Theoretical Computer Science 412 (2011) 3616–3628

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Communication complexity in number-conserving and monotone cellular automata

E. Goles [a,b], A. Moreira [c,b,*], I. Rapaport [d]

[a] Universidad Adolfo Ibáñez, Av. Diagonal Las Torres 2640, Peñalolén, Santiago, Chile

[b] Institute for Complex Systems (ISCV), Subida Artillería 470, Valparaíso, Chile

[c] Departamento de Informática and Centro Tecnológico de Valparaíso (CCTVal), Universidad Técnica Federico Santa María, Casilla 110-V, Valparaíso, Chile

[d] DIM, CMM (UMI 2807 CNRS), Universidad de Chile, Av. Blanco Encalada 2120, Santiago, Chile

ARTICLE INFO

Article history:
Received 20 October 2010
Received in revised form 3 March 2011
Accepted 14 March 2011
Communicated by B. Durand

Keywords:
Cellular automata
Communication complexity
Elementary cellular automata
Number-conserving

ABSTRACT

One third of the elementary cellular automata (CAs) are either number-conserving (NCCAs) or monotone (increasing or decreasing). In this paper we prove that, for all of them, we can find linear or constant communication protocols for the prediction problem. In other words, we are able to give a succinct description for their dynamics. This is not necessarily true for general NCCAs. In fact, we also show how to explicitly construct, from any CA, a new NCCA which preserves the original communication complexity.

1. Introduction

Cellular automata are extremely complex (highly nonlinear) objects and therefore the language of computer science appears to be particularly suitable for studying them. In that spirit, our approach consists in splitting the cells into (two) groups in order to describe the CA's dynamics by finding simple communication protocols taking place between these parts [1,2]. Our motivation is the following: if we were capable of giving a simple algorithmic description of a CA, then we would have understood its behavior.

In the present paper we apply this communication approach to study all the number-conserving and monotone elementary CAs (which comprise about 1/3 of the elementary CAs). In these well-studied CAs the state of each cell can be interpreted as the number of particles in it [3–7]. The sum of these numbers can either stay constant (NCCAs), increase (ICAs) or decrease (DCAs) during the evolution. NCCAs have been studied in several contexts (in particular for traffic models).

We prove here that the prediction problem, for all of these CAs, can be solved by constant or linear communication protocols. Recall that a linear (and, more so, a constant) protocol for a CA corresponds to a succinct way to describe or represent its dynamics.

It is important to point out that these previous positive results are no longer true for NCCAs with more states. In fact, we already know the existence of intrinsically universal NCCAs [8]. Therefore, from [9], we can conclude that the communication complexity for the prediction problem on such intrinsically universal NCCAs is maximal. In this paper we go yet further by showing how to explicitly construct, from any CA, a new NCCA which preserves the original communication complexity.

* Corresponding author at: Departamento de Informática and Centro Tecnológico de Valparaíso (CCTVal), Universidad Técnica Federico Santa María, Casilla 110-V, Valparaíso, Chile. Tel.: +33 9 97776652.

E-mail addresses: eric.chacc@uai.cl (E. Goles), andres.moreira@usm.cl (A. Moreira), rapaport@dim.uchile.cl (I. Rapaport).

0304-3975/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2011.03.008

2. Definitions and notation

Cellular automata. A one-dimensional cellular automaton (CA) is a discrete dynamical system where the configuration space is $S^{\mathbb{Z}}$, for some finite set of states S, and the dynamics $F : S^{\mathbb{Z}} \to S^{\mathbb{Z}}$ is given by the synchronous application of a local function $f : S^{\ell+1+r} \to S$, $\ell, r \geq 0$, so that $F(x)_i = f(x_{i-\ell}, \ldots, x_i, \ldots, x_{i+r})$. Without loss of generality, we will assume that $\ell = r$ (the "radius" of the CA) and that $S \subset \mathbb{Z}$. The family of *elementary* cellular automata (ECAs) consists of the CAs with radius 1 and $S = \{0, 1\}$; we number them according to the standard Wolfram coding: $code(f) = \sum_{abc} f(abc)2^{4a+2b+c}$.

Notice that, after n time steps, the state of a cell will depend on the initial values of $1 + 2rn$ cells: itself, and its nr nearest neighbors to both left and right. More precisely, we define the nth iteration of f as the function $f^n : S^{2rn+1} \to S$ given by

$$f^n(x_{-rn}, \ldots, x_{-1}, x_0, x_1, \ldots, x_{rn}) = f^{n-1}(f(x_{-rn}, \ldots, x_{-rn+2r}), \ldots, f(x_{rn-2r}, \ldots, x_{rn}))$$

for $n \geq 2$ and $f^1 = f$.

Number-conserving CAs. Let S^* be the set of all finite words on S, $L(w)$ be the length of a given word, and for $w \in S^*$, $w = w_1, \ldots, w_{L(w)}$ define $\overline{w} \in S^{\mathbb{Z}}$ as $\overline{w}_i = w_{i \bmod L(w)}$. A *number-conserving* CA (NCCA) is a CA F such that

$$\sum_{i=1}^{L(w)} \overline{w}_i = \sum_{i=1}^{L(w)} F(\overline{w})_i \qquad \forall w \in S^* \tag{1}$$

Note that S can always be chosen so that $\min S = 0$ (if not, then we subtract $\min S$ from each state). Condition (1) is equivalent to the preservation, under F, of the total sum of any configuration with a finite number of non-zero states. Thus, a natural way to look at the situation is in terms of "particles" that can be neither created nor destroyed, and whose distribution along $\mathbb{Z}$ is given by the values in the cells [3,4]. It turns out that, for (one-dimensional) NCCAs, there is always an alternative description of the dynamics from the particles' point of view: there is a local function which describes the movement of the particles of a cell, and whose parallel application on the whole configuration yields the same image as the CA [5,6]. If in addition we ask that the movement preserves the particles' order (*i.e.*, they never jump over each other) then this alternative description is unique.

Motion representation. An intuitive and succinct way to write these "motion rules", following Boccara and Fukś [3], is through a list of local configurations, each of which indicates the movement it induces on the particle(s) at the origin; configurations with no movement are not shown, and only the number of particles that leaves the origin is given in each case. If more than one configuration applies to a situation, the first one listed is applied. Let us consider, for instance, the four following rules:

$$M_1 = \{\overset{\curvearrowright}{10}\}, \qquad M_2 = \{\overset{\curvearrowright}{10}, 0\overset{\curvearrowleft}{01}1\}, \qquad M_3 = \{\overset{2}{\overset{\curvearrowright}{20}}, \overset{1}{\overset{\curvearrowright}{21}}\}, \qquad M_4 = \{\overset{1}{\overset{\curvearrowright}{2\bullet}}\}$$

Here $S = \{0, 1\}$ for M_1 and M_2, while $S = \{0, 1, 2\}$ for M_3 and M_4. M_1 is read as follows: a particle moves to the right if and only if that site is empty. In the case of M_2, the particle will still move to the right if it can, but if not, then it will move one position to the left if it sees two empty positions in that direction. In any other case, it keeps its current position. Numbers may be added to the arrows when a cell may be occupied by more than one particle. This is the case in M_3: if two particles are in a cell, then one, two, or none of them may move to the right, depending on the space available there. In M_4, the bullet ($\bullet$) is a wildcard: if there are 2 particles in a cell, then one of them moves to the right, regardless of what is there.

Monotone CAs. If we modify Eq. (1) by allowing inequalities (more precisely, $\sum \overline{w}_i \geq \sum F(\overline{w})_i$), then we obtain the family of DCAs, *decreasing cellular automata*, where particles may vanish; similarly, by putting a "$\leq$" sign, we may define the family of ICAs, *increasing CAs*. Of course, NCCAs correspond to the intersection of DCAs and ICAs; furthermore, any ICA can be turned into a DCA (and vice versa) by replacing S with $-S$ (or any affine map of the same sign). The notation for motion rules can be easily extended to DCAs by adding a hat sign on particles that vanish; thus in

$$M_5 = \{\overset{\curvearrowleft}{01}, \overset{\curvearrowleft}{11}, 2\hat{1}\}$$

the 1's will travel to the left until they meet a 2, and then will disappear. For a given one-dimensional DCA there is always such a motion representation [7]; however, unlike the conservative case, there is no "canonical" motion rule for DCA.

Communication complexity. The concept of *communication complexity* was introduced by Yao [10] as a tool for understanding the needs of information exchange in parallel computation, by providing lower bounds for it. In Yao's model there is a function $F : X \times Y \to \mathbb{Z}$, where X and Y are finite sets. Two parties, Alice and Bob, must compute $F(x, y)$, but x is given only to Alice, and y to Bob. Knowing F in advance, they will apply a previously agreed protocol so that the number of exchanged bits is minimal in the worst case. This worst case cost (in bits) is the *many-round communication complexity*, $cc(F)$. A related notion is the *one-round communication complexity*, $cc_1(F)$, where the protocol is only allowed to send information in one direction: from Alice to Bob, or from Bob to Alice; the worst of both directions is considered.

This notion has been applied to the analysis of (one-dimensional) CAs [1,11,2,12,9]. It provides a new way to look at CAs complexity, and furthermore it has been shown [9] that *intrinsic universality* implies maximal communication complexity for a number of problems; this turns their analysis into a tool for possibly proving the (otherwise elusive) non-universality

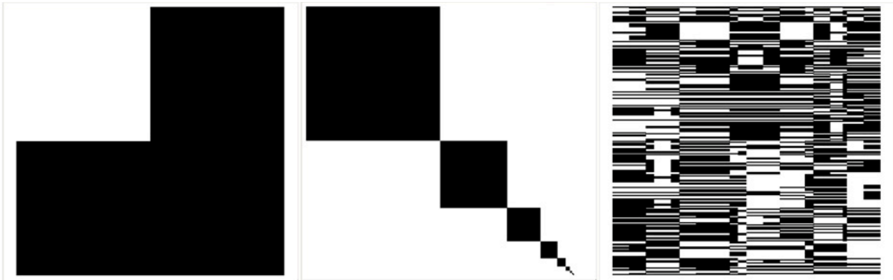

Fig. 1. Matrices for three different elementary rules: from left to right, $M^{1,8}_{200}$, $M^{1,8}_{132}$, $M^{1,8}_{110}$. They exhibit constant, linear and (conjectured) exponential communication complexity, respectively. Zeros are shown as white, ones as black.

of some CAs. Several problems have been studied, but the simplest and most natural is PRED (prediction): $F = f^n$ (where f is the local rule of the CA), and X and Y stand for the left and the right side of the initial configuration (the center is shared by both). The growth of

$$cc_1(f, n) = \max_{c \in S} cc_1(f^n(\cdot, c, \cdot))$$

is then studied as $n \to \infty$. Useful in this context, for computation as well as visualization, are the sequences of matrices defined for each $c \in S$ as

$$M_f^{c,n}(x, y) = f^n(x, c, y) \quad \text{with } x, y \in S^{rn}$$

It is not difficult to see that cc_1 is related to these matrices, or rather, to their maximum numbers of different rows and columns: if there are N different rows, then Alice must tell Bob which of these N possibilities occurs on her side: otherwise, there will be some case on Bob's side for which he will not be able to compute the answer. In order to distinguish between the N cases, Alice needs to send $\lceil \log_2 N \rceil$ bits, and Bob will have to do the same for the number of different columns. In general, if we denote by $d(M)$ the maximum between the number of different rows and the number of different columns in M, what we have (following [1] who applied [13]) is

$$cc_1(f, n) = \max_{c \in S} \lceil \log_2 d(M_f^{c,n}) \rceil$$

Abusing the language, we will talk of rules with *linear* (or constant, quadratic, exponential, etc.) communication complexity to refer to the way in which the number of rows/columns grows (hence, to the growth of 2^{cc_1}, rather than cc_1). Some intuition of the kind of behavior can be obtained from the graphical representation of the matrices (see Fig. 1).

3. Transference of communication complexity results from CAs to NCCAs

Theorem 1. *Let F be a (one-dimensional) CA. Then F is directly simulated by a (one-dimensional) NCCA $\widetilde{F}$ such that $|cc_1(\widetilde{F}, n) - cc_1(F, n)| \leq 2$.*

Proof. Let r be the radius of F, f its local function, and $S = \{1, \ldots, q\}$ its set of states (we assign the numbers arbitrarily). We define $\widetilde{F}$ as the CA with state set $\widetilde{S} = -S \cup \{0\} \cup S$, radius $\widetilde{r} = 2r + 1$, and $\widetilde{f}(x_{-2r-1}, \ldots, x_{2r+1})$ defined with the following values:

- x_0 if $0 < x_0 \neq -x_1$ or $0 > x_0 \neq -x_{-1}$.
- $f(x_{-2r}, x_{-2r+2}, \ldots, x_{2r})$ if $x_{2i} = -x_{2i+1} > 0$ for all $-r \leq i \leq r$.
- $f(-x_{-2r}, -x_{-2r+2}, \ldots, -x_{2r})$ if $x_{2i} = -x_{2i-1} < 0$ for all $-r \leq i \leq r$.
- 0 otherwise.

The construction is almost the same as in the proof of Theorem 4 in [8]; the only difference is the addition of a propagating 0. In that theorem the purpose was to show intrinsic universality, by simulating any CA in real time through an NCCA; the same simulation is done here. In order to evaluate f using $\widetilde{f}$ we "split" each cell into two, and copy its value in both (with different signs). Then we apply $\widetilde{F}$, which will do the following: if a cell and its neighbor are part of a valid (alternating) configuration, then both change according to F (and the sum remains 0). If they see a discrepancy, and they are a valid pair (of the form a, $-a$), then they change to 0 (and again, the sum remains 0). If they are not a valid pair, then they keep their values. Notice that, since 0 is not a part of the valid configurations, it will propagate, and $\widetilde{f}^n$ will evaluate to 0 if the initial configuration is not a valid one but the origin is part of a valid pair (see Fig. 2); if the origin is not part of a valid pair, it will remain constant through the entire iteration.

Suppose now that we have an optimal protocol for F which requires $cc_1(F, n)$ bits. Then we construct a protocol for $\widetilde{F}$ as follows (we will assume that Alice talks to Bob):

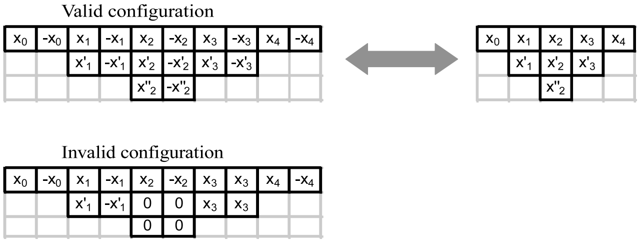

Fig. 2. Behavior of the NCCA: invalid pairs of cells remain constant, valid pairs in the presence of valid (local) configurations iterate according to the original rule, valid pairs in the presence of invalid configurations become 0.

- If the center c is negative and the value of the cell to its left is not $-c$, then Alice sends a 1, signaling to Bob that the center is invalid and will remain constant. Otherwise she sends a 0.
- Now Alice sends a bit informing whether the rest of her input is a valid configuration. If it is invalid, Bob will know the final value, which will be either 0 or c.
- If the two previous bits were 0, now Alice projects here side into a configuration for F, and sends whatever message F's protocol indicates.

This completes a protocol for $\widetilde{F}$, and hence $cc_1(\widetilde{F}) \leq cc_1(F) + 2$. On the other hand, suppose that we have an optimal protocol for $\widetilde{F}$. Then we can use it for solving the problem for F: given an input for F, we construct an input for $\widetilde{F}$ by the duplication described above, apply the protocol for $\widetilde{F}$, and project the answer (which will be different from 0, since it arises from a valid configuration) back into S. Hence, $cc_1(F) \leq cc_1(\widetilde{F})$. □

Corollary 2. *There exist NCCAs for which the communication complexity grows as n^k for any k; there also exist NCCAs for which the communication complexity grows exponentially in n.*

Proof. This follows from the previous theorem and from the examples of CA outlined in [1] for these different speeds of growth. □

Remark. Notice that the argument in this proof also applies if we consider *many-round* protocols; thus, cc can also be transferred from any one-dimensional CA to a NCCA.

4. Communication complexity for elementary monotone and conservative rules

Among the family of ECAs ($|S| = 2, r = 1$) there are 87 rules with monotone behavior: 5 of these are NCCAs, while the rest splits evenly between DCAs and ICAs. Each ICA can be seen as a conjugated version of a DCA (by replacing $0 \leftrightarrow 1$); moreover, many rules are obtained from others through horizontal reflexion. Table 1 lists all the rules, with DCAs on the left followed by the corresponding ICA; NCCA are listed at the end. The results of this section are summarized by the last column, which indicates the kind of communication complexity found, and will be proved in each case for the rule mentioned in the first column (*i.e.*, for the group formed by rules 40, 96, 235 and 249, we will prove the rule 40).

4.1. Some useful observations

There are several general facts that can simplify the study of the growth of $cc_1(f, n)$ for $n \to \infty$; all of them rely on the possibility of communicating constant amounts of data, or of considering a constant number of possible answers.

Convention on coordinates: First, let us state a convention which will be used in the rest of this section: whenever necessary, we will use for reference a coordinate system centered at the middle of the initial configuration, so that (p, t) denotes position p at time $t \geq 0$; the value present at that position and time will be denoted with v_p^t, for $|p| \leq (n-t)r$ (otherwise the value is in general unknown to us). The initial condition is thus $(v_{-nr}^0, \ldots, v_{nr}^0)$, and for each time t we can write $v^t = f^t(v^0)$. Alice and Bob know $(v_{-nr}^0, \ldots, v_0^0)$ and $(v_0^0, \ldots, v_{nr}^0)$, respectively, and their goal is to compute v_0^n.

Lemma 3. *If there is a constant protocol for the communication in one direction, then both directions have constant protocols.*

Proof. Let S be the set of states, and suppose that Alice has a constant protocol of M bits that allows Bob to compute f^n. Then there is a protocol for Bob talking to Alice, in which Bob sends the values of f^n that he would compute for each of the 2^M messages he may get from Alice. Thus, Bob's message specifies a function from $2^M \to S$, which requires $2^M \times \log_2 |S|$ bits (a constant amount). □

Lemma 4. *For a fixed value k, it may be assumed that Alice and Bob receive as input the values of $(v_{-(n-k)r}^k, \ldots, v_0^k)$ and $(v_0^k, \ldots, v_{(n-k)r}^k)$, respectively. In other words, we may assume that Alice and Bob are dealing with a configuration that has already been iterated k times.*

Table 1
List of elementary NCCAs, DCAs, ICAs and their communication complexity for PRED.

DCA		ICA		
Rule	Reflected	Conjugate	Reflected conj.	Growth
0	–	255	–	Constant
2	16	191	247	Constant
4	–	223	–	Constant
8	64	239	253	Constant
10	80	175	245	Constant
12	68	207	221	Constant
24	66	231	189	Constant
32	–	251	–	Constant
34	48	187	243	Constant
40	96	235	249	Linear
42	112	171	241	Constant
56	98	227	185	Linear
72	–	237	–	Constant
76	–	205	–	Constant
128	–	254	–	Constant
130	144	190	246	Constant
132	–	222	–	Linear
136	192	238	252	Constant
138	208	174	244	Constant
140	196	206	220	Constant
152	194	230	188	Linear
160	–	250	–	Constant
162	176	186	242	Constant
168	224	234	248	Linear
200	–	236	–	Constant
170	240	170	240	Constant } NCCA
184	226	226	184	Linear } NCCA
204	–	204	–	Constant } NCCA

Proof. We may in fact assume something slightly more general: that the knowledge shared by Alice and Bob is not just v_0^0, but rather $v_{-kr}^0, \ldots, v_0^0, v_1^0, \ldots, v_{kr}^0$ (with this they can compute the kth iteration of their sides).

Consider the case of Alice talking to Bob, and let $\phi(v_{-(n-k)r}^k, \ldots, v_0^k) = \phi(f^k(v_{-nr}^0, \ldots, v_0^0, v_1^0, \ldots, v_{kr}^0))$ be the message she would send if she knew the additional information. Since she does not really know it, she sends the entire set $\{\phi(f^k(v_{-nr}^0, \ldots, v_0^0, w)) : w \in S^{kr}\}$; Bob will know which element of the set applies. Furthermore, Alice must send the values of $v_{-kr}^0, \ldots, v_1^0$. The cost of the protocol is thus an affine function of the cost of ϕ (and has the same growth). □

Lemma 5. *Let F be the global action of a CA with radius r and let j and k be integers such that*

$$F(F^k(c)) = \sigma^j(F^k(c)) \quad \forall c \in S^{\mathbb{Z}}$$

where σ is the left shift ($(\sigma(c))_i = c_{i+1}$). Then F has constant communication complexity.

Proof. From Lemma 4, we can assume that the CA is already σ^j for some j. If $j > 0$, then Bob's protocol is to say the result, and Alice's protocol is empty; for $j < 0$ their roles are reversed. □

4.2. Constant cases

Proposition 6 (*Proved in [1,11]*). *The computational complexity of the prediction problem is constant for ECAs* 0, 2, 4, 8, 10, 12, 16, 24, 32, 34, 42, 48, 64, 66, 68, 72, 76, 80, 112, 128, 130, 136, 138, 140, 144, 160, 162, 170, 171, 174, 175, 176, 186, 187, 189, 190, 191, 192, 196, 200, 204, 205, 206, 207, 208, 220, 221, 223, 231, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 250, 251, 252, 253, 254 *and* 255.

Proof. These rules are quite simple, for the most part, and have already being correctly classified elsewhere. We will give here the precise reason for each case, since it was not always explicitly stated in the references, and the space required is short. We discuss only one rule for each row of Table 1 labeled as Constant, and the rest are obtained through reflection or through $0 \leftrightarrow 1$ substitution.

- Lemma 5 solves the cases of rules 0, 4, 12, 76 and 200 (for which $k = 2, j = 0$ would work), 8 and 72 ($k = 3, j = 0$), 2, 10, 34, 42 and 138 ($k = 2, j = 1$) and 66 ($k = 3, j = 1$). Rules 170 (σ itself) and 204 (the identity) fall trivially into this case too.

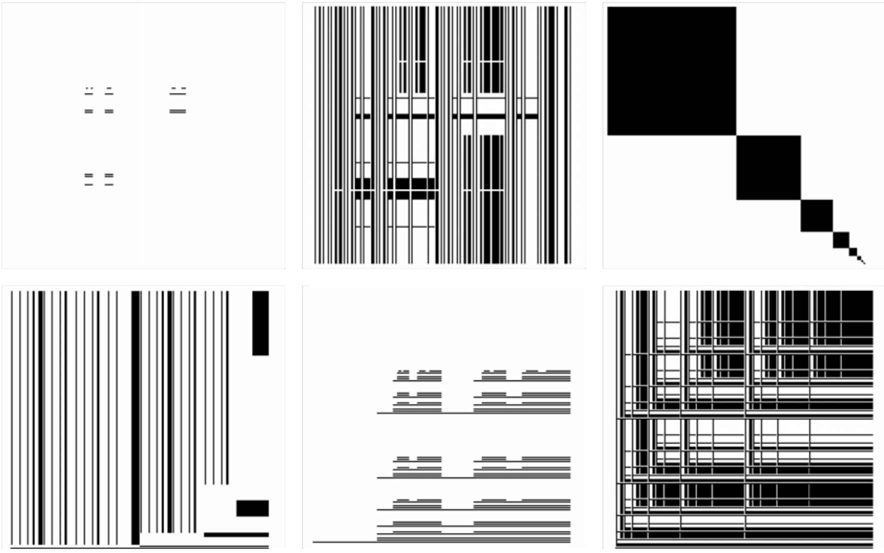

Fig. 3. PRED matrices for the linear cases in number-conserving and monotone ECA; in all cases $n = 8$ and $v_0^0 = 1$. Top: rules 40, 56 and 132. Down: rules 152, 168 and 184.

- Rules 32, 128, 160 and 136: For these rules $v_0^n = 1$ iff the initial condition follows a specific pattern; Alice and Bob only need to say whether they have their side of it. The pattern is 1010...101 for rule 32, 111...111 for rule 128, and 1?1?1...1?1 for rule 160 (where "?" represents any value). For rule 136, the center and the whole right side must be 1.
- Rule 140: Its motion representation is $M_{140} = \{1\hat{1}0\}$; thus, 0 never changes to 1, and 1 only changes to 0 through the local rule $f(110) = 0$. It is easy to see that groups of 1's are erased by 0's on their right, except for the leftmost 1 in the group, which survives. The predicted value will be 1 iff the whole right side (including the origin) are 1's (the group would not be erased) or if there is a 1 at the origin with a 0 next to it on the left (if the group is erased, the particle at the origin will be the survivor).
- Rules 130 and 162: their motion representations are

  $$M_{130} = \{01\hat{1},\ 10\hat{1},\ \overset{\frown}{\bullet 1}\} \qquad M_{162} = \{01\hat{1},\ \overset{\frown}{\bullet 1}\}$$

  Hence, they are shifts to the left except for two and one cases where a 1 may vanish. For both, if Bob has a 0 on his side, then he can safely ignore Alice's side and iterate (anything to the left of that 0 will just shift away or disappear). Thus, if Bob talks to Alice, he just gives the result, or informs that he has "1 . . . 1" (in which case Alice computes the result). For the other direction we apply Lemma 3. □

4.3. Linear cases

The remaining DCAs have linear communication complexity for PRED. Some insight on their relative complexities can be found in their matrices (see Fig. 3); for further illustration, Fig. 4 gives some sample iterations for three of the linear rules. The most complicated case turned out to be the only NCCA, 184, as could be guessed from its matrix; we will deal with it first.

Rule 184 is usually considered the most basic model for "highway traffic". The reason for this is its motion representation, where particles move one position to the right, if and only if that spot is empty:

$$M_{184} = \{\overset{\frown}{10}\}$$

Theorem 7. *The computational complexity of the prediction problem is linear for rules* 184 *and* 226.

Proof. We will work with rule 184 (226 is its reflexion). Consider first the case where $v_0^0 = 0$, and the message is sent from Bob to Alice. From the motion representation it is easy to see the following.

- Bob knows that the movement of all the particles originally located on his side of the starting configuration (he may lose track of them when they exit the space–time cone determined by the initial condition, but in that case their whereabouts are no longer relevant).
- Since the particles keep their order, and they are only affected by the position of the particle ahead of them, only the trajectory $\{p_t : 0 \le t \le n\}$ of the leftmost of Bob's particles ("β") is relevant to the movement of Alice's particles (see Fig. 5).

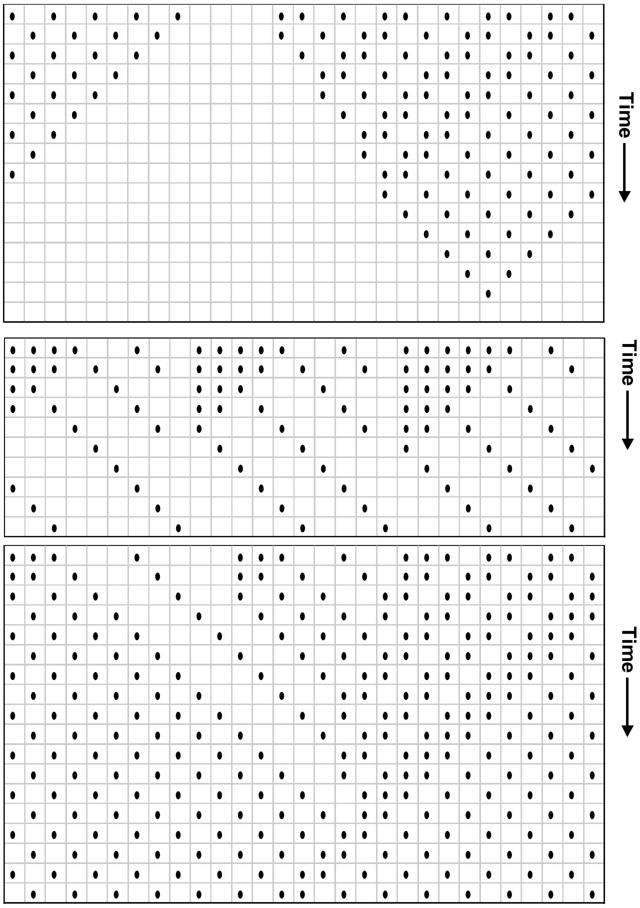

Fig. 4. Sample dynamics for rules 40 (top), 152 (center) and 184 (bottom), with periodic boundary condition.

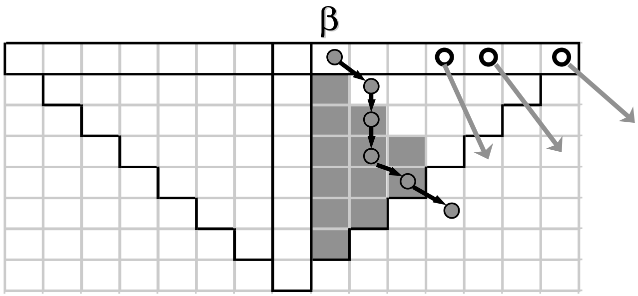

Fig. 5. Only the leftmost particle of Bob (β) is seen by Alice's particles. Moreover, only its movement within the shaded zone is relevant.

- Only the part of β's trajectory for which $p_t \leq \min\{t, n-t\}$ is relevant: if $p_t > t$, even the rightmost particle of Alice, traveling at maximum speed, will never see $v^t_{p_t}$; on the other hand, if $p_t > n-t$, then β is no longer in the cone that influences the final state.
- It would thus suffice if Bob gives to Alice the trajectory of β within the area shaded in Fig. 5.

We claim that only the final position of β within this area is relevant for the result: Alice needs to know only which of the positions (if any) marked with a dot in Fig. 6 was the last visible (p_t, t) (i.e., the point in β's trajectory which maximizes t, while having $p_t \leq \min\{t, n-t\}$). These possible last positions are the pairs $(k, n-k)$ for $k = 1, \ldots, \lfloor \frac{n}{2} \rfloor$, and $(k, n-k-1)$ for $k = 1, \ldots, \lfloor \frac{n-1}{2} \rfloor$ (see Fig. 6); thus, there are always $n-1$ possibilities, plus the case where β is not present at all.

To see why this is true, consider two slightly different trajectories for β, p and p', whose last visible position is (p_s, s). Suppose that p and p' only differ at a time $t^* < s$, for which $p'_{t^*} = p_{t^*} + 1$ (see Fig. 7). For p' to be a valid trajectory, p has to verify $p_{t^*} = p_{t^*-1} = p_{t^*+1} - 1$: in p' the particle β will move forward after time $t^* - 1$, instead of waiting until t^*. What are the effects of this change? There are two possibilities:

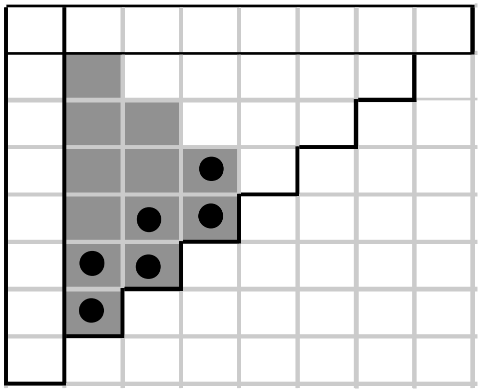

Fig. 6. The possible final positions of Bob's last particle within the shaded area are marked with bullets.

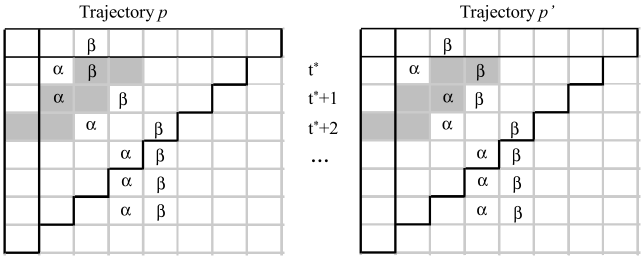

Fig. 7. Propagation (in gray) of a change in the trajectory of Bob's leftmost particle (β).

- $v_{p_{t^*}-1}^{t^*} = 0$. The change goes unnoticed, and the space–time diagram for $t > t^*$ remains the same.
- $v_{p_{t^*}-1}^{t^*} = 1$. The foremost particle of Alice ("α") notices the change. The positions of α at times t^*, t^*+1 and t^*+2 were $p_{t^*}-1$, $p_{t^*}-1$ and p_{t^*}, respectively, when β followed p, and will be $p_{t^*}-1$, p_{t^*} and p_{t^*} when β follows p'.

Thus, the change from β's trajectory causes an identical change in α's trajectory, but one time step later and one position to the left. The perturbation may propagate further, always shifting to the left, or may die out: at time t^*+k it will be at $p_{t^*}-k$ and $p_{t^*}-k+1$. In order for the perturbation to affect v_0^n, we would need $k = n-t^*$ and hence either $p_{t^*}-n+t^* = 0$, in which case (p_{t^*}, t^*) lies in the bottom border, contradicting $t^* < s$, or else $p_{t^*}-n+t^*-1 = 0$, in which case (p_{t^*}, t^*) is one position to the left of this bottom border. Since $t^* < s$, the only possibility for (p_s, s) is $s = t^*+1$, $p_s = p_{t^*}$, but this contradicts the condition required above for p' to be a valid trajectory.

Notice now that through the successive applications of these small transformations, any trajectory p whose last visible position is (p_s, s) can be transformed into another trajectory p' such that $p'_t = p_s$ $\forall t < s$, and $p'_t = p_t$ $\forall t \geq s$. There are therefore at most n different cases that must be distinguished in Bob's message.

If $v_0^0 = 1$ the situation is almost the same: this will be the particle β whose trajectory matters. There appear two additional cases to be considered as final positions, $(0, n-1)$ and $(0, n)$, and one case disappears (that in which β was never seen). Hence, Bob must distinguish between (at most) $n+1$ cases.

If Alice is the one sending the message to Bob, the situation is the same as before, thanks to a useful property of rule 184: it can be interpreted as 1's traveling to the right in a sea of 0's, or, equivalently, as 0's traveling to the left in a sea of 1's. Thus, the same analysis applies: Alice must send enough information to distinguish between n cases (if a 1 is at the origin) or $n+1$ cases (if a 0 is there) for the last visible position of her rightmost 0 within a triangle symmetric to the one considered before.

Lower bound. Since cc_1 is defined as a maximum, we will show the lower bound by taking the case of Alice talking to Bob, with the center position starting in state 1. The configurations on Alice's side will be of the form $0^{n-k}1^k$, with $0 \leq k \leq n/2$. For two different configurations in this family, with $k' < k''$, we consider the right side $1^{n-2k''}0^{2k''}$. As can be readily seen in Fig. 8, this will produce a final result of 1 in the case of k'', and of 0 for k'. If follows that Alice's message must distinguish between the $n/2$ cases of the form $0^{n-k}1^k$, a linear quantity. □

Theorem 8. *The computational complexity of the prediction problem is linear for ECAs* 40, 56, 96, 98, 132[1], 152, 168, 185, 188, 194, 222, 224, 227, 230, 234, 235, 248 *and* 249.

Proof. As before, we will show the result for some rules (40, 56, 132, 152 and 168) and the rest are implied through equivalences.

---

1 The proof for rule 132 (and hence also for 222) is given for the sake of completeness, since it was already proved in [1].

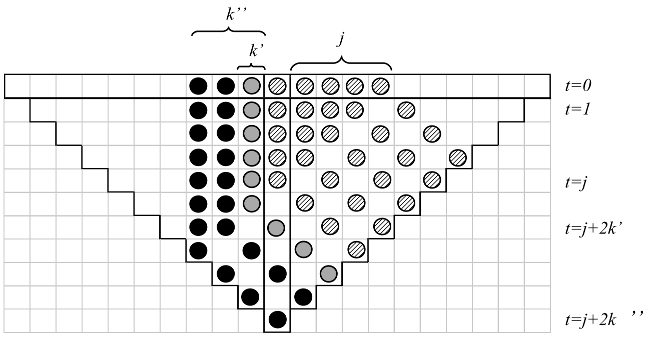

Fig. 8. Configuration for the lower bound: a single block of particles, bridging the left and right sides. By imposing $j = n - 2k''$, the final result will depend on whether $k' < k''$.

Rule 40

The motion representation may be written as

$$M_{40} = \{11\hat{1}, 00\hat{1}, \overset{\curvearrowleft}{\bullet 1}\}$$

However, the configuration 111 cannot appear except in the initial configuration, and in this case it is immediately destroyed. By Lemma 4 we may assume that it never appears, and hence the relevant motion representation is

$$M'_{40} = \{00\hat{1}, \overset{\curvearrowleft}{\bullet 1}\}$$

which is a left shift, except for the absorbing role played by blocks of two or more 0's. Moreover, because of the shift, such blocks of 0's will grow one position to the left in each iteration, and one position to the right each time they receive a new 0 (see Fig. 4, top).

Since there is never a 1 in a given cell for three consecutive time steps, we may assume (by Lemma 4) that $v_0^0 = 0$. Furthermore, some specific cases can be dealt with in a constant part of Alice's or Bob's protocols. If the first or the last cell on Bob's side are 0, then the final result is $v_0^n = 0$ (if the first is 0, then we have a 00 block; if the last is 0, there is no 1 to shift into the center at time n). Hence, Alice may give her message assuming that Bob has $v_1^0 = v_n^0 = 1$ (if not, Bob can just ignore her), and if Bob talks to Alice, he will just tell her the answer (0). Similarly, both may assume that there is no 00 block on Bob's side, since this would also imply a 0 result.

The interesting case then is when Bob has a shifting configuration, matching the regular expression $(10 + 110)^*1$. The rightmost particle will arrive at $(0, n)$ iff it is not stopped by a block of zeros, and that will happen iff there is a block of zeros on Alice's starting configuration, which gets enough additional shifting zeros (coming from either Alice's or Bob's side) to reach the center. The protocol is:

- if Alice talks to Bob, she finds the position $B \leq 0$ of the rightmost zero in the rightmost block of ≥ 2 zeros on her side, and the number of isolated zeros she sees from there to the right, Z_a. She sends the value $B + Z_a \in \{-n, \ldots, 0\}$;
- if Bob talks to Alice, he gives the number of (isolated) zeros on his side (not counting the center, which is counted by Alice), Z_b.

After the message any of them can compute the answer: $v_0^n = 1$ iff $B + Z_a + Z_b < 0$.

Lower bound. Consider n of the form $n = 6k + 1$ and configurations of the following form

$$0^{n-3i}(011)^i, 0, (110110)^j(101010)^{k-j}1 \qquad (2)$$

where j has the same parity of k, $i = \frac{3k-j}{2}$, and the commas surround v_0^0. We have $B = -3i - 1$, $Z_a = i + 1$ and $Z_b = 3k - j$, which gives 0 as the final position of the block of zeros at time n ($\implies v_0^n = 0$), just barely preventing the rightmost particle from reaching the center. If we keep the left side constant but use $j' < j$ on the right side, we increase Z_b and the result will be the same; on the other hand, if we reduce Z_b (by picking $j' > j$), $B + Z_a + Z_b$ becomes negative and $v_0^n = 1$. Thus, we have a family of right sides (for the different values of j) which are pairwise distinguishable by left side configurations; the number of members of the family grows linearly in n.

Rule 56

The motion representation for this rule is

$$M_{56} = \{\overset{\curvearrowright}{10}, 1\hat{1}1\}$$

Thus, it is similar to rule 184 discussed before, except that blocked particles which are being "pushed" from behind will vanish instead of staying put. However, the configuration 111 can only appear in the starting condition, and then it disappears (there is no preimage for 111). Alice and Bob may, by Lemma 4, assume that it never occurs. The dynamics is therefore exactly that of rule 184 and the same protocol applies.

Lower bound. Notice that

- Rule 56 is identical to rule 40, except for $f(100)$.
- 100 does not appear in configuration (2), nor in its evolution.

It follows that the lower bound found for rule 40 applies to rule 56 as well.

Rule 132

The rule is quite simple: 0 remains 0, 010 remains 010, and for larger blocks of particles we have $01^k0 \to 01^{k-2}0$. Thus, blocks will either disappear or leave a residue in their center, depending on their parity. Let k_A and k_B be the number of contiguous particles that Alice and Bob each see starting from $(0, 0)$, to the left and the right respectively; then we have $v_0^n = 1 \iff k_A = k_B > 0$. These values are thus the protocols.

Lower bound. We take the n different right side configurations of the form 1^k0^{n-k}, and $v_0^0 = 1$. They are pairwise distinguishable by configurations on the left side: if $k \neq k'$, $0^{n-k}1^k11^k0^{n-k}$ will evaluate to 1, while $0^{n-k'}1^{k'}11^k0^{n-k}$ evaluates to 0.

Rule 152

Particles separated by at least two 0's will shift to the right. Blocks of contiguous particles act as fixed walls, killing incoming particles on the left, but are not stable: at every time step they lose one particle, which alternatively vanishes or escapes to the right (see Fig. 4, center). A motion representation is

$$M_{152} = \{\overset{\frown}{1}00, \hat{1}01\}$$

The following facts about the behavior are easily verified through consideration of possible configurations:

1. Particles only move to empty sites; thus, if a cell is occupied at consecutive times, the occupying particle must be the same.
2. If a particle stays in a cell between two time steps, then it has never moved. Moreover, if the particle is fixed until time T, then at time 0 the T cells to its right were occupied.
3. The preimage of 1#1 is 1#11, where "#" indicates the same position. If two particles are contiguous, then none of them has ever moved.
4. Once a particle moves, it cannot stop: it will either keep moving, or vanish (if it hits a wall of fixed particles).
5. If the "killing" configuration 101 appears in a given iteration, then one and only one of the two particles must have been fixed from the beginning: either the left particle is fixed and the right one is leaving a wall of 1's, or else the right one is fixed (part of a wall) and the left one is arriving.
6. A particle that just moved for the first time will kill the particle that used to be next to it on the left (if any); after that, it has no effect on other particles (and could be safely removed without altering the movement of the rest).
7. The behavior of a particle α can only be (directly) affected by the particle that precedes it (on the right) in the initial configuration. For a different particle, β, to affect α at some time, the intermediate particles would need to have vanished, but in that case β had to move, and by fact 6, it can no longer affect α.

Consider first the case $v_0^0 = 0$. One special case is when Bob has only 1's: the particle at $(1, 0)$ stays there until time $n-1$, ensuring $v_0^n = 0$. We claim that in any other case, $v_0^n = v_{-m}^m$ for $n = 2m$ and $v_0^n = v_{-m}^{m+1}$ for $n = 2m + 1$. To show this, we will show that $v_{-k}^{n-k} = v_{-k-1}^{n-k-1}$ for any $0 \leq k < \lfloor \frac{n}{2} \rfloor$. The idea is that the particle at $(-\lfloor \frac{n}{2} \rfloor, \lceil \frac{n}{2} \rceil)$ (if any) moves without interruption until it reaches the origin at time n.

- $v_{-k}^{n-k} = 1 \implies v_{-k-1}^{n-k-1} = 1$. Otherwise, the particle at $(-k, n-k)$ must come from $(-k, n-k-1)$, and by fact 2 we have that $v_{-k}^0 = v_{-k+1}^0 = \cdots = v_{n-2k}^0 = 1$, contradicting $v_0^0 = 0$.
- $v_{-k-1}^{n-k-1} = 1 \implies v_{-k}^{n-k} = 1$. Otherwise, the particle at $(-k-1, n-k-1)$ either stayed or died. If it stayed, fact 2 implies $v_{-k-1}^0 = v_{-k}^0 = \cdots = v_{n-2k-1}^0 = 1$, contradicting again $v_0^0 = 0$. If it died, then by fact 5 we have that either the particle at $(-k-1, n-k-1)$ or the one at $(-k+1, n-k-1)$ must have been fixed from the beginning; in either case, we use fact 2 to again contradict $v_0^0 = 0$.

For $k = \lfloor \frac{n}{2} \rfloor$ the argument fails, and indeed, the diagonal can change at that point. Thus the final result corresponds to the value at $(-\lfloor \frac{n}{2} \rfloor, \lceil \frac{n}{2} \rceil)$, which for even values of n depends on $v_{-n}^0, \ldots, v_0^0$, but for odd values requires $v_{-n}^0, \ldots, v_1^0$. Bob's protocol is therefore constant, and consists of two bits: one to signal whether he has only 1's, and another which equals v_1^0. By Lemma 3, Alice's protocol is constant too.

Consider now the case $v_0^0 = 1$; let β denote that particle. The protocol for Bob is simple: the dynamics of his particles (including β) does not depend on Alice's side. He tells Alice the last time t^* at which β remains at 0 , and whether in $t^* + 1$ it moved or died (equivalently, he may send the smallest p^* for which $v_{p^*}^0 = 0$, along with $v_{p^*+1}^0$). This is all Alice needs to know: from fact 6, anything that happens to β afterward is irrelevant, and from fact 7 other particles from Bob's side are irrelevant too.

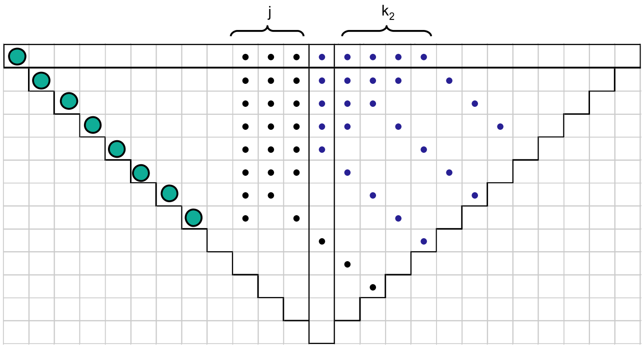

Fig. 9. Configuration for the lower bound of rule 152. We impose $j = \frac{1}{2}(n - k_2) - 1$ to ensure that the left of the block will remain fixed at position $-j$ until time $j + k_2$, at which time it will cause the traveling particle to vanish. Any right side with fewer than k_2 particles will let the particle pass through.

The remaining case is Alice's message to Bob, when $v_0^0 = 1$. There are two distinct possibilities, depending on whether the block of particles around v_0^0 has dissipated and moved past 0 by time n. If it has not, then it will determine v_0^n; all that Bob needs to learn in order to know if this is the case, and what the result will be, is the size k_A of the block on Alice's side (i.e., the number of particles contiguous to the one on $(0, 0)$).

To provide for the case in which the block does dissipate (something that only Bob will know), Alice sends a further bit of information, which is the value c of v_0^n under the assumption that Bob's side is filled with 0's: if Bob sees from k_A and his own side that the block will be gone by time n, then the answer is c. The value will be correct, even if the assumption does not hold:

- If $c = 1$, this particle ("β") must have been on the diagonal $(-n + t, t)$ (the bottom of the light-cone) since at least time $\frac{n}{2}$ (otherwise β would have been fixed for more than $\frac{n}{2}$ time steps in a position at distance less than $\frac{n}{2}$ from the origin, which together with fact 2 puts β inside the block around the origin, which is a contradiction).
- Even if the block takes longer to dissipate, the effect will be (if any) the destruction of some particles ahead of β which would have crossed into the right side otherwise; the effect will not affect β itself, since that would require reaching the bottom diagonal and then the block determines the result. There is no indirect effect either: any effect will be on particles which have already moved (otherwise they do not reach the block) and then fact 6 applies. This argument still applies in the case when β does not exist ($c = 0$).

There are therefore linear protocols for both Alice and Bob; in both cases there are additional bits (a constant part of the protocol) and a linear part which describe the size of the block of particles around the origin, in each direction.

Lower bound. Consider the set of values in $\{0, \dots, n-1\}$ that have the same parity as n, and let $k_1 < k_2$ be two values in it. Compute $j = \frac{1}{2}(n - k_2) - 1$; then

$$1 = f^n\left(10^{n-j-1}1^j, 1, 1^{k_1}0^{n-k_1}\right) \neq f^n\left(10^{n-j-1}1^j, 1, 1^{k_2}0^{n-k_2}\right) = 0$$

where the commas surround the center cell. Idea: the block around the center is just large enough, with k_2, to kill the particle coming from the left; any shorter block will vanish before and will let the particle reach 0 at time n. This gives the lower bound, since any pair of values (in a set of size $\frac{n}{2}$) defines a pair of right sides which are distinguished by an appropriate left side (Fig. 9).

Rule 168

We may represent this rule with

$$M_{168} = \{00\hat{1}, \overset{\curvearrowleft}{\bullet 1}\}$$

from which we realize that it behaves like rule 40, except for the case of a 1 preceded by 11: whereas in rule 40 it vanished (and 111 gave rise to a 00 wall), here it shifts; thus, everything shifts except when a constant, absorbent wall of ≥ 2 zeros is found. Moreover, the wall of zeros expands to the left by one position at each time step, and to the right every time it receives a shifting 0 (the same happened for rule 40).

The analysis and protocol mirror those of rule 40. If Bob has a wall of zeros, or if his rightmost cell has a 0, then the configuration evaluates to 0. Otherwise, the answer depends on whether the rightmost wall of zeros on Alice's side, together with all the isolated zeros to its right, reach the center or not. As before, the message from Bob gives Z_b, and the message from Alice gives $B + Z_a$.

Lower bound. Notice that the configurations in (2) do not contain 111; moreover, 111 does not appear in their evolution either. Hence, the family gives a linear lower bound for the present rule too: rules 168 and 40 coincide on them. □

5. Discussion

We have given protocols, and lower bounds of the same order, that together solve the one-round communication complexity of the prediction problem for about 1/3 of all elementary CAs, corresponding to the family of monotone rules. These CA can be represented as systems of interacting indistinguishable particles, a property which turned out to be very useful for first finding and then proving the different protocols. On the one hand, the particle representation is usually concise, and more intuitive than the local rule of the CA; on the other hand, it has the useful property of preserving the particles' order, which often allows us to disregard any particles other than the immediate neighbors.

Since the communication complexity was found to be at most linear for these rules, a corollary is that none of these rules can be intrinsically universal. A more vague but probably more important product of the proofs is the insight that the different protocols give into the dynamics of the rules: by decomposing the computation of the iterated local rule into the contributions of the left and right sides, we see how the future value depends on their interplay and on the incoming flow of information. This understanding is made specially intuitive thanks to the particle representation of the rules, protocols and proofs.

Despite the (at most) linear growth of their communication complexity, some of the rules are far from trivial: the relatively sophisticated protocol for rule 184 is a case in point. A further interesting aspect is the relation between the different rules and protocols: not only did some protocols and lower bound configurations apply to more than one rule, but even when they did not, there was a certain family resemblance between the constructions. Unfortunately, no general construction could be derived from this, and indeed, even if some generalization is possible, it would not encompass all DCAs, or even all NCCAs: Theorem 1 ensures that some of the latter will have exponential complexity.

Theorem 1 does not, however, ensure exponential complexity for 2-state CAs, and indeed, a reviewer of the present manuscript wondered whether perhaps all binary NCCA share some property giving them low complexities. We had considered the question too, and the answer is negative. To see why, let us sketch a direct simulation of arbitrary one-dimensional CAs, which, unlike Theorem 1, uses only binary NCCAs. Let F be a CA with $s > 2$ states and radius r. We can represent the s states through the blocks $\{1^k 0 1^{s-1-k} : 0 \leq k < s\}$, like $s-1$ beads in an abacus, and separate the rows of the abacus by putting a 1001 block between the "cell-blocks" representing adjacent cells of F. For instance, in a CA with states $\{s_0, s_1, s_2\}$ we would map a configuration $s_1 s_0 s_2 s_1$ to

1001.101.1001.011.1001.110.1001.101.1001

where the dots are added as a reference for the reader. They are not really required, since '00' only appears in the punctuating blocks. Thus any "bead" can check whether it is located in a valid configuration, with r additional cell-blocks to left and right; it will also know its place within its own cell-block. So, it can move (or stay) within its cell-block, to ensure that it changes according to F's rule (if the configuration is invalid, the particle does not move). What we obtain is an NCCA with two states and a neighborhood which is $4 + s$ times larger than that of F; this new CA simulates F in real time. There are therefore intrinsically universal binary NCCAs, which will have exponential communication complexity.

However, other general results may exist for more restricted families of CAs, and they are one of the open lines for future research. A possible starting point would be a generalization (if possible) of rule 184's protocol to more general "traffic" rules, as found in the literature [14]. A different direction would be the study of the communication complexity for problems different from PRED. Finally, since we are dealing with CAs that admit an alternative representation as particles, it may be interesting to consider questions of communication complexity stated in terms of them: this could be done by using the same setup as here, but asking questions about the particles' movements, or, alternatively, by using the particles' point of view and, for instance, the iterated neighborhood of their motion rule.

Acknowledgements

This work was partially supported by Fondecyt projects 1080592 (AM and EG) and 1090156 (IR), BASAL-CMM project (EG), Conicyt Anillo ACT-88, as well as DGIP-UTFSM grant 241016 (AM). Part of this work was done during a 2010 stay of EG at the Santa Fe Institute.

References

[1] C. Durr, I. Rapaport, G. Theyssier, Cellular automata and communication complexity, Theoretical Computer Science 322 (2004) 355–368.
[2] E. Goles, C. Little, I. Rapaport, Understanding a non-trivial cellular automaton by finding its simplest underlying communication protocol, in: S.-H. Hong, H. Nagamochi, T. Fukunaga (Eds.), Proceedings of the 19th International Symposium on Algorithms and Complexity, ISAAC 2008, in: Lecture Notes in Computer Science, vol. 5369, Springer, Gold Coast, Australia, 2008, pp. 593–605.
[3] N. Boccara, H. Fukś, Cellular automaton rules conserving the number of active sites, Journal of Physics A: Mathematical and General 31 (1998) 6007–6018.
[4] N. Boccara, H. Fukś, Number-conserving cellular automaton rules, Fundamenta Informaticae 52 (2001) 1–13.
[5] H. Fukś, A class of cellular automata equivalent to deterministic particle systems, in: S. Feng, A.T. Lawniczak, S.R.S. Varadhan (Eds.), Hydrodynamic Limits and Related Topics, American Mathematical Society, 2000, pp. 57–70.
[6] M. Pivato, Conservation laws in cellular automata, Nonlinearity 15 (2002) 1781–1794.
[7] A. Moreira, N. Boccara, E. Goles, On conservative and monotone one-dimensional cellular automata and their particle representation, Theoretical Computer Science 325 (2004) 285–316.
[8] A. Moreira, Universality and decidability of number-conserving cellular automata, Theoretical Computer Science 292 (2003) 711–721.

[9] E. Goles, P.-E. Meunier, I. Rapaport, G. Theyssier, Communication complexity and intrinsic universality in cellular automata, Theoretical Computer Science 412 (2011) 2–21.
[10] A. Yao, Some complexity questions related to distributed computing, in: 11th Annual Symposium on Theory of Computing, ACM Press, New York, 1979, pp. 209–213.
[11] E. Goles, P.-E. Meunier, I. Rapaport, G. Theyssier, Communications in cellular automata, in: T. Neary, D. Woods, T. Seda, N. Murphy (Eds.), International Workshop on the Complexity of Simple Programs, CSP08, in: Electronic Proceedings of Theoretical Computer Science, vol. 1, Elsevier, Cork, Ireland, 2009, pp. 81–92.
[12] E. Goles, P. Guillon, I. Rapaport, Traced communication complexity of cellular automata, in: P. de Oliveira, J. Kari (Eds.), 15th International Workshop on Cellular Automata and Discrete Complex Systems, AUTOMATA 2009, São José dos Campos, Brazil, 2009, pp. 318–327.
[13] E. Kushilevitz, N. Nisan, Communication Complexity, Cambridge University Press, 1997.
[14] S. Maerivoet, B.D. Moor, Cellular automata models of road traffic, Physics Reports 419 (2005) 1–64.