Geometrical accumulations and computably enumerable real numbers
(extended abstract)

Jérôme Durand-Lose*

LIFO, Université d'Orléans,
B.P. 6759, F-45067 ORLÉANS Cedex 2.

Abstract. Abstract geometrical computation involves drawing colored line segments (traces of signals) according to rules: signals with similar color are parallel and when they intersect, they are replaced according to their colors. Time and space are continuous and accumulations can be devised to unlimitedly accelerate a computation and provide, in a finite duration, exact analog values as limits.
In the present paper, we show that starting with rational numbers for coordinates and speeds, the time of any accumulation is a *c.e.* (*computably enumerable*) real number and moreover, there is a signal machine and an initial configuration that accumulates at any *c.e.* time. Similarly, we show that the spatial positions of accumulations are exactly the *d-c.e.* (*difference of computably enumerable*) numbers. Moreover, there is a signal machine that can accumulate at any *c.e.* time or *d-c.e.* position.

Key-words. Abstract geometrical computations; Computable analysis; Geometrical accumulations; *c.e.* and *d-c.e.* real numbers; Signal machine.

1 Introduction

Starting from a few aligned points, lines are initiated. When they intersect, they end and new line segments start. Each line is given a color and lines with the same color should be parallel. The new line segments are colored according to the colors of the removed ones.

What can kind of figure can one build with finitely many colors? Is this system computing in some way?

Such a system computes. It does so in the understandings of both *Turing* computability, the original *Blum, Shub and Smale* model [Blum et al., 1989, Durand-Lose, 2007, 2008a] and *Computable analysis* [Weihrauch, 2000, Durand-Lose, 2009b, 2010b]. The so-called *Black-hole model* of computation can be embedded too [Etesi and Németi, 2002, Hogarth, 2004, Lloyd and Ng, 2004, Andréka et al., 2009, Durand-Lose, 2006a, 2009a].

Given that the underlying space and time are Euclidean, thus continuous, can there be any accumulation? What can be said about them?

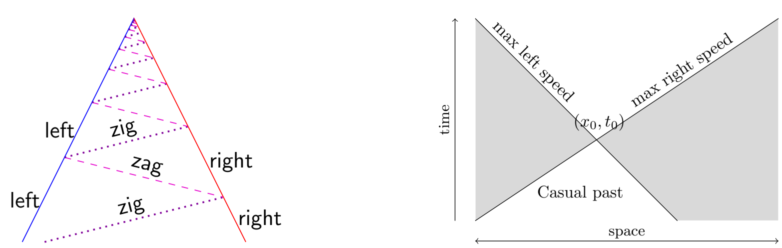
* `http://www.univ-orleans.fr/lifo/Members/Jerome.Durand-Lose`,
`Jerome.Durand-Lose@univ-orleans.fr`

Geometrical accumulation is a common phenomenon (as in Fig. 1, time is always elapsing upward). With a shift and a rescaling, it could happen anywhere. It is the key to embedding analog computing as well as the Black-hole model.

In the present paper, we show that if the system is based on rational numbers then the temporal and spatial coordinates of any isolated accumulation belong to some particular sets of real numbers. The times are exactly the *computably enumerable numbers* (*c.e.* numbers for short): the limits of (converging) increasing computable sequences of rational numbers. The spatial positions are exactly the differences of two such numbers (*d-c.e.* numbers). Usual *computable* numbers (limits of effectively converging computable sequences of rational) are a strict subset of *c.e.* numbers which is a strict subset of *d-c.e.* numbers [Zheng, 2006].

The geometric system described above is a *signal machine* in the context of *abstract geometrical computations*. It is inspired by a continuous time and space counterpart of cellular automata [Durand-Lose, 2008b] and related to the approaches of Jacopini and Sontacchi [1990], Takeuti [2005] and Hagiya [2005]. It is also an idealization of collision computing [Adamatzky, 2002, Adamatzky and Durand-Lose, 2010].

A *signal machine* gathers the definition of meta-signals (colors, like zig and right in Fig. 1(a)) and collision rules (like $\{\mathsf{zig}, \mathsf{right}\} \rightarrow \{\mathsf{zag}, \mathsf{right}\}$). An instance of a meta-signal is a dimensionless point called a *signal*. Each signal moves uniformly, its speed only depends on the associated meta-signal. The traces of signals on the space-time diagrams form line segments and as soon as they correspond to the same meta-signal, they are parallel. When signals meet, they are removed and replaced by new signals. The emitted signals only depend on the nature of colliding ones.



(a) Isolated accumulation (b) Casual past of backward light cone.

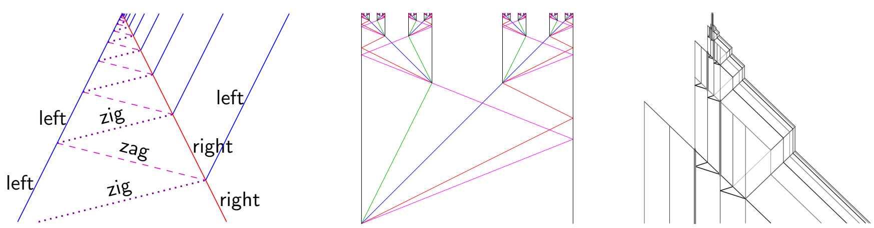
Fig. 1. Example of a space-time diagram and light cone.

One key feature of AGC is that space and time are continuous. This has been used to do fractal parallelism [Duchier et al., 2010]. Moreover, Zeno effects can be implemented to generate unbounded acceleration; in particular to allow infinitely many discrete transitions during a finite duration. This has been used to decide the halting problem and implementing the Black-hole model [Durand-Lose, 2009a]. It has also been used to carry out exact analog computations [Durand-Lose, 2008a, 2009b].

This is achieved with rational signal machines: speeds as well as initial positions are rational numbers. Since the positions of collisions are defined by linear equations in rational numbers, the collisions all happen at rational positions. This is important since rational numbers can be handled exactly in classical discrete computability.

One early question in the field was whether, starting from a rational signal machine, accumulation could lead to an irrational coordinate. An accumulation at $\sqrt{2}$ was provided in Durand-Lose [2007]. The question was then to characterize all the possible accumulation points. Please note that forecasting accumulation for a rational signal machine is as undecidable as the strict partiality of a computable function (Σ_2^0-complete in the arithmetical hierarchy [Durand-Lose, 2006b]).

In the present article, we are interested in *isolated accumulations*: in the space-time diagram, sufficiently close to it, there is no accumulation point and nothing except in the casual past as in Fig. 1(a). The accumulation on Fig. 2(a) is not isolated because of infinitely many left signals on the right. The ones on Fig. 2(b) form a cantor set and the ones of Fig. 2(c) are on a curve (right upper limit) and are almost all accumulations of signals away from any collision.



(a) Non isolated single accumulation (b) Accumulating on a Cantor set (c) Accumulating on a curve (upper right)

Fig. 2. Non-isolated accumulations.

The time of an accumulation is *c.e.* because the system can be simulated exactly on a computer and the time is the above limit of a Zeno phenomenon. For the spatial position, the difference is exhibited by slanting the space-time diagram to exhibit an increasing sequence, the drifting correction provide the negative *c.e.* term.

In Durand-Lose [2009b], we use a two level construction scheme where: the inner level simulates a TM that output orders to the outer structure. The outer structure undergoes a shrinking process —generating the accumulation— driven by the received orders. We use specially designed inner and outer structures to provide (rational) signal machines and initial configurations that accumulates at any *c.e.* time (resp. *d-c.e.* spatial position).

This paper goes beyond Durand-Lose [2010a] which ends up having a major flaw: the accumulation positions do not need to be computable.

Definitions are gathered in Sect. 2. Section 3 shows that the temporal (resp. spatial) coordinate of isolated accumulations is always *c.e.* (resp. *d-c.e.*). Section 4 presents the two level structure. Section 5 provides the control layer. Sections 6 and 7 present outer structures to accumulate respectively at a *c.e.* time and at a *d-c.e.* spatial position. Section 8 concludes the paper.

2 Definitions

2.1 Abstract geometrical computation

A *signal machine* collects the definitions of available meta-signals, their speed and the collision rules. For example, the machine to generate Fig. 1(a) is composed of the following meta-signals (with speed): left ($\frac{1}{2}$), zig (4), zag (-4), and right (-$\frac{1}{2}$). Two collision rules are defined:

$$\{\mathsf{left}, \mathsf{zag}\} \longrightarrow \{\mathsf{left}, \mathsf{zig}\} \quad \text{and} \quad \{\mathsf{zig}, \mathsf{right}\} \longrightarrow \{\mathsf{zag}, \mathsf{right}\} \ .$$

It might happen that exactly three (or more) meta-signals meet. In such a case, collisions rules involving three (or more) meta-signals are used. There can be any number of meta-signals in the range of a collision rule, as long as they speeds differ.

A *configuration* is a function from the real line (space) into the set of meta-signals and collision rules plus two extra values: $\oslash$ (for nothing there) and ✺ (for accumulation). If there is a signal of speed s at x, then, unless there is a collision before, after a duration Δt, its position is $x + s.\Delta t$. At a collision, all incoming signals are immediately replaced by outgoing signals in the following configurations according to collision rules. Moreover, any signal must be spatially isolated —nothing else arbitrarily closed—, locations with $\oslash$ value must form an open set and the accumulation points of non $\oslash$ locations must be ✺. (This is a spatial, static, accumulation like on Fig. 2(c).)

A *space-time diagram* is the collection of consecutive configurations which form a two dimensional picture. It must also verify that any accumulation point of collisions in the picture is ✺. (This is a dynamical accumulation like on Fig. 1(a).)

Considering the definition of light cone as on Fig. 1(b), an accumulation at (x_0, t_0) is *isolated* if, sufficiently close to (x_0, t_0):
- there is nothing but $\oslash$ out of the casual past, and
- there are infinitely many signals and collisions but no accumulation in the casual past.

It is a purely dynamical and local accumulation.

A signal machine is *rational* if all the speeds are rational (numbers) and only rational positions are allowed for signals in the initial configuration. Since the position of collisions are solutions of systems of rational linear equations, they are rational. In any space-time diagram of a rational signal machine, as long as there is no accumulation, the coordinates of all collisions are rational.

The dynamics is uniform in both space and time. Space and time are continuous; there is no absolute scale. So that if the initial configuration is shifted or scaled so is the whole space-time diagram.

2.2 *c.e.* and *d-c.e.* real numbers

A computable sequence is defined by a Turing machine that on input n output the nth term of the sequence.

Definition 1 (*c.e.* and *d-c.e.* numbers). *A real number is* c.e. *(*computably enumerable*) if there is an increasing computable sequence of rational numbers that converges to it.*

A real number is d-c.e. *(*difference of computably enumerable*) if it is the difference of two* c.e. *numbers.*

The *c.e.* numbers are closed by rational addition and positive rational multiplication but they not closed under subtraction. On the other side *d-c.e.* numbers form a closed field [Ambos-Spies et al., 2000] and are also characterized by:

Theorem 1 (Ambos-Spies et al. [2000]). *A real number is* d-c.e. *iff there is a computable sequence* (x_n) *that* weakly effectively *converges to it in the sense that the sum* $\sum_{n\in\mathbb{N}} |x_{n+1} - x_n|$ *converges.*

3 Only (*d*-)*c.e.* coordinates

Let us consider any (rational) isolated accumulation at (x_0, t_0). The configurations is "clipped" sufficiently closed to the accumulation so that there is nothing out of the casual past. It is rational and finite.

From a (rational) signal machine and a (finite) configuration, it is easy to build a Turing machine that treats the collisions and updates the configuration forever (and indeed this has been programmed in java). Each time a collision is treated, let it output the (rational) time. This sequence is increasing and converges to the time of the accumulation.

Lemma 1. *The time of any (rational) isolated accumulation is* c.e.

A space-time diagram can be slanted by adding the same "drift" to all signals. This is done by increasing all the speeds by the same amount. For example, starting from Fig. 1(a), by adding 1, 2 and 4 to all the speeds, the diagrams of Fig. 3 are generated.

With a sufficiently large integer drift, all speed become positive, so that the configuration has to move to the right, or at least, the positions of its leftmost signal is. Consider a modification of above Turing machine so that each time it treats a collision, it outputs the spatial position of the leftmost signal. This produces an increasing sequence that converges to the spatial position of the drifted accumulation. This position, y_0 is *c.e.*

While the times remain unchanged, the spatial positions of signals and collisions (and hence accumulation) are moved by $d.t$ where d is the drift and t is the time. To correct the drift for the accumulation, $d.t_0$ has to be removed where t_0 is the time of the accumulation. Since t_0 is *c.e.* and d is an integer, $d.t_0$ is *c.e.* So that $x_0 = y_0 - d.t_0$ is *d-c.e.*

With a sufficiently large negative drift, a decreasing converging sequence is generated. This generates the opposite of a *c.e.* (a *co-c.e.*) real number.

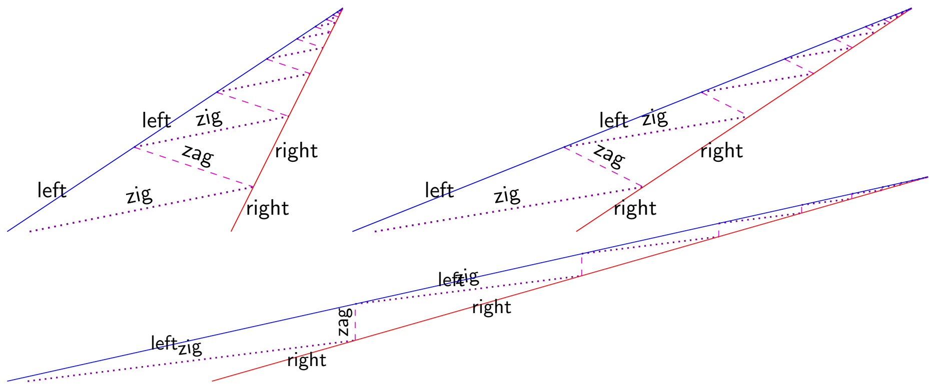


Fig. 3. Examples of drifts by 1, 2 and 4.

Lemma 2. *The spatial position of any (rational) isolated accumulation is* d-c.e. *The coordinates of an isolated accumulation can be expressed as* $(y - d.t_0, t_0)$ *or* $(d.t_0 - y', t_0)$ *where* y, y' *and* t_0 *are* c.e. *and* d *is an integer.*

4 Controlled shrinking structure

This section presents a general scheme based on a two level structure. The outer one handles the shrinking of the whole structure according to the messages received from the inner one which acts as a control.

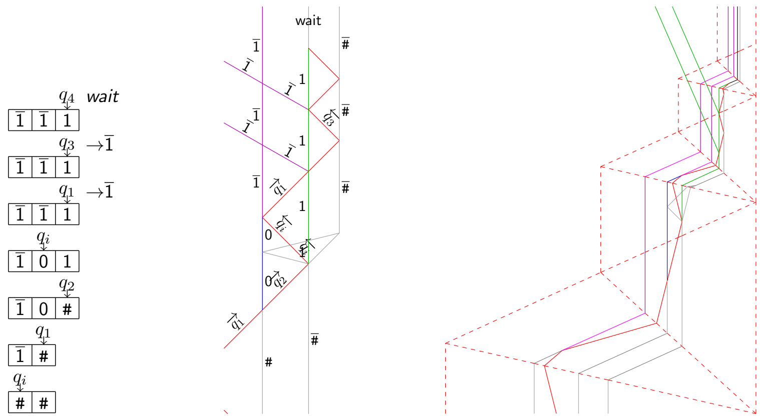
The shrinking process is not detailed here (it is in Durand-Lose [2010b]). It works by steps. Each step ensures that the structure and everything that is embedded inside it is scaled down by a constant factor. Three such steps are displayed in Fig. 4(c). This is repeated forever generating an accumulation.

4.1 Control/inner structure

The inner structure simulates a Turing machine that outputs orders to the outer structure. Each order is formed by signals sent on the left. Outputting is blocking: the control has to receive some acknowledgement signal to resume and send the next order.

The computation is embedded inside a shrinking structure to ensure a bounded delay between outputs. The shirking process is also blocked after the output and resume on relaying the acknowledgement. This is done to ensure that this structure does not generate an accumulation.

Simulating a Turing machine with a signal machine is only exemplified by figures 4(a) and 4(b) where $\overline{11}$ is output. The cells of the tape are encoded by motionless signals (vertical lines) displayed in a geometrical sequence so that it works in a bounded space. A more detailed construction can be found in [Durand-Lose, 2010b].



(a) run (b) simulation (c) whole inner structure

Fig. 4. Blocking inner structure including a Turing machine.

The output leaves on the left, unaffected by the inner shrinking. The latter is stopped by the presence of wait. The output is collected and processed by the outer structure.

4.2 Outer structure

The outer structure waits a fixed time before collecting and processing the next output. So that the inner structure has a limited activation time for outputting. Since it has also a shrinking structure, the Turing machine has an unlimited number of iterations ahead of it — this is a form of unbounded acceleration to ensure the output comes in due time. (Special case is taken so that the inner structure have space as if it would never halt.)

When an order is issued, the inner structure is blocked with all its signal parallel. Parallel signals are easy to move preserving their relative position as illustrated on Fig. 4(c) where 0-speed signals amounting for the tape cells go from one triangle to the next. Each time their distances are scaled by one half (this is the shrinking scheme). This is more exemplified in the next sections where special constructions are provided.

This structure provides the isolated accumulation but it moves or waits at each step so that to make the accumulation happen at some position according to the control.

5 Controls for *d-c.e.*

Let x be any *d-c.e.* number. There is a Turing machine that generates a sequence x_n such that this sequence converges to x and $\sum_{n\in\mathbb{N}} |x_{n+1} - x_n|$ converges. If x is *c.e.*, it is also requested that the sequence is increasing.

Let α be any positive rational number. Let us define the sequences:

$$y_0 = \left\lfloor \frac{1}{\alpha} x_0 \right\rfloor \qquad y_{n+1} = \left\lfloor \frac{2^{n+1}}{\alpha}(x_{n+1} - x_n + e_n) \right\rfloor \tag{1}$$

$$e_0 = x_0 - \alpha.y_0 \qquad e_{n+1} = x_{n+1} - x_n + e_n - \frac{\alpha}{2^{n+1}} y_{n+1} \tag{2}$$

where $\lfloor u \rfloor$ is the greatest integer less or equal to u ($\lfloor u \rfloor \leq u < \lfloor u \rfloor + 1$). It follows that

$$|e_n| \leq \frac{\alpha}{2^n} \ . \tag{3}$$

So that e_n converges to 0 and the sequence defined below, z_n, converges to x because, using (2),

$$\begin{aligned} z_n = \sum_{i=0}^{n} \frac{\alpha}{2^i} y_i &= x_0 - e_0 + \sum_{i=1}^{n} (-e_i + x_i - x_{i-1} + e_{i-1}) \\ &= x_n - e_n \ . \end{aligned}$$

Since x_n is a computable sequence of rational numbers, so are y_n and e_n. Moreover y_n is a sequence of integers. If x is a positive *c.e.*, then y_n should a sequence of natural numbers.

In the following, the control output y_n in unary (with $\overline{1}$ for negative values and with 1 for positive values) then waits. This loop is repeated forever.

If x is *c.e.*, then the computable sequence is increasing, so that only non-negative values are output.

6 Accumulating at a *c.e.* time

On the nth iteration, the outer structure receives y_n (an unary encoded natural number). It waits y_n time a delay and then shrinks the whole configuration by one half and wakes up the inner control.

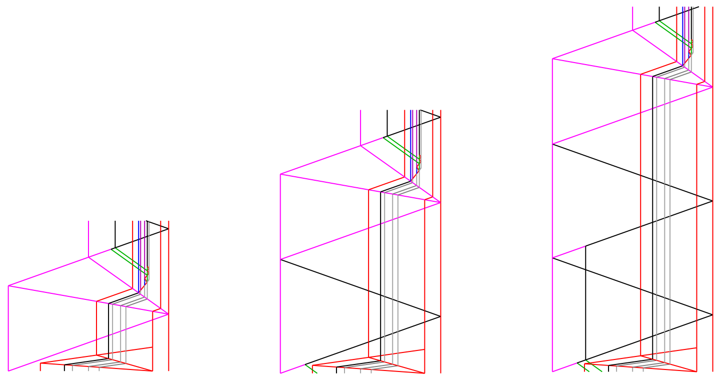
If y_n is zero, the outer structure just shrinks as in Fig. 5(a). Figure 5(b) illustrates a unit delay: the bottom signal that crosses the configuration left to right encounters the unique 1 output. It collects it and goes forth and back to the left to start again. If the delay is more important, like in Fig. 5(c), the other 1's are stored (vertical black line) and each time one is collected and processed.

The inner structure does not produce any accumulation since it freezes each time it output anything (including for the empty word) which it does infinitely often.

Since the configuration is shrunk by one half each time, the unit delay sequence is geometrical with one half factor. The sum of delays is z_n.

The outer shrinking process alone provides a term to the final accumulation time. This term, g, is the sum of a geometrical sequence of factor one half, which is rational. It is easy to scale down the initial configuration so that g is less than t. Since *c.e.* are stable by rational addition, $t - g$ is *c.e.* and the control should output the sequence corresponding to it and α (as given by the scale of the outer structure).

Figure 7(a) illustrates a longer run with a machine that always outputs 11.



(a) delay 0 (b) delay 1 (c) delay 2

Fig. 5. Shrinking step and delays.

7 Accumulating at a *d-c.e.* spatial position

The whole structure shifts left or right according to the inner control. Figure 6(c) illustrates the case where nothing is output (it is similar to the Fig. 5(a)). For each $\overline{1}$ output the whole configuration is shifted by its width on the left. This is done as on Fig. 6(a): all signals are drifting on the left with identical speed (*i.e.*, they are parallel). Two fast parallel signals ensure that the distance is the same; they form a parallelogram.

For the right shift in Fig. 6(b), the parallelogram is incomplete, but the second diagonal is here. As before if the movement is of more that one unit, all other units are preserved and treated one after the other as depicted in Fig. 6(d).

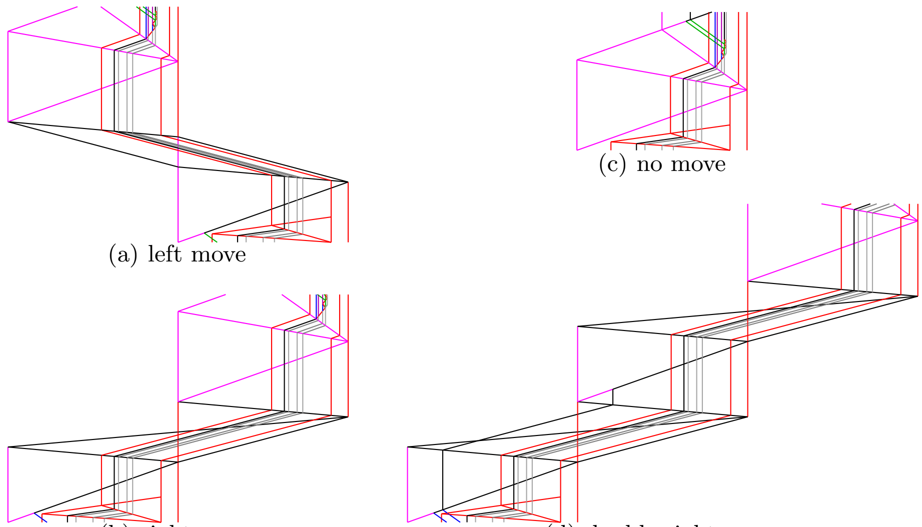
A larger run is displayed in Fig. 7(b).

To accumulate on a given *d-c.e.* number, the right extremity of the structure should be set at coordinate 0; then α can be chosen to be 1 by scaling.

The whole structure accumulates because the total time is: the outer structure time —which is finite— plus the time for the shifts. The unitary shifts, whether on left or right, have the same duration. This duration is proportional to the shift so we have to ensure that $\sum_n |y_n| \, 2^{-n}$ converges.

$$
\begin{aligned}
|y_{n+1}| &\leq \left| \frac{2^{n+1}}{\alpha} (x_{n+1} - x_n + e_n) \right| + 1 && \text{(from (1))} \\
\frac{\alpha}{2^{n+1}} |y_{n+1}| &\leq |x_{n+1} - x_n| + |e_n| + \frac{\alpha}{2^{n+1}} \\
\frac{\alpha}{2^{n+1}} |y_{n+1}| &\leq |x_{n+1} - x_n| + 3 \frac{\alpha}{2^{n+1}} && \text{(from (3))} \\
\sum_{1 \leq n} \frac{\alpha}{2^{n+1}} |y_{n+1}| &\leq \sum_{1 \leq n} |x_{n+1} - x_n| + 3 \sum_{1 \leq n} \frac{\alpha}{2^{n+1}}
\end{aligned}
$$

From Th. 1 the first sum converges.



(a) left move (b) right move (c) no move (d) double right move

Fig. 6. Shrinking step and shifts.

8 Conclusion

By considering a universal Turing machine in the control and union of signal machines (one for space and one for time), comes:

Theorem 2. *There is a rational signal machine that can generate isolated accumulation at any* c.e. *time or* d-c.e. *spatial position depending on the initial configuration.*

Following the restriction on the coordinated expressed in Lem. 2, we conjecture that there can be an isolated accumulation at any such coordinates, *i.e.*, time and spatial position simultaneously.

When spatial dimension 2 and above is addressed, it seems that each spatial coordinate can be traded independently and the same result holds with a similar relation with time.

Acknowledgement

The author would like to thank Laurent Bienvenu for inspiring discussions on *d-c.e.* numbers.

Bibliography

A. Adamatzky, editor. *Collision based computing.* Springer London, 2002.

A. Adamatzky and J. Durand-Lose. Collision computing. In D. Corne, editor, *Handbook of Natural Computing: Theory, Experiments, and Applications, Part II.* Springer, 2010. Book in press.

K. Ambos-Spies, K. Weihrauch, and X. Zheng. Weakly computable real numbers. *J. Complexity*, 16(4):676–690, 2000. doi: 10.1006/jcom.2000.0561.

H. Andréka, I. Németi, and P. Németi. General relativistic hypercomputing and foundation of mathematics. *Nat. Comput.*, 8(3):499–516, 2009.

L. Blum, M. Shub, and S. Smale. 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.

D. Duchier, J. Durand-Lose, and M. Senot. Fractal parallelism: Solving SAT in bounded space and time. In C. Otfried, K.-Y. Chwa, and K. Park, editors, *Int. Symp. on Algorithms and Computation (ISAAC '10)*, number 6506 in LNCS, pages 279–290. Springer, 2010. doi: 10.1007/978-3-642-17517-6_26.

J. Durand-Lose. Abstract geometrical computation 1: Embedding black hole computations with rational numbers. *Fund. Inf.*, 74(4):491–510, 2006a.

J. Durand-Lose. Forcasting black holes in abstract geometrical computation is highly unpredictable. In J.-Y. Cai, S. B. Cooper, and A. Li, editors, *Theory and Applications of Models of Computations (TAMC '06)*, number 3959 in LNCS, pages 644–653. Springer, 2006b. doi: 10.1007/11750321_61.

J. Durand-Lose. Abstract geometrical computation and the linear Blum, Shub and Smale model. In S. B. Cooper, B. Löwe, and A. Sorbi, editors, *Computation and Logic in the Real World, 3rd Conf. Computability in Europe (CiE '07)*, number 4497 in LNCS, pages 238–247. Springer, 2007. doi: 10.1007/978-3-540-73001-9_25.

J. Durand-Lose. Abstract geometrical computation with accumulations: Beyond the Blum, Shub and Smale model. In A. Beckmann, C. Dimitracopoulos, and B. Löwe, editors, *Logic and Theory of Algorithms, 4th Conf. Computability in Europe (CiE '08) (abstracts and extended abstracts of unpublished papers)*, pages 107–116. University of Athens, 2008a.

J. Durand-Lose. The signal point of view: from cellular automata to signal machines. In B. Durand, editor, *Journées Automates cellulaires (JAC '08)*, pages 238–249, 2008b.

J. Durand-Lose. Abstract geometrical computation 3: Black holes for classical and analog computing. *Nat. Comput.*, 8(3):455–472, 2009a. doi: 10.1007/s11047-009-9117-0.

J. Durand-Lose. Abstract geometrical computation and computable analysis. In J. F. Costa and N. Dershowitz, editors, *Int. Conf. on Unconventional Computation 2009 (UC '09)*, number 5715 in LNCS, pages 158–167. Springer, 2009b. doi: 10.1007/978-3-642-03745-0_20.

J. Durand-Lose. The coordinates of isolated accumulations [includes] computable real numbers. In F. Ferreira, H. Guerra, E. Mayordomo, and J. Rasga, editors, *Programs, Proofs, Processes, 6th Int. Conf. Computability in Europe (CiE '10) (abstracts and extended abstracts of unpublished papers)*, pages 158–167. CMATI, U. Azores, 2010a.

J. Durand-Lose. Abstract geometrical computation 5: embedding computable analysis. *Nat. Comput.*, 2010b. doi: 10.1007/s11047-010-9229-6. Special issue on Unconv. Comp. '09.

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

M. Hagiya. Discrete state transition systems on continuous space-time: A theoretical model for amorphous computing. In C. Calude, M. J. Dinneen, G. Paun, M. J. Pérez-Jiménez, and G. Rozenberg, editors, *Unconventional Computation, 4th Int. Conf., UC '05, Sevilla, Spain, October 3-7, 2005, Proceedings*, volume 3699 of *LNCS*, pages 117–129. Springer, 2005.

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

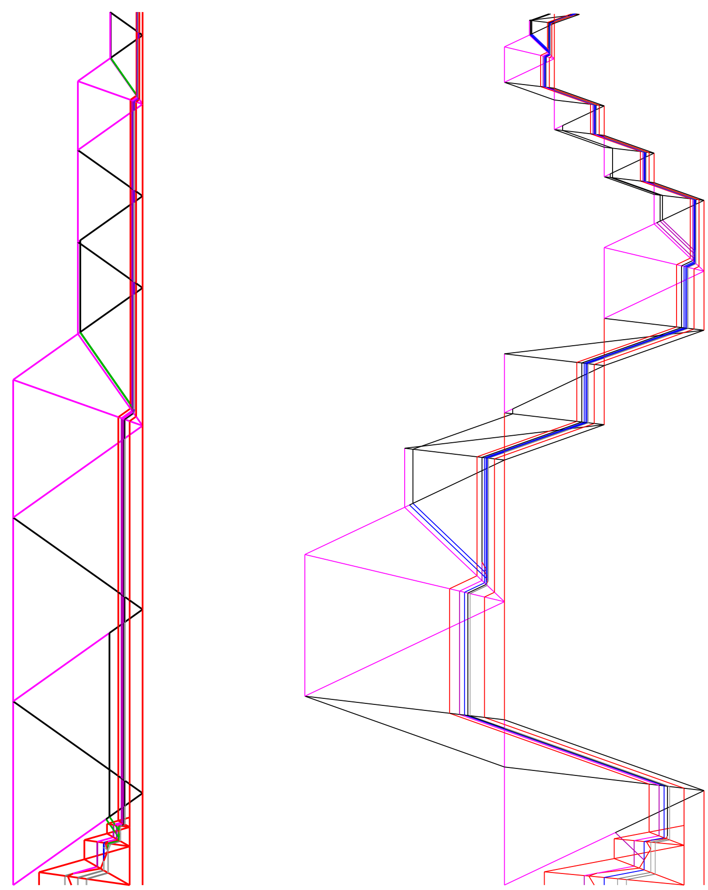
G. Jacopini and G. Sontacchi. Reversible parallel computation: an evolving space-model. *Theoret. Comp. Sci.*, 73(1):1–46, 1990.

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

I. Takeuti. Transition systems over continuous time-space. *Electr. Notes Theor. Comput. Sci.*, 120:173–186, 2005.

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

X. Zheng. A computability theory of real numbers. In A. Beckmann, U. Berger, B. Löwe, and J. V. Tucker, editors, *Logical Approaches to Computational Barriers, Second Int. Conf. on Computability in Europe, CiE '06, Swansea, UK, June 30-July 5, 2006, Proceedings*, volume 3988 of *LNCS*, pages 584–594. Springer, 2006. ISBN 3-540-35466-2. doi: 10.1007/11780342_60.



(a) $\mathsf{11.11.11}\ldots$ delays (b) $\bar{\mathsf{1}}.\mathsf{11}.\overline{\mathsf{111}}.\mathsf{1111}\ldots$ left and right moves

Fig. 7. Longer runs.