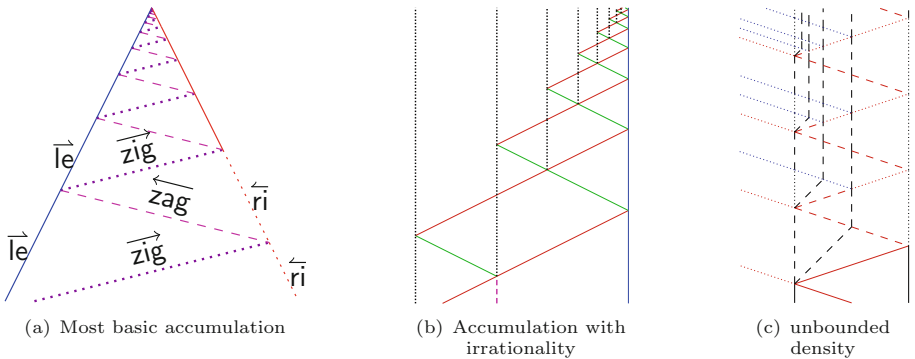
**Fig. 2.** Identification of runs through the labeled DAG representation.

one. The dynamics of a run of a SM lies entirely in this *causal order* which can be represented as a Labeled Directed Acyclic Graph (LDAG). Figure 2 provides a LDAG identification of discrete and continuous space-time diagrams.

The present paper considers 3-speed rational SM: only three possible speeds are available and speeds as well as positions of signals in initial configurations are rational numbers. The behavior of these machines is limited since signals are trapped inside a regular, periodic-like *mesh* [2, 12] which is essentially discrete (ensuring the discretization).

The existence of such meshes is not guaranteed with four or more speeds or irrationality. Moreover, if discretization is always locally possible it is not at a large scale nor for an infinite computation. Zeno-like behaviors/space-time compression known as *accumulation* (which are highly unpredictable [8]) leads to an infinite number of “objects” in bounded portion of a space-time diagram as depicted in Fig. 3(a) and (b).

An accumulation-less space-time diagram may not be discretizable into a CA as shown the one in Fig. 3(c). On the left, one solid signal is sent after 0, 1, 2, 3 . . . dotted ones. This means that any number can be “stored” in-between



**Fig. 3.** Non discretizable space-time diagrams.

the solid vertical signals. Discretization would lead to a finite number of cells and a bounded “storage capacity”.

Outside the case covered in the present article, discretization is only possible on special cases designed for with an extra piece of information (whatever it may be) ensuring the process to work.

The automatic conversion starts with a normalization to get simple speeds. Then meta-signals and collision rules are turned into states and transitions. The translation of configurations follows the same patterns: normalization into integer positions then conversion. Special care is taken so that the discrete signals evolve inside a discrete mesh. The correctness of the process comes from the “preservation” of it and the LDAG inside it.

In Sect. 2, the SM and CA models are defined as well as SM-meshes and CA-signals. Section 3 focuses on dynamics and simulation as well as normalization of SM. Section 4 describes the discretization: generated states and transitions and initial configurations. Section 5 deals with the correctness of the construction through CA-meshes and LDAG. Section 6 gathers concluding remarks.

## 2 Definitions and Properties

A signal machine regroups the definitions of its meta-signals and their dynamics: constant speed outside of collisions and rewriting rules at collisions.

**Definition 1.** A *signal machine* (SM),  $\mathfrak{A}$ , is a triplet  $(M, S, R)$  such that:  $M$  is a finite set of *meta-signals*;  $S : M \rightarrow \mathbb{R}$  is the *speed function* (each meta-signal has a constant speed); and  $R$  is a finite set of *collision rules* written  $\rho = \rho^- \rightarrow \rho^+$  where  $\rho^-$  and  $\rho^+$  are sets of meta-signals of distinct speeds. Each  $\rho^-$  must have at least two meta-signals.  $R$  is deterministic:  $\rho \neq \rho'$  implies  $\rho^- \neq \rho'^-$ .

If no collision rule is defined for a set of meta-signals, the collision is *blank*: the same meta-signals are output ( $\rho^+ = \rho^-$ ).

A  $(\mathfrak{A}-)$ *configuration*,  $\mathfrak{c}$ , is a mapping from the real line to either a meta-signal, a collision rule or the value  $\emptyset$  indicating that there is nothing there. There are finitely many non- $\emptyset$  locations.

If there is a signal of speed  $s$  at  $x$ , then after a duration  $\Delta t$  its position is  $x + s \cdot \Delta t$ , unless it enters a collision before. At a collision, all incoming signals are immediately replaced according to collision rules by outgoing signals in the following configurations.

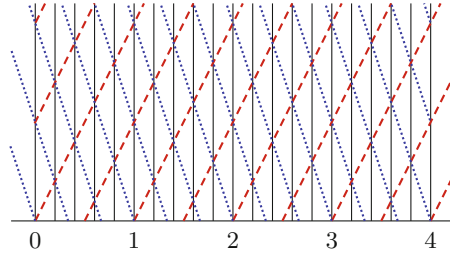
A *space-time diagram* is the collection of consecutive configurations forming a two dimensional picture (time is always elapsing upwards as in Fig. 2).

**Definition 2.** A signal machine is *3-speed rational* (3S-Q) if: only three speeds are available for the meta-signals and all speeds are rational numbers as well as any non- $\emptyset$  position in any initial configuration.

These machines have their signals trapped inside a mesh made of the union of half-lines following the three speeds like the one in Fig. 4.

**Definition 3.** Let  $p, q$  and  $n$  be positive integers,  $p$  and  $q$  relatively prime, the  $(p, q, n)$ - $(SM)$ -mesh corresponds to the union of the following half-lines of  $\mathbb{R} \times \mathbb{R}^+$ :

- $\mathfrak{V}_v : 0 \leq t$  and  $x = v/(p+q)$  where  $v \in \{0, 1, 2, \dots, n(p+q)\}$ ,
- $\mathfrak{L}_l : x \leq n$  and  $x = (l - t)/q$  where  $l \in \mathbb{N}$ , and
- $\mathfrak{R}_r : 0 \leq x$  and  $x = (t - r)/p$  where  $r \in \{(-n.p), \dots, -1, 0, 1, \dots\}$ .



**Fig. 4.** The  $(2, 3, 4)$ -mesh.

**Properties 1** ([2, Lemma 1] and [12, Lemma 1]). *In the space-time diagram generated from a  $3s - \mathbb{Q}$  signal machine with speeds  $-\frac{1}{q}, 0$  and  $\frac{1}{p}$ ,  $p$  and  $q$  relatively prime, on an initial configuration where non- $\emptyset$  values are in  $\{0, 1, \dots, n\}$ , all the non- $\emptyset$  positions belong to the  $(p, q, n)$ - $(SM)$ -mesh.*

In the  $(p, q, n)$ -mesh, the  $(SM)$ -encounter of coordinates  $(v, r)$ ,  $\mathbf{e}_v^r$ , is the intersection of  $\mathfrak{V}_v$ ,  $\mathfrak{R}_r$  and  $\mathfrak{L}_{p+q}$ ; that is  $(\frac{v}{p+q}, r + \frac{p}{p+q}v)$ . Encounter  $\mathbf{e}_v^{r+1}$  directly depends on signals coming from  $\mathbf{e}_{v-1}^{r+1}$ ,  $\mathbf{e}_v^r$ , and  $\mathbf{e}_{v+1}^{r-1}$  (if they exist) or the initial configuration. The integers  $v$  and  $r$  are such that  $0 \leq v \leq n(p+q)$  and  $-n.q \leq r$ . Encounters ordered according to dependencies form a well founded order used for inductive proofs.

**Definition 4.** A (1-dimensional radius-1) *cellular automaton* (CA) is a triplet  $(Q, f, \#)$  such that:  $Q$  is a finite set of *states*,  $f : Q^3 \rightarrow Q$  is the local *transition function*, and  $\#$  is a special state such that  $f(\#, \#, \#) = \#$  (the *quiescent state*). A *configuration* maps cells to states, i.e. it is an element of  $Q^{\mathbb{Z}}$ . In the evolution from a configuration  $c$ , the *site*,  $c_x^t$ , is the cell  $x \in \mathbb{X}$  at time-step  $t \in \mathbb{N}$ . It is computed by  $c_x^{t+1} = f(c_{x-1}^t, c_x^t, c_{x+1}^t)$ .

**Definition 5.** Let  $W = M \cup R$  be the set of meta-signals and collision rules of a signal machine  $\mathfrak{A}$  and  $Q$  be the set of states of a CA  $\mathcal{A}$ . A  $\mathfrak{A}$ - $\mathcal{A}$  representation relation  $\mathcal{R}$  relates  $W$  and  $Q$  with the intended meaning that  $\lambda \mathcal{R} q$  iff  $\lambda$  is *represented by*  $q$ . A state representing a collision rule does not represent anything else. A meta-signal can be represented by many states and a state can represent more than one meta-signal. A state representing nothing corresponds to  $\emptyset$ .

**Definition 6.** For any meta-signal  $\mu$ , a  $\mu$ -CA-signal is defined by  $(x, a, b, \varphi)$  where:  $x \in \mathbb{Z}$  is the *base position*,  $a, b \in \mathbb{N}$  with  $a \leq b$  ( $b$  can be  $+\infty$ ) are the *birth* and *death* dates, and  $\varphi \in [0, 1)$  is the *phase*. It corresponds to the set of sites:

- $\{ (x + \lfloor \varphi + (t - a) \cdot S(\mu) \rfloor, t) \mid t \in [a, b] \}$ , if  $0 < S(\mu)$  (rightward signal),
- $\{ (x - \lfloor \varphi - (t - a) \cdot S(\mu) \rfloor, t) \mid t \in [a, b] \}$ , if  $S(\mu) < 0$  (leftward signal), or
- $\{ (x, t) \mid t \in [a, b] \}$ , otherwise (stationary signal).

It must represent  $\mu : \mu \mathcal{R}c_x^t$  for all  $(x, t)$  in the  $\mu$ -CA-Signal. Moreover it should be maximal (not extendable as a  $\mu$ -CA-Signal). Its *speed* is the one of  $\mu$ .

After the normalization, the non-stationary speeds are of the form  $\pm 1/d$ . Any non-stationary CA-signal has a simple periodic dynamics: it moves one cell on the side every  $d$  iterations. Figure 5 provides examples of CA-signals.

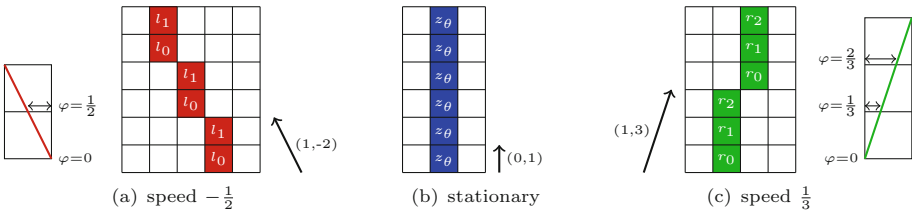


Fig. 5. Examples of CA-signals and phases.

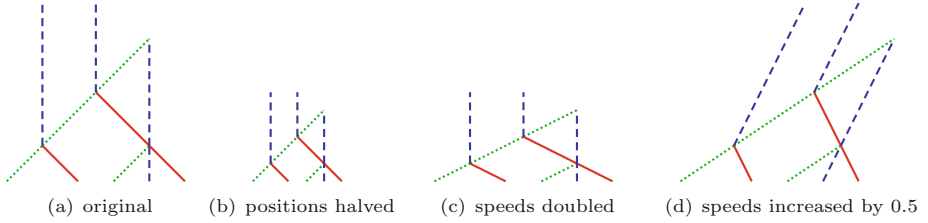
For rightward signals, the phase  $\varphi$  is the distance between the bottom left corner of the cell and the place where the (continuous) signal enters the cell at the bottom. If the signal is leftward; the phase is the distance to the lower right corner of the cell. Phases are illustrated in Fig. 5.

### 3 Dynamics, Simulation and Normalization

The *causal order* of collisions in a space-time diagram can be represented as a Labeled Directed Acyclic Graph (LDAG) as illustrated in Fig. 2(c). If the LDAG's of two runs are identical, then the runs are *dynamic-wise* identical.

Formally, a  $\mathfrak{A}$ -DAG is a LDAG where edges (resp. vertices) are labeled with elements of  $M$  (resp.  $R \cup \{\perp, \top\}$ ). Bottom (resp. top) leaves are exactly the vertices labeled with  $\perp$  (resp.  $\top$ ). The label  $\perp$  is used for signals present in the initial configuration. The label  $\top$  is used for never-ending signals. The LDAG generated from an initial configuration  $\mathfrak{c}$  is denoted  $\widehat{\mathfrak{c}}$ .

A signal machine  $\mathfrak{B}$  *simulates* a SM  $\mathfrak{A}$  if there exist a *conversion function*  $\zeta : \mathcal{C}_{\mathfrak{A}} \rightarrow \mathcal{C}_{\mathfrak{B}}$  and a *relabeling function*  $\psi : M_{\mathfrak{B}} \cup R_{\mathfrak{B}} \rightarrow M_{\mathfrak{A}} \cup R_{\mathfrak{A}}$  such that:  $\forall \mathfrak{c} \in \mathcal{C}_{\mathfrak{A}}, \widehat{\mathfrak{c}} = \psi(\widehat{\zeta(\mathfrak{c})})$  where  $\psi$  is canonically extended to LDAG as relabeling each edge and vertex.



**Fig. 6.** Examples of linear transformations.

Linearly (with positive coefficients) changing the speeds of the meta-signals or the initial positions does not affect the dynamics [7, Chap. 5]. Examples of such transformations are provided in Fig. 6.

A 3-speed rational signal machine is normalized with a positive linear operation so that its speeds are  $-\frac{1}{q}, 0, \frac{1}{p}$  with  $p$  and  $q$  relatively prime. From now on,  $p$  and  $q$  are always used to refer to these denominators. Similarly, the positions in initial configurations are lifted onto  $\mathbb{N}$ .

Let  $\mathcal{R}$  be a  $\mathfrak{A}$ - $\mathcal{A}$  representation relation, and  $c$  a  $\mathcal{A}$ -configuration. The LDAG  $\widehat{c}^{\mathcal{R}}$ , as illustrated in Fig. 2(c), is formed as follows. The edges correspond to all the CA-signals in the space-time diagram. The vertices correspond to all the sites that represent a collision rule plus one  $\perp$  vertex for each CA-signal present in the initial configurations and one  $\top$  vertex for each infinite CA-signal. The edges are in-incident to vertices if the vector from the topmost site of the signal to the vertex site is  $(-1, 1)$ ,  $(0, 1)$  or  $(1, 1)$  (or an infinite CA-signal and the dedicated  $\top$  vertex). The edges are out-incident to vertices if the vector from the bottom-most site of the signal to the vertex site is  $(-1, -1)$ ,  $(0, -1)$  or  $(1, -1)$  (or present in the initial configuration and the dedicated  $\perp$  vertex). The labels correspond to  $\mu$  for  $\mu$ -CA-signal and the (unique) collision rule represented by the vertex. If the LDAG is ill formed or non-unique, then  $\widehat{c}^{\mathcal{R}}$  is undefined.

A cellular automaton  $\mathcal{A}$  simulates a SM  $\mathfrak{A}$  if there exists a conversion function  $\zeta : \mathcal{C}_{\mathfrak{A}} \rightarrow \mathcal{C}_{\mathcal{A}}$  and a  $\mathfrak{A}$ - $\mathcal{A}$  representation relation  $\mathcal{R}$  such that:  $\forall c \in \mathcal{C}_{\mathfrak{A}}, \widehat{c} = \widehat{\zeta(c)}^{\mathcal{R}}$ .

Dynamic-wise simulation preserves collision interactions but discard anything relevant to positions. Nothing prevents the resulting space-time diagrams to be “bent” (as in [6]) and one might want to preserve the geometry up to some linear operator (like *grouping* for CA in [15]).

To cope with this, we say that a simulation is *geometry-preserving* if the positions of the CA-collisions can be computed from the one of the corresponding SM-collision with a linear function (plus some integral rounding). The construction presented here is geometry-preserving.

## 4 Formal Discretization

The notation 

	d	
a	b	c

 defines the CA-transition  $f(a, b, c) \rightarrow d$ . The quiescent state # is left blank in the pictures. Any non-specified transition results

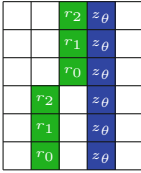
in #. It is understood that  $r$  and  $r'$  are any rightward meta-signals,  $l$  and  $l'$  are any leftward ones and  $z$  and  $z'$  are any stationary ones.

The following states and transitions ensure the movement of isolated signals as illustrated in Fig. 5:

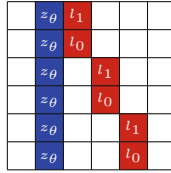
- for each (rightward)  $r$ : states  $r_k$  with  $0 \leq k < p$  and  $\forall k, 0 \leq k < p - 1$ :  $\begin{array}{|c|c|c|} \hline & r_{k+1} & \\ \hline \# & r_k & \# \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & r_0 & \\ \hline r_{p-1} & \# & \# \\ \hline \end{array}$ ,
- for each (leftward)  $l$ : states  $l_m$  with  $0 \leq m < q$  and  $\forall k, 0 \leq k < q - 1$ :  $\begin{array}{|c|c|c|} \hline & l_{k+1} & \\ \hline \# & l_k & \# \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & l_0 & \\ \hline \# & \# & l_{q-1} \\ \hline \end{array}$ ,
- for each (stationary)  $z$ : state  $z_\theta$  and  $\begin{array}{|c|c|c|} \hline & z_\theta & \\ \hline \# & z_\theta & \# \\ \hline \end{array}$ . The subscript indicates a specific phase  $\theta = \frac{q}{p+q}$  used to position exactly the collision.

When signals are moving closer as illustrated in Fig. 7, the following transitions are defined:

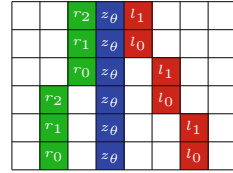
- $\forall k, 0 \leq k < p - 1$ :  $\begin{array}{|c|c|c|} \hline & r_{k+1} & \\ \hline \# & r_k & z_\theta \\ \hline \end{array}$ ,  $\begin{array}{|c|c|c|} \hline & z_\theta & \\ \hline r_k & z_\theta & \# \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & r_0 & \\ \hline r_{p-1} & \# & z_\theta \\ \hline \end{array}$ .
- $\forall k, 0 \leq k < q - 1$ :  $\begin{array}{|c|c|c|} \hline & l_{k+1} & \\ \hline z_\theta & l_k & \# \\ \hline \end{array}$ ,  $\begin{array}{|c|c|c|} \hline & z_\theta & \\ \hline \# & z_\theta & l_k \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & l_0 & \\ \hline z_\theta & \# & l_{q-1} \\ \hline \end{array}$ .



(a) rightward and stationary



(b) leftward and stationary



(c) three speeds

**Fig. 7.** Examples of  $\mathfrak{A}$ -CA-Signals closing on each other.

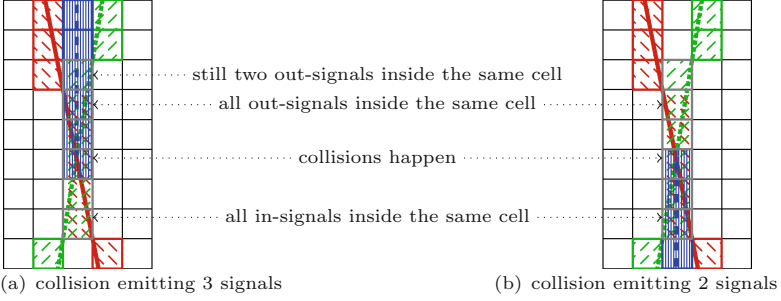
Let  $\rho : \{r, z, l\} \rightarrow \{l', z', r'\}$  be any collision rule (any single  $r, z, l$  could be missing and any  $r', z', l'$  could be missing). When signals enter the same cell, the following phase hypothesis is assumed: non-stationary signals start with phase 0 and stationary ones with phase  $\theta$ . The continuous signals are considered to locate precisely the collision. They correspond to the lines  $y = px$  (rightward),  $y = -qx + q$  (leftward) and  $x = \theta$  (stationary). Resolving this system gives:  $x = \frac{q}{p+q}$  and  $y = \frac{p \cdot q}{p+q}$ . The collisions always happen inside a cell because  $\frac{p \cdot q}{p+q} \notin \mathbb{Z}$  (since  $p$  and  $q$  are relatively prime). The delay from getting in the same cell to the actual collision is denoted  $\kappa_0$ .

The states  $\rho_k$  represent the steps of the collision when two or more signals are inside the same cell:  $\rho_k$  means  $(r_k : z_\theta : l_k)$  when  $k < \kappa_0$  and  $(r'_k : z'_\theta : l'_k)$  when  $\kappa_0 < k$ . Let  $\kappa_1$  be the last time-step at which all the out-signals are inside the same cell and  $\kappa_2$  the last time-step at which two out-signals are inside the same cell. Their values are given on Table 1 and illustrated in Fig. 8.



**Table 1.** Main steps of a collision.

$\kappa_0 = \frac{p \cdot q}{p + q}$	Emitted signals $z'$ and $r'$		$l'$ and $z'$		$l'$ and $r'$		$z'$ and $r'$	
	$\kappa_1$	$p - 1$	$q - 1$	$\min(p, q) - 1$				
$\kappa_2$	/	/	/		$\max(p, q) - 1$			

**Fig. 8.** Illustration of  $\kappa_0$ ,  $\kappa_1$  and  $\kappa_2$  with  $p = 6$  and  $q = 5$ .

The signals enter the same cell (under the hypothesis) with  $\begin{array}{|c|c|c|} \hline & \rho_0 & \\ \hline \# & z_\theta & l_{q-1} \\ \hline \end{array}$ ,  $\begin{array}{|c|c|c|} \hline & \rho_0 & \\ \hline r_{p-1} & \# & l_{q-1} \\ \hline \end{array}$ , or  $\begin{array}{|c|c|c|} \hline & \rho_0 & \\ \hline r_{p-1} & z_\theta & l_{q-1} \\ \hline \end{array}$ . Then, they are getting closer until the exact collision:  $\forall k, 0 \leq k < \kappa_0$ ,  $\begin{array}{|c|c|c|} \hline & \rho_{k+1} & \\ \hline \# & \rho_k & \# \\ \hline \end{array}$ .

If the collision emits no signal,  $\begin{array}{|c|c|c|} \hline & \# & \\ \hline \# & \rho_{\kappa_0} & \# \\ \hline \end{array}$  is added.

If the collision emits one signal as illustrated in Fig. 9(a), add:

- if  $\rho^+ = \{r'\}$ :  $\begin{array}{|c|c|c|} \hline & r'_0 & \\ \hline \rho_{\kappa_0} & \# & \# \\ \hline \end{array}$  if  $\kappa_0 + 1 = p$ , and  $\begin{array}{|c|c|c|} \hline & r'_{\kappa_0+1} & \\ \hline \# & \rho_{\kappa_0} & \# \\ \hline \end{array}$  otherwise;
- if  $\rho^+ = \{l'\}$ :  $\begin{array}{|c|c|c|} \hline & l'_0 & \\ \hline \# & \# & \rho_{\kappa_0} \\ \hline \end{array}$  if  $\kappa_0 + 1 = q$ , and  $\begin{array}{|c|c|c|} \hline & l'_{\kappa_0+1} & \\ \hline \# & \rho_{\kappa_0} & \# \\ \hline \end{array}$  otherwise;
- if  $\rho^+ = \{z'\}$ :  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline \# & \rho_{\kappa_0} & \# \\ \hline \end{array}$ .

If the collision emits more than one signal, it progresses until a signal leaves:

$\forall k, \kappa_0 \leq k < \kappa_1$ :  $\begin{array}{|c|c|c|} \hline & \rho_{k+1} & \\ \hline \# & \rho_k & \# \\ \hline \end{array}$ .

If the collision emits two signals as illustrated in Fig. 9(b):

- if  $\rho^+ = \{z', r'\}$ , add:  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline \# & \rho_{\kappa_1} & \# \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & r'_0 & \\ \hline \rho_{\kappa_1} & \# & \# \\ \hline \end{array}$ ;
- if  $\rho^+ = \{l', z'\}$ , add:  $\begin{array}{|c|c|c|} \hline & l'_0 & \\ \hline \# & \# & \rho_{\kappa_1} \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline \# & \rho_{\kappa_1} & \# \\ \hline \end{array}$ ;
- if  $\rho^+ = \{l', r'\}$ ,
- if  $q < p$  ( $q < p$  is symmetrical), add:  $\begin{array}{|c|c|c|} \hline & l'_0 & \\ \hline \# & \# & \rho_{\kappa_1} \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & r'_{\kappa_1+1} & \\ \hline \# & \rho_{\kappa_1} & \# \\ \hline \end{array}$ .

- if  $q = p = 1$ , add:  $\begin{array}{|c|c|c|} \hline & l'_0 & \\ \hline \# & \# & \rho_{\kappa_1} \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & r'_0 & \\ \hline \rho_{\kappa_1} & \# & \# \\ \hline \end{array}$ .

If the collision emits three signals, in the case  $q \leq p$  (the rest is symmetric) as illustrated in Fig. 9(c), the following is added:

- $\begin{array}{|c|c|c|} \hline & l'_0 & \\ \hline \# & \# & \rho_{\kappa_1} \\ \hline \end{array}$ , and  $\begin{array}{|c|c|c|} \hline & r'_0 & \\ \hline \rho_{\kappa_2} & \# & \# \\ \hline \end{array}$ ;
- let  $j = \kappa_2 - \kappa_1 - 1$ ,  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline l'_j & \rho_{\kappa_2} & \# \\ \hline \end{array}$  if  $0 < j < q$ , and  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline \# & \rho_{\kappa_2} & \# \\ \hline \end{array}$  otherwise;
- for  $\kappa_1 < k < \kappa_2$ :  $\begin{array}{|c|c|c|} \hline & \rho_{k+1} & \\ \hline l'_j & \rho_k & \# \\ \hline \end{array}$  for  $j = k - \kappa_1 - 1$  and  $j < q$  and  $\begin{array}{|c|c|c|} \hline & \rho_{k+1} & \\ \hline \# & \rho_k & \# \\ \hline \end{array}$  for  $\kappa_1 + q < k$ ;
- for  $0 \leq j < q - 1$ :  $\begin{array}{|c|c|c|} \hline & l'_{j+1} & \\ \hline \# & l'_j & \rho_k \\ \hline \end{array}$  for  $k = j + \kappa_1 + 1$  and  $k \leq \kappa_2$ .

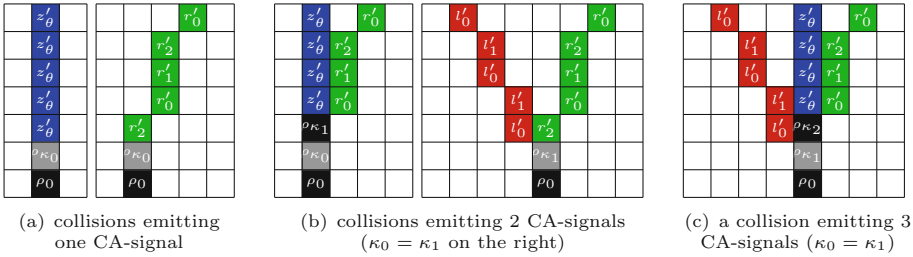


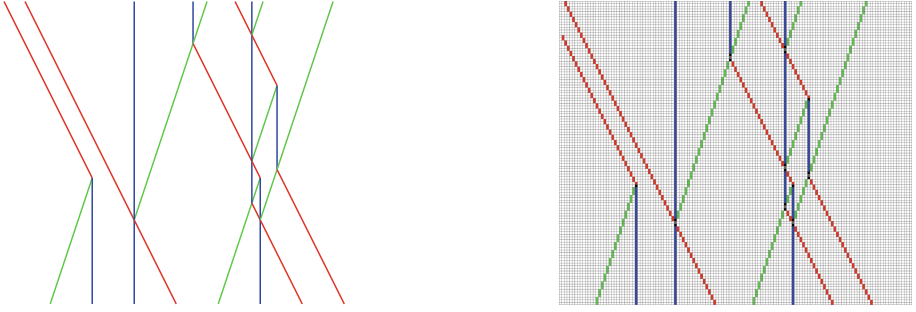
Fig. 9. Collisions and CA-Signals getting away from each other with  $p = 3$  and  $q = 2$ .

After the collision, signals move away from each other as illustrated in Fig. 9. This is handled with the transitions:

- $\forall k, 0 \leq k < p - 1$ :  $\begin{array}{|c|c|c|} \hline & r'_{k+1} & \\ \hline z'_\theta & r'_k & \# \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline \# & z'_\theta & r'_k \\ \hline \end{array}$ ,
- $\forall k, 0 \leq k < q - 1$ :  $\begin{array}{|c|c|c|} \hline & l'_{k+1} & \\ \hline \# & l'_k & z'_\theta \\ \hline \end{array}$  and  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline l'_k & z'_\theta & \# \\ \hline \end{array}$ , and
- $\forall k, m, 0 \leq m < q - 1$  and  $0 \leq k < p - 1$ :  $\begin{array}{|c|c|c|} \hline & z'_\theta & \\ \hline l'_m & z'_\theta & r'_k \\ \hline \end{array}$ .

A scale is used to ensure that the continuous mesh is correctly discretized: spaces between stationary signals should be integers as well as times of collisions. From the definition of encounter, any (positive) multiple of  $p + q$  is enough. Let  $\delta = 3(p + q)$  be the *discretization scale*.

Let  $\mathbf{c}$  be an initial configuration of the discretized SM. The corresponding *initial CA-configuration*,  $c_0$ , is defined by:  $\forall i \in \mathbb{Z}, c_0(i) = \psi(\mathbf{c}(i/\delta))$  where  $\psi(\circ) = \#$ ,  $\psi(z) = z_\theta$  for any stationary meta-signal, and  $\psi(\mu) = \mu_0$  for any other. This ensures the validity of the phase hypothesis at the first round of collisions (Fig. 10).



**Fig. 10.** A continuous space-time diagram and the generated CA discretization.

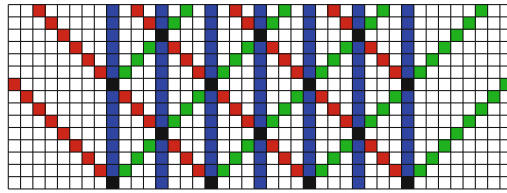
## 5 Correctness

In this section,  $q \leq p$  is assumed (the rest follows by symmetry).

A discrete mesh is the discrete counterpart of the continuous meshes as illustrated in Fig. 11. The  $(\delta, p, q, n)$ -CA-Mesh corresponds to the union of:

- $V_v = \{3v\} \times \mathbb{N}$  with  $v \in \{0, 1, 2, \dots, n(q+p)\}$ , ( $3 = \delta/(q+p)$ ),
- $L_l = \left\{ (x, t) \mid 0 \leq t \vee x \leq n\delta \vee x = \left\lceil \frac{l\delta - t}{q} \right\rceil \right\}$  with  $l \in \mathbb{N}$ ,
- $R_r = \left\{ (x, t) \mid 0 \leq t \vee 0 \leq x \vee x = \left\lfloor \frac{t - r\delta}{p} \right\rfloor \right\}$  with  $r \in \{(-n.p), \dots, -1, 0, 1, \dots\}$ .

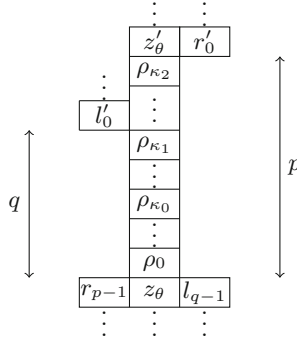
Each set corresponds to a CA-signal (not necessary issued from the initial configuration). They can overlap for consecutive iterations (as in Figs. 9 and 12).



**Fig. 11.** The  $(4,1,1,3)$ -CA-Mesh ( $\delta = 8 = 4(1+1)$ ).

In the  $(\delta=3(p+q), p, q, n)$ -(CA-)mesh, the *CA-encounter* of coordinates  $(v, r)$ ,  $c_v^r$ , corresponds to:  $\left( \left\lfloor \frac{\delta}{p+q}v + \frac{q}{p+q} \right\rfloor, \left\lfloor r\delta + \frac{p\delta}{p+q}v + \frac{pq}{p+q} \right\rfloor \right) = (3v, 3pv + r\delta + \kappa_0)$  where the  $(v, r)$  coordinates are integers such that  $0 \leq v \leq n(p+q)$  and  $-n.q \leq r$ . The last term in the formula comes from the phases.

Figure 12 depicts an entire (CA-)collision. Under the hypothesis that the  $\kappa_0$  site is located at some  $e_v^r$ ,  $\rho_0$  has coordinates  $(3v, 3pv + r\delta)$  and thus corresponds to the SM-encounter  $\epsilon_v^r$  at scale  $\delta$ . This relates discrete and continuous meshes.



**Fig. 12.** Entire generic collision  $\rho = \{r, z, l\} \rightarrow \{l', z', r'\}$ .

With the same hypothesis, the following holds:  $\rho_0$  belongs to  $V_v, R_r$  and  $L_{v+r}$ ,  $\rho_{\kappa_2}$  belongs to  $R_r$ , and  $\rho_{\kappa_1}$  belongs to  $L_{v+r}$ . This fixes all three CA-signals over a period. The whole collision is in the  $(\delta, p, q, n)$ -CA-Mesh. As can be seen in Fig. 12, meta-signals extend out of the collision from both ends so that CA-signals directly connect into.

The hypothesis is always satisfied as can be proved by induction: CA-signals in the initial configuration are in the CA-Mesh with the right phases and if CA-signals from the mesh collide, so are the collisions and the resulting CA-signals. Altogether, it proves that:

**Theorem 1.** *All the non-quiescent sites of the run from the simulating CA-configuration are located inside the  $(\delta, p, q, n)$ -CA-Mesh. Up to scaling, this mesh coincides with the continuous  $(p, q, n)$ -SM-mesh.*

The  $\mathfrak{A}$ - $\mathfrak{A}$  representation relation is defined straightforwardly from the construction. Let  $l, z$  and  $r$  be any leftward, stationary and rightward meta-signals:

$$\forall i, 0 \leq i < q, l\mathcal{R}l_i \quad z\mathcal{R}z_\theta \quad \forall i, 0 \leq i < p, r\mathcal{R}r_i .$$

Let  $\rho = \{r, z, l\} \rightarrow \{l', z', r'\}$  be any rule.

$$\begin{aligned} \forall i, 0 \leq i < \kappa_0, r\mathcal{R}\rho_i, \quad z\mathcal{R}\rho_i, \text{ and } l\mathcal{R}\rho_i, \\ \rho\mathcal{R}\rho_{\kappa_0}, \\ \forall i, \kappa_0 \leq i < \kappa_1, r'\mathcal{R}\rho_i, \quad z'\mathcal{R}\rho_i, \text{ and } l'\mathcal{R}\rho_i, \\ \forall i, \kappa_1 < i \leq \kappa_2, r'\mathcal{R}\rho_i \text{ and } z'\mathcal{R}\rho_i. \end{aligned}$$

A structural induction on the mesh proves that the SM and CA LDAG's are identical, so that:

**Theorem 2.** *Any 3-speed rational signal machine can be simulated exactly by a cellular automaton on any infinite run preserving the geometry.*

## 6 Conclusion

The exact conversion of a 3-speed rational signal machine into a cellular automata has been proved and implemented in Java. The construction relies on preserving a dynamics that is trapped inside a discrete-like mesh. Moreover this is valid on any infinite run and preserves the relative position of collisions.

This result is tight: as soon as either irrationality [2] or four speeds is allowed, Zeno-like phenomena (infinitely many collisions in finite duration) is possible and there is no hope for exact discretization. Even if the techniques presented here can be extended to any number of (rational) speeds, there is no information on how long the discretization remains valid. We believe that testing the validity for a given amount of time is as complex as running the signal machine and its simulation.

The construction presented here naturally extends to any SM computation that is constrained to remain on a mesh (we do not expect such a property to be decidable whether the mesh is provided or not in the general case). One perspective is to either work with some extra piece of information proving the containment into a mesh or the non-existence of an accumulation (which is highly non decidable [8]) or other problematic behavior. One important step would be to discretize rational signal machines with four or more speeds known to be Turing complete without accumulating as in [11].

Another perspective is to provide approximation. The quality of the approximation would have to be defined and if possible a bound guaranteed.

## References

1. Adamatzky, A. (ed.): Collision Based Computing. Springer, London (2002)
2. Becker, F., Chapelle, M., Durand-Lose, J., Levorato, V., Senot, M.: Abstract geometrical computation 8: Small machines, accumulations & rationality (2013, submitted). <http://arxiv.org/abs/1307.6468>
3. Blum, L., Shub, M., Smale, S.: On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines. *Bull. Amer. Math. Soc.* **21**(1), 1–46 (1989)
4. Cook, M.: Universality in elementary cellular automata. *Complex Syst.* **15**, 1–40 (2004)
5. Crutchfield, J.P., Mitchell, M., Das, R.: The evolutionary design of collective computation in cellular automata. In: Crutchfield, J.P., Schuster, P.K. (eds.) *Evolutionary Dynamics-Exploring the Interplay of Selection, Neutrality, Accident, and Function*, pp. 361–411. Oxford University Press, New York (2003)
6. Durand-Lose, J.: Reversible space-time simulation of cellular automata. *Theor. Comp. Sci.* **246**(1–2), 117–129 (2000)
7. Durand-Lose, J.: Calculer géométriquement sur le plan - machines à signaux. Habilitation à Diriger des Recherches, École Doctorale STIC, Université de Nice-Sophia Antipolis (2003, in French)
8. Durand-Lose, J.: Forecasting black holes in abstract geometrical computation is highly unpredictable. In: Cai, J.-Y., Cooper, S.B., Li, A. (eds.) *TAMC 2006*. LNCS, vol. 3959, pp. 644–653. Springer, Heidelberg (2006)

9. Durand-Lose, J.: Reversible conservative rational abstract geometrical computation is Turing-universal. In: Beckmann, A., Berger, U., Löwe, B., Tucker, J.V. (eds.) CiE 2006. LNCS, vol. 3988, pp. 163–172. Springer, Heidelberg (2006)
10. Durand-Lose, J.: Abstract geometrical computation and the linear Blum, Shub and Smale Model. In: Cooper, S.B., Löwe, B., Sorbi, A. (eds.) CiE 2007. LNCS, vol. 4497, pp. 238–247. Springer, Heidelberg (2007)
11. Durand-Lose, J.: Abstract geometrical computation 4: small Turing universal signal machines. *Theor. Comp. Sci.* **412**, 57–67 (2011)
12. Durand-Lose, J.: Irrationality is needed to compute with signal machines with only three speeds. In: Bonizzoni, P., Brattka, V., Löwe, B. (eds.) CiE 2013. LNCS, vol. 7921, pp. 108–119. Springer, Heidelberg (2013)
13. Fischer, P.C.: Generation of primes by a one-dimensional real-time iterative array. *J. ACM* **12**(3), 388–394 (1965)
14. Goto, E.: *Ōtomaton ni kansuru pazuru* [Puzzles on automata]. In: Kitagawa, T. (ed.) *Jōhōkagaku eno michi* [The Road to information science], pp. 67–92. Kyoristu Shuppan Publishing Co., Tokyo (1966)
15. Mazoyer, J., Rapaport, I.: Inducing an order on cellular automata by a grouping operation. In: Meinel, C., Morvan, M. (eds.) STACS 1998. LNCS, vol. 1373, pp. 116–127. Springer, Heidelberg (1998)
16. Mazoyer, J., Terrier, V.: Signals in one-dimensional cellular automata. *Theor. Comp. Sci.* **217**(1), 53–80 (1999)
17. Neumann, J.: *Theory of Self-Reproducing Automata*. University of Illinois Press, Urbana (1966)
18. Ollinger, N., Richard, G.: Four states are enough!. *Theor. Comp. Sci.* **412**(1–2), 22–32 (2011)
19. Richard, G.: *Systèmes de particules et collisions discrètes dans les automates cellulaires*. Ph.D. thesis, Aix-Marseille Université (2008)
20. Yunès, J.-B.: Simple new algorithms which solve the firing squad synchronization problem: a 7-states  $4n$ -steps solution. In: Durand-Lose, J., Margenstern, M. (eds.) MCU 2007. LNCS, vol. 4664, pp. 316–324. Springer, Heidelberg (2007)