Forecasting black holes in Abstract geometrical computation is highly unpredictable

Jérôme Durand-Lose*

Laboratoire d'Informatique Fondamentale d'Orléans, Université d'Orléans, B.P. 6759, F-45067 ORLÉANS Cedex 2.

Abstract. In *Abstract geometrical computation for black hole computation (MCU '04, LNCS 3354)*, the author provides a setting based on rational numbers, abstract geometrical computation, with super-Turing capability: any recursively enumerable set can be decided in finite time. To achieve this, a Zeno-like construction is used to provide an accumulation similar in effect to the black holes of the black hole model.
We prove here that forecasting an accumulation is Σ_2^0-complete (in the arithmetical hierarchy) even if only energy conserving signal machines are addressed (as in the cited paper). The Σ_2^0-hardness is achieved by reducing the problem of deciding whether a recursive function (represented by a 2-counter automaton) is strictly partial. The Σ_2^0-membership is proved with a logical characterization.

Key-words. Abstract geometrical computation, Accumulation forecasting, Arithmetical hierarchy, Black hole model, Energy conservation, Super-Turing computation, Turing universality, Zeno phenomena.

1 Introduction

The foundations of computability are currently being questioned as many super-Turing models of computation are being unveiled. Some models use analog or hybrid settings [AM95,Bou99,Bra95], infinite computations [EN93,Ham02,HL00], or black holes [EN02,LN04]. To our knowledge, there are only few researches on the whereabouts of the artifacts providing super-Turing capability. In this article, we are interested in providing an example where the phenomenon used, albeit being easy to generate, is not easy to forecast. More precisely, we are interested in the black hole embedding in (rational) signal machines as described in [DL05]. In this paper, the author shows how to produce and use an accumulation to provide the black hole effect. We prove that it is undecidable to predict whether an accumulation will even happen, even when restricted to the conservative (ensuring some energy conservation) signal machines: it is Σ_2^0-complete in the arithmetical hierarchy. On the one hand, this means that it is not even semi-decidable, but on the other hand it is not so bad considering that it is the key to decide recursively enumerable problems.

* `http://www.univ-orleans.fr/lifo/Members/Jerome.Durand-Lose`, `Jerome.Durand-Lose@univ-orleans.fr`

Abstract geometrical computation considers Euclidean lines. The support of space and time is $\mathbb{R}$. Computations are produced by *signal machines* which are defined by finite sets of *meta-signals* and of *collision rules*. Signals are atomic information, corresponding to meta-signals, moving at constant speed thus generating Euclidean line segments on space-time diagrams. Collision rules are pairs *(incoming meta-signals, outgoing meta-signals)*, that define a mapping over sets of meta-signals. A configuration is a mapping from $\mathbb{R}$ to meta-signals, collision rules, and two special values: void (*i.e.* nothing there) and accumulations (amounting for black holes). The time scale is $\mathbb{R}^+$; there is no such thing as a "next configuration". The following configurations are defined by the uniform movement of signals. In the configurations following a collision, incoming signals are replaced by outgoing signals according to a collision rule.

Zeno like acceleration and accumulation can be constructed as on Fig. 2 of Sect. 2. This provides the black hole-like artifact for deciding $\mathcal{R}.\mathcal{E}.$ problems. But accumulations can lead to an uncontrolled burst of signals producing infinitely many signals in finite time (as in the right of Fig. 2). To avoid this, a *conservativeness* condition is imposed: a positive energy is defined for every meta-signal, the sum of these energies must be conserved by each rule. Thus no energy creation is possible; the number of signals is bounded.

Abstract geometrical computation (AGC) comes from the common use, in the literature on *cellular automata* (CA), of Euclidean lines to model discrete lines in space-time diagrams of CA (*i.e.* colorings of $\mathbb{Z} \times \mathbb{N}$ with states as on the left of Fig. 1) to access dynamics or to design. The main characteristics of CA, as well as abstract geometrical computation, are: *parallelism, synchronicity, uniformity* and *locality* of updating. Discrete lines are often observed and idealized as on Fig. 1. They can be the keys to understanding the dynamics like in [Ila01, pp. 87–94] or [BNR91,JSS02]. They can also be the tool to design CA for precise purposes like Turing machine simulation [LN90] or reversible simulation [DL97]. These discrete line systems have also been studied on their own [MT99,DM02].

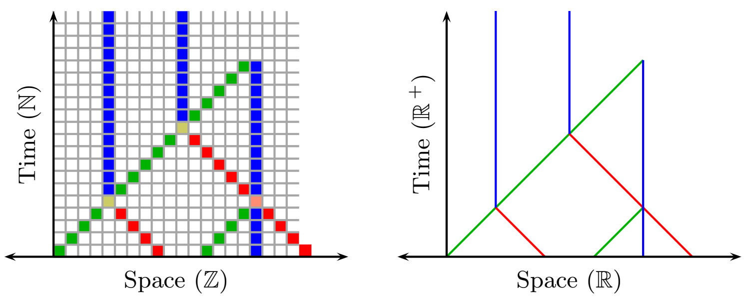


Fig. 1. Space-time diagram of a cellular automaton and its signal machine counterpart.

To our knowledge, AGC is the only computing model that is a dynamical system with continuous time and space but finitely many local values. The closest model we know of is the Mondrian automata of Jacopini and Sontacchi [JS90]. Their space-time diagrams are mappings from $\mathbb{R}^n$ to a finite set of colors representing bounded finite polyhedra. Another close model is the piecewise-constant derivative system [AM95,Bou99]: $\mathbb{R}^n$ is partitioned into finitely many polygonal regions; trajectories are defined by a constant derivative on each region and form sequences of (Euclidean) line segments.

In this paper, space and time are restricted to rational numbers. This is possible since all the operations used preserve rationality. All quantifiers and intervals should be understood over $\mathbb{Q}$, not $\mathbb{R}$. Since rational numbers can be implemented exactly on a computer (which is impossible for real numbers), decision problems concerning AGC can be expressed in classical computability.

It was proved in [DL05] that any 2-counter automaton can be simulated by a conservative (rational) signal machine. In the same article, a conservativeness preserving construction to embed this simulation into an accumulation is provided. We modify this construction so that the accumulation does not take place when the simulation stops. This provides a reduction of the Halting problem and Σ_1^0-hardness. We then provide a higher lever structure that tries all possible initial values, one after the other. This way, if a computation never stops then an accumulation happens, otherwise no accumulation will even happen. This is a reduction of the problem of deciding whether a recursive function is total or not, which is Σ_2^0-complete. Each construction preserves conservativeness.

Signal machines are defined in Sect. 2. The forecasting decision problems, arithmetical hierarchy and 2-counter automata are presented in Sect. 3. In Sect. 4 we prove the Σ_1^0-hardness and then its Σ_2^0-hardness of Conservative-AGC-accumulation-Forecasting. The membership of the general case (AGC-accumulation-Forecasting) is proved by a logical characterization in Sect. 5. Conclusion and perspective are gathered in Sect. 6.

2 Abstract geometrical computations

Abstract geometrical computations are defined by the following machines:

Definition 1 A *(rational) signal machine* is defined by (M, S, R) where M (*meta-signals*) is a finite set, S (*speeds*) a mapping from M to $\mathbb{Q}$, and R (*collision rules*) a partial mapping from the subsets of M of cardinality at least 2 into the subsets of M (speeds must differ in both domain and range).

Each instance of a meta-signal is a *signal*. The mapping S assigns rational *speeds* to meta-signals, which corresponds the slopes of the segments in space-time diagrams. The *collision rules*, denoted $\rho^- \to \rho^+$, define what happens when two or more signals meet.

The *extended value set*, V, is the union of M and R plus two symbols: one for void, $\oslash$, and one for an accumulation (or black hole) ❋. A *configuration*, c, is a total mapping from $\mathbb{Q}$ to V such that the set $\{\, x \in \mathbb{Q} \mid c(x) \neq \oslash \,\}$ is finite.

A signal corresponding to a meta-signal μ at a position x, *i.e.* $c(x) = \mu$, is moving uniformly with constant speed $S(\mu)$. A signal must start (resp. end) in the initial (resp. final) configuration or in a collision. This corresponds to condition 2 in Def. 2. At a $\rho^- \to \rho^+$ collision, all, and only, signals corresponding to the meta-signals in ρ^- (resp. ρ^+) must end (resp. start); no other signal should be present (condition 3). A black hole corresponds to an accumulation of collisions and disappears without a trace (condition 4).

Let S_{min} and S_{max} be the minimal and maximal speeds. The *causal past*, or *light-cone*, arriving at position x and time t, $J^-(x,t)$, is defined by all the positions that might influence the information at (x,t) through signals, formally:

$$J^-(x,t) = \{\ (x',t') \mid x - S_{max}(t-t') \leq x' \leq x - S_{min}(t-t')\ \}\ .$$

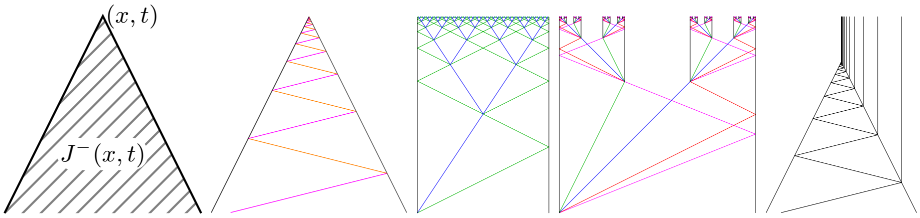


Fig. 2. Light-cone, a simple accumulation and three unwanted phenomena.

Definition 2 The *space-time diagram* issued from an initial configuration c_0 and lasting for T, is a mapping c from $[0,T]$ to configurations (*i.e.* a mapping from $\mathbb{Q} \times [0,T]$ to V) such that, $\forall (x,t) \in \mathbb{Q} \times [0,T]$:

1. $\forall t \in [0,T]$, $\{\, x \in \mathbb{Q} \mid c_t(x) \neq \oslash \,\}$ is finite,
2. if $c_t(x) = \mu$ then $\exists t_i, t_f \in [0,T]$ with $t_i < t < t_f$ or $0 = t_i = t < t_f$ or $t_i < t = t_f = T$ s.t.:
   - $\forall t' \in (t_i, t_f)$, $c_{t'}(x + S(\mu)(t'-t)) = \mu$,
   - $t_i = 0$ or $c_{t_i}(x_i) \in R$ and $\mu \in (c_{t_i}(x_i))^+$ where $x_i = x + S(\mu)(t_i - t)$,
   - $t_f = T$ or $c_{t_f}(x_f) \in R$ and $\mu \in (c_{t_f}(x_f))^-$ where $x_f = x + S(\mu)(t_f - t)$;
3. if $c_t(x) = \rho^- \to \rho^+ \in R$ then $\exists \varepsilon$, $0 < \varepsilon$, $\forall t' \in [t-\varepsilon, t+\varepsilon] \cap [0,T]$, $\forall x' \in [x-\varepsilon, x+\varepsilon]$,
   - $c_{t'}(x') \in \rho^- \cup \rho^+ \cup \{\oslash\}$,
   - $\forall \mu \in M, c_{t'}(x') = \mu \Rightarrow \bigvee \begin{cases} \mu \in \rho^- \text{ and } t' < t \text{ and } x' = x + S(\mu)(t'-t)) \ , \\ \mu \in \rho^+ \text{ and } t < t' \text{ and } x' = x + S(\mu)(t'-t)) \ ; \end{cases}$
4. if $c_t(x) =$ ❊ then
   - $\exists \varepsilon > 0$, $\forall (x',t') \notin J^-(x,t)$, $(\ |x - x'| < \varepsilon \text{ and } |t - t'| < \varepsilon\ ) \Rightarrow c_{t'}(x) = \oslash$,
   - $\forall \varepsilon > 0$, $\{\ (x',t') \in J^-(x,t) \mid t - \varepsilon < t' < t \wedge c_{t'}(x') \in R\ \}$ is infinite.

On the illustrating space-time diagrams, time is always increasing upwards. The three space-time diagrams of Fig. 2 provide examples un-compatible with Def. 2 at the time of accumulation. In each case, the number of signals is bursting to infinity and black holes are not isolated. To prevent this, the following restriction is imposed.

Definition 3 A signal machine is *conservative* when an atomic positive energy is defined for all meta-signals ($E : M \to \mathbb{N}^*$) such that the total energy of the system is preserved, *i.e.* the sum of all the energy of existing signals is a constant of the system. This is equivalent to have each rule preserving the energy: the sum of the energy of incoming meta-signals equals the sum of outgoing ones.

It follows automatically that given a conservative signal machine and an initial configuration, the number of signals in any following configuration, as well as the number of accumulations, is bounded (by the total energy divided by the least atomic energy).

3 Decision problems, arithmetical hierarchy and 2-counter automata

Instance AGC-accumulation-Forecasting

$\mathcal{M}$: rational signal machine, and

c: (rational) configuration for $\mathcal{M}$.

Question

Is there any accumulation in the space-time generated by $\mathcal{M}$ from c?

Since rational numbers can be encoded by natural numbers, this problem is expressible in classical computability theory. The problem CONSERVATIVE-AGC-ACCUMULATION-FORECASTING is defined similarly but with an extra condition on $\mathcal{M}$: the machine must be conservative.

Arithmetical hierarchy deals with non recursive sets. It is defined by:

Definition 4 A set S belongs to Σ_n^0 if it can be defined by a logic formula consisting of a total recursive predicate preceded by an alternation of n universal/existential quantifiers over a numerable set starting with an existential quantifier:

$$S \in \Sigma_{2k}^0 \iff S = \{x | \exists n_1, \forall n_2, \exists n_3 \ldots \forall n_{2k}, \phi(x, n, n_2 \ldots n_{2k})\} \ ,$$
$$S \in \Sigma_{2k+1}^0 \iff S = \{x | \exists n_1, \forall n_2, \exists n_3 \ldots \exists n_{2k+1}, \phi(x, n, n_2 \ldots n_{2k+1})\} \ ,$$

where ϕ is a recursive total predicate.

Thus Σ_0^0 are Σ_1^0 the set of respectively recursive and recursively enumerable sets. To address a decision problem, the set of positive instances is considered. A set is said to be complete in a class if and only if it belongs to the class and any problem of its class can be many-one-reduced to it (*i.e.*, there is a recursive function mapping positive –resp. negative– instances of the first problem into the positive –resp. negative– instances of the second one). The halting problem is Σ_1^0-complete. The following problem is Σ_2^0-complete [Odi99, p. 621]: given a Turing-machine, is there an entry such that the computation never stops. This corresponds to deciding whether a recursive function is not total.

A 2-counter automaton is a finite automaton coupled with two counters, A and B. The possible actions on any counter are *add/subtract* 1 and *branch if non-zero*. These machines can be described with a six-operations assembly language with branching labels as on the left part of Fig. 4 (see [Min67] for more on 2-counter automata). Since Turing machines and 2-counter automata compute exactly the same functions, the following problem is also Σ_2^0-complete:

Instance Not-Total-2CA

$\mathcal{A}$: 2-counter automaton.

Question

Is there an initial value such that the computation of $\mathcal{A}$ never stops?

4 Energy concerving case

4.1 Reduction from the halting problem

It is possible to simulate any 2-counter automaton with a conserving signal machine [DL05]. Figure 3 shows how the counters are encoded using two fixed signals zero and one as a scale. A signal amounting for the current line zigzags between these signals. Figure 4 presents the code of a simple 2-counter automaton and some simulations. When a simulation stops, a signal stop appears and is locked inside the ribbon, bouncing between zero and one.

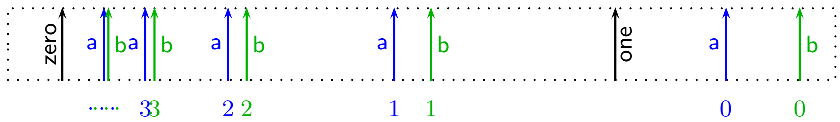


Fig. 3. Encoding positions of counters.

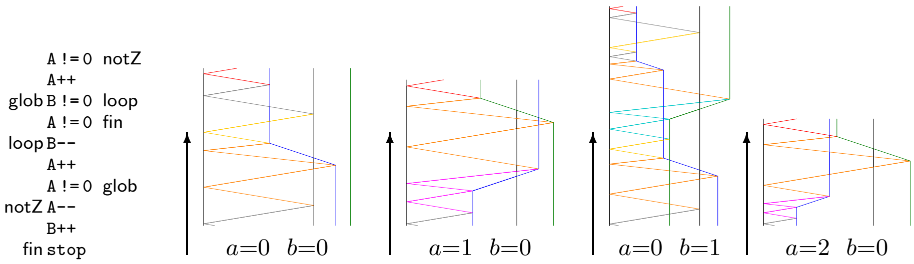
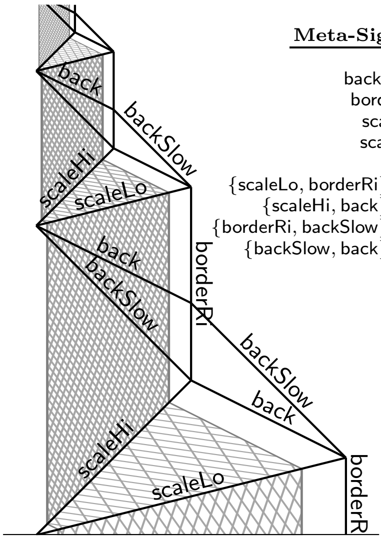


Fig. 4. 2-counter automaton $\mathcal{A}_{next}$ and its simulations for three different initial values.

Let us note that the automaton $\mathcal{A}_{next}$ on Fig. 4 computes the following function: if A is zero then $(A', B') = (B+1, 0)$ otherwise $(A', B') = (A-1, B+1)$. Starting from $(0, 0)$ successive applications of it generate all the elements of $\mathbb{N} \times \mathbb{N}$. On Fig. 4, each computation yields the counter values for the next. This is used in Subsect. 4.2 to start one after the previous finishes all the computations possible by any 2-counter automaton.

Figure 5 sketches the construction of a structure that allows to transform any spatially-bounded computation into another computation that is also temporally-bounded. The iterated shrinking structure always brings out an accumulation. On the example on the right of Fig. 5, a two-counter simulation is embedded inside this structure. This construction preserves conservativeness. Details can be found in [DL05,DL06].

Preventing the accumulation of the structure. To achieve this, the signal machine is modified so that stop erases and collects the energy of scaleHi, back and backSlow. The resulting signal stop8l leaves on the left side. Collecting starts when stop meets scaleHi. Signal stop collects it and all signals until it encounters



Meta-Signal	Speed	E
back	$-2\nu_0$	1
backSlow	$-\nu_0$	1
borderRi	0	1
scaleHi	ν_0	1
scaleLo	$4\nu_0$	1

{scaleLo, borderRi} $\longrightarrow$ {back, backSlow}
{scaleHi, back} $\longrightarrow$ {backSlow, borderRi}
{borderRi, backSlow} $\longrightarrow$ {back, borderRi}
{backSlow, back} $\longrightarrow$ {scaleHi, scaleLo}

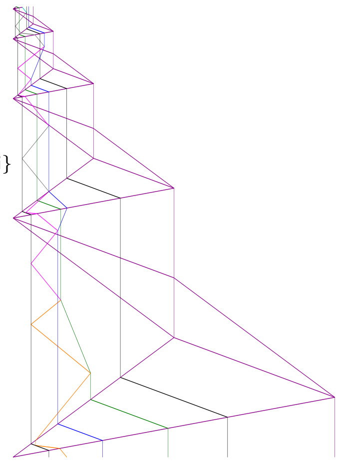
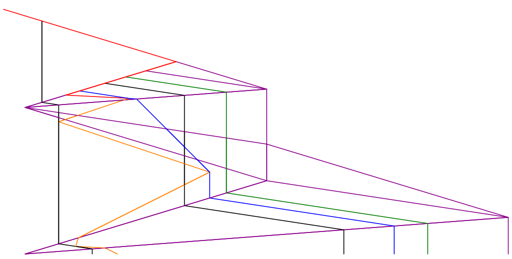


Fig. 5. Structure, meta-signals and rules for the iterated shrinking.

backSlow. When it collects backSlow, it turns back to collect all the remaining signals. This is done by adding the meta-signals and rules given on Fig. 6 (for clarity, we use the same name for unstrained strained signals).



Meta − signal	Speed	E
stop$i$r $(i \in 2..7)$	ν_0	i
stop$i$l $(i \in 3..8)$	$-\nu_0$	i

{scaleHi, stop} $\longrightarrow$ {stop2r}
{stop$i$r, a} $\longrightarrow$ {stop$(i{+}1)$r} $(i \in 2..7)$
{stop$i$r, b} $\longrightarrow$ {stop$(i{+}1)$r} $(i \in 2..7)$
{stop$i$r, one} $\longrightarrow$ {stop$(i{+}1)$r} $(i \in 2..7)$
{stop$i$r, back} $\longrightarrow$ {stop$(i{+}1)$r} $(i \in 2..7)$
{stop$i$r, backSlow} $\longrightarrow$ {stop$(i{+}1)$l} $(i \in 3..8)$
{stop$i$l, a} $\longrightarrow$ {stop$(i{+}1)$l} $(i \in 3..8)$
{stop$i$l, b} $\longrightarrow$ {stop$(i{+}1)$l} $(i \in 3..8)$
{stop$i$l, zero} $\longrightarrow$ {stop$(i{+}1)$l} $(i \in 3..8)$

Fig. 6. stop prevents the accumulation: example and modifications.

This provides a reduction from the Halting problem; Conservative-AGC-accumulation-Forecasting is Σ^0_1-hard.

4.2 Reduction from Not-Total-2CA

Let $\mathcal{A}$ be any 2-counter automaton. We provide a higher structure for iterating the previous construction. It is divided in two parts. On the left side there is a simulation of $\mathcal{A}_{next}$ (of Fig. 4) that holds and updates the initial values for $\mathcal{A}$. On the right side $\mathcal{A}$ starts with the initial configuration copied from $\mathcal{A}_{next}$ in a shrinking structure. If the simulation stops, then everything on the right side is erased and collected as in previous subsection.

At the beginning, the $\mathcal{A}_{next}$ simulation holds $(0, 0)$ and the copy process is launched. All the signals for simulating $\mathcal{A}$ and the shrinking structure are copied as in Fig. 7. As soon as signals are set in position, they act normally.

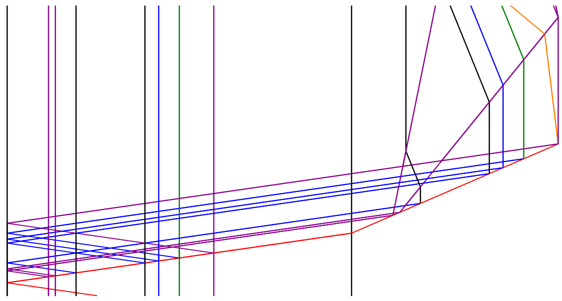


Fig. 7. Copying process (vertically stretched for clarity).

If the computation of $\mathcal{A}$ on (a_0, b_0) does not stop then the shrinking structure is not prevented from producing an accumulation. Otherwise, stop8l collects all the energy of both the structure and the simulation, preventing any accumulation. When stop8l reaches the left side, it restarts $\mathcal{A}_{next}$ which produces the next value of the enumeration of $\mathbb{N} \times \mathbb{N}$. This automaton always stops. The copying process and a new iteration start.

The copying process is conservative, the energy that has been gathered in stop8l is released bit by bit. All together, each and every counter initial values is tested one after the other, there is an accumulation as soon as there is a non halting $\mathcal{A}$-computation, otherwise all values are tested.

Lemma 5 Conservative-AGC-accumulation-Forecasting *is Σ_2^0-hard.*

5 Σ_2^0 membership of the general case

Lemma 6 AGC-accumulation-Forecasting *belongs to Σ_2^0.*

This is proved with the following expression of the problem matching Def. 4:

Lemma 7 *There is an accumulation in the space-time generated by $\mathcal{M}$ from c if and only if the following formula is true:*
$\exists (x,t) \in \mathbb{Z}\times\mathbb{N}, \forall n\in\mathbb{N}, \{$*there is at least n collisions in the casual past of* $(x, y)\}$.

The predicate "there is at least n collision..." is total and recursive: compute the collision in the light-cone until there are at least n or no more collision. This is a total and recursive predicate quantified by $\exists\forall$ over numerable sets.

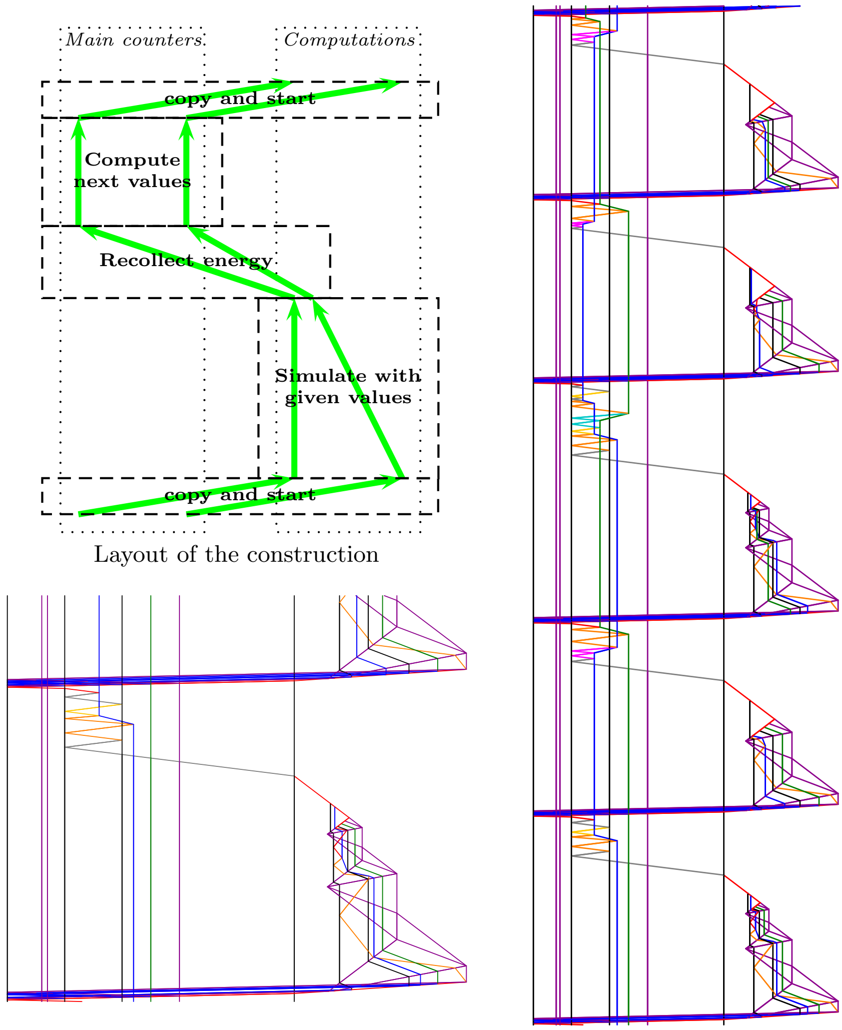


Fig. 8. Layout and first iterations.

6 Conclusion

Theorem 8 *Both* AGC-ACCUMULATION-FORECASTING *and* CONSERVATIVE-AGC-ACCUMULATION-FORECASTING *are Σ_2^0-complete.*

The process does never stop, but we are not interested in the halting problem but in the apparition of an accumulation. Accumulation is disconnected from infinite duration computation. We believe that even restrained to signal machines that preserve the number signal (each rule has as many in signals as out signals) and are reversible the problem is still Σ_2^0-complete.

References

[Ada02] A. Adamatzky, ed. *Collision based computing.* Springer, 2002.

[AM95] E. Asarin and O. Maler. Achilles and the Tortoise climbing up the arithmetical hierarchy. In *FSTTCS '95*, number 1026 in LNCS, pp. 471–483, 1995.

[BNR91] N. Boccara, J. Nasser, and M. Roger. Particle-like structures and interactions in spatio-temporal patterns generated by one-dimensional deterministic cellular automaton rules. *Phys. Rev. A*, 44(2):866–875, 1991.

[Bou99] O. Bournez. Achilles and the Tortoise climbing up the hyper-arithmetical hierarchy. *Theoret. Comp. Sci.*, 210(1):21–71, 1999.

[Bra95] M. S. Branicky. Universal computation and other capabilities of hybrid and continuous dynamical systems. *Theoret. Comp. Sci.*, 138(1):67–100, 1995.

[DL97] J. Durand-Lose. Intrinsic universality of a 1-dimensional reversible cellular automaton. In *STACS '97*, number 1200 in LNCS, pp. 439–450. Springer, 1997.

[DL05] J. Durand-Lose. Abstract geometrical computation for black hole computation (extended abstract). In M. Margenstern, ed., *Machines, Computations, and Universality (MCU '04)*, number 3354 in LNCS, pp. 176–187. Springer, 2005.

[DL06] J. Durand-Lose. Abstract geometrical computation 1: embedding black hole computations with rational numbers. *Fundamenta Informaticae*, 74(4):491–510, 2006.

[DM02] M. Delorme and J. Mazoyer. Signals on cellular automata. in [Ada02], pp. 234–275, 2002.

[EN93] J. Earman and J. D. Norton. Forever is a day: supertasks in Pitowsky and Malament-Hogarth spacetimes. *Philos. Sci.*, 60(1):22–42, 1993.

[EN02] G. Etesi and I. Nemeti. Non-Turing computations via Malament-Hogarth space-times. *Int. J. Theor. Phys.*, 41(2):341–370, 2002. gr-qc/0104023.

[Ham02] J. D. Hamkins. Infinite time Turing machines: Supertask computation. *Minds and Machines*, 12(4):521–539, 2002. arXiv:math.LO/0212047.

[HL00] J. D. Hamkins and A. Lewis. Infinite time turing machines. *J. Symb. Log.*, 65(2):567–604, 2000. arXiv:math.LO/9808093.

[Ila01] A. Ilachinski. *Cellular automata –a discrete universe.* World Scientific, 2001.

[JS90] G. Jacopini and G. Sontacchi. Reversible parallel computation: an evolving space-model. *Theoret. Comp. Sci.*, 73(1):1–46, 1990.

[JSS02] M. H. Jakubowsky, K. Steiglitz, and R. Squier. Computing with solitons: a review and prospectus. in [Ada02], pp. 277–297, 2002.

[LN90] K. Lindgren and M. G. Nordahl. Universal computation in simple one-dimensional cellular automata. *Complex Systems*, 4:299–318, 1990.

[LN04] S. Lloyd and Y. J. Ng. Black hole computers. *Scientific American*, 291(5):31–39, November 2004.

[Min67] M. Minsky. *Finite and infinite machines.* Prentice Hall, 1967.

[MT99] J. Mazoyer and V. Terrier. Signals in one-dimensional cellular automata. *Theoret. Comp. Sci.*, 217(1):53–80, 1999.

[Odi99] P. Odifreddi. *Classical Recursion Theory. Volume 2.* Number 143 in Studies in Logic and the Foundations of Mathematics. Elsevier, Amsterdam, 1999.