Necessary conditions for density classification by cellular automata

Mathieu S. Capcarrère* and Moshe Sipper
Logic Systems Laboratory, Swiss Federal Institute of Technology, 1015 Lausanne, Switzerland
(Received 18 April 2001; published 27 August 2001)

Classifying the initial configuration of a binary-state cellular automaton (CA) as to whether it contains a majority of 0*s* or 1*s*—the so-called density-classification problem—has been studied over the past decade by researchers wishing to glean an understanding of how locally interacting systems compute global properties. In this paper we prove two necessary conditions that a CA must satisfy in order to classify density: (1) the density of the initial configuration must be conserved over time, and (2) the rule table must exhibit a density of 0.5.

DOI: 10.1103/PhysRevE.64.036113 PACS number(s): 89.75.−k, 02.70.−c, 07.05.Bx, 89.70.+c

I. THE DENSITY-CLASSIFICATION PROBLEM

How does one obtain locally interacting systems that perform global computations? Such systems exhibit global information-processing capabilities that are not explicitly represented in their elementary components or in their local interconnections. Designing such *cellular computers* [1] is an arduous task, which has received much attention during the past several years.

Cellular automata (CA's) are the quintessential example of cellular computers, as well as the first to historically appear on the scene. A CA consists of a regular array of cells, each of which can be in one of a finite number of possible states, updated synchronously in discrete time steps, according to a local, identical interaction rule. The state of a cell at the next time step is determined by the current states of a surrounding neighborhood of cells. This transition is often specified in the form of a rule table, delineating the cell's next state for each possible neighborhood configuration.

An example of a cellular computation is to use a CA to determine the global density of bits in an initial-state configuration. This *density-classification problem* has been studied extensively over the past decade. Packard [2] was the first to introduce the following version of the problem: a one-dimensional (1D), two-state CA is presented with an arbitrary initial configuration, and should converge in time to a state of all 1*s* if the initial configuration contains a density of 1*s*>0.5, and to all 0*s* if this density <0.5; for an initial density of 0.5, the CA's behavior is undefined [Fig. 1(a)]. Spatially periodic boundary conditions are used, resulting in a circular grid. Though this version was proved to be unsolvable [3], it has nonetheless attracted several researchers aiming to evolve high-performance (though imperfect) CA rules by employing evolutionary algorithms [4–6].

Capcarrère, Sipper, and Tomassini [7] showed that there exists a perfect solution to the density-classification problem (i.e., one that classifies all input configurations correctly), upon defining a different output specification [Fig. 1(b)].

Considering the problem of density classification by cellular automata, we prove two necessary conditions that a CA must satisfy in order to classify density perfectly:

(1) The density of the initial configuration must be conserved over time.

(2) The rule table must exhibit a density of 0.5.

The first condition is of particular interest as it creates a link between the problem of density classification and the well-studied class of density-conserving CA's. Effectively, these latter have received much attention within the physics community, e.g., for modeling of traffic flow [8] and surface growth [9].

II. NOTATION AND DEFINITIONS

A *configuration* is the state of all cells of the CA at a given time step. The *transition rule s* is the complete lookup table, delineating a cell's state at the next time step for every possible local configuration of neighboring states. The *successor function S* is derived by simultaneously applying *s* to the entire configuration yielding the configuration at the next time step. σ denotes a configuration of states, σ_0 denotes the input configuration at time $t=0$, and σ_t denotes the configuration at time step *t*, resulting from *t* successive applications of *S* to σ_0, i.e., $\sigma_t = S^t(\sigma_0)$.

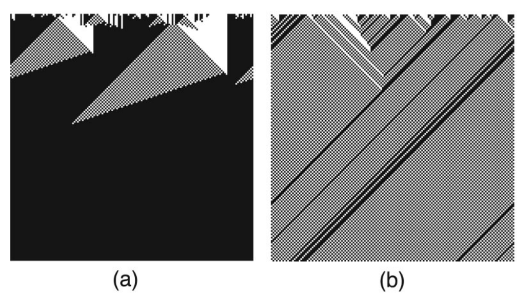

FIG. 1. Two 1D CA density classifiers. White squares represent cells in state 0, black squares represent cells in state 1. Grid size is $n=149$. The pattern of configurations is shown for the first 150 time steps, with time increasing down the page. The random initial configuration (i.e., input) contains a majority of 1*s* in both cases. (a) The GKL CA ($r=3$), which correctly classifies approximately 81.5% out of a random sample of initial configurations. (b) The CA of Capcarrere *et al.* [7] ($r=1$) which classifies perfectly all initial configurations using a different output definition; if there is a majority of 1*s* (respectively, 0*s*) in the input, then the output consists of one or more blocks of at least two consecutive 1*s* (0*s*), interspersed by an alternation of 0*s* and 1*s*.

*Email address: mathieu capcarrere@epfl.ch; lslwww.epfl.ch

Let $I(\sigma)$ be the number 1*s* of configuration σ. Density $D(\sigma)$ thus equals $I(\sigma)/|\sigma|$, where $|\sigma|$ is the length (i.e., number of cells) of σ. The bitwise inversion of configuration σ is denoted by $\overline{\sigma}$.

Following Wolfram [10], the transition rule *s* can be written as a string containing the next-state bit for every neighborhood configuration.

For 1D CA's, $\sigma^{(i,j)}$ denotes the $|i-j|$ bits of configuration σ positioned between bits *i* and $(j-1)$, inclusive. $(\sigma)^k$ is the concatenation of *k* configurations σ.

In this paper we consider two-state, *d*-dimensional toroidal CA's, whose radius *r* is defined as an extension of the von-Neumann neighborhood: a cell has *r* neighbors on both sides of each dimension; in addition, the cell itself is included in its neighborhood.

The density-classification problem is defined as follows:

Definition. Considering a toroidal, two-state CA, a successor function *S* is said to be a *perfect density classifier*, if *S*, when applied to an arbitrary initial configuration of any length, progresses toward a configuration, that allows to effectively distinguish whether the density of 1*s*, in the original configuration, is greater or smaller than a predetermined threshold ρ. [This definition is not mathematically tight, as it rests upon the notion of "effective computation"—as indeed does the famous Church-Turing thesis. We have opted for such a definition because, otherwise, many clearly ineffective CA's might be considered as density classifiers (e.g., the identity rule, which simply maps any configuration to itself).]

III. A PERFECT DENSITY CLASSIFIER MUST CONSERVE DENSITY

In this section we prove that a perfect CA density classifier cannot alter the density of the input configuration. We first prove this result for one-dimensional CA's and then provide an informal argument as to the validity of the proof to any dimension.

Theorem 1. Let *S* be a successor function of a perfect one-dimensional density classifier. Then

$$\forall \sigma_0,\ \forall t,\ D(\sigma_0)=D[S^t(\sigma_0)].$$

The proof of this theorem involves five lemmas proved below.

Lemma 1.1. Let *S* be a perfect density classifier successor function. Then, $\forall \sigma_0, \forall t$, $D(\sigma_0)<\rho \Rightarrow D[S^t(\sigma_0)]<\rho$, and $D(\sigma_0)>\rho \Rightarrow D[S^t(\sigma_0)]>\rho$.

Proof. This follows straightforwardly from our earlier definition of the density-classification problem. Since a CA is deterministic and memoryless, if it ever reaches a configuration σ_n belonging to the complementary class, it will then wrongly classify σ_0 and σ_n as belonging to the same class.

Lemma 1.2. Let *s* be the transition rule of a perfect density classifier with radius *r*. Then, $s(0^{2r+1})=0$ and $s(1^{2r+1})=1$.

Proof. If $s(0^{2r+1})=1$ and $s(1^{2r+1})=1$, or $s(0^{2r+1})=0$ and $s(1^{2r+1})=0$, then the input configurations 0^n and 1^n, where *n* is the size of the CA, are classified as belonging to the same class, thus contradicting *s*'s being a perfect density classifier transition rule. If $s(0^{2r+1})=1$ and $s(1^{2r+1})=0$, then 0^n and 1^n give rise to a cycle of alternating configurations, thus contradicting *s*'s being a perfect density classifier [12].

Lemma 1.3. For any input configuration σ_0 of size *n*, and for any density threshold ρ, there exist m_0, m_1 such that $D(0^{m_0}\sigma_0 1^{m_1})>\rho$ and $D(0^{(m_0+1)}\sigma_0 1^{(m_1-1)})<\rho$.

Proof. Assuming $1/(1-\rho)$ is not an integer, then, setting $m_0+m_1=\lceil n/(1-\rho)\rceil-n$, it is straightforward to see that if $I(\sigma_0)=0$, we can set $m_1=\lceil n/(1-\rho)\rceil-n$ and $m_0=0$, with the result that $D(0^{m_0}\sigma_0 1^{m_1})>\rho$ and $D(0^{(m_0+1)}\sigma_0 1^{(m_1-1)})<\rho$. Now, if $I(\sigma_0)\neq 0$, then decreasing m_1 by $I(\sigma_0)$ and increasing m_0 by the same amount will satisfy the lemma.

If $1/(1-\rho)$ is an integer, then setting $m_0+m_1=\lceil n/(1-\rho)\rceil-n+1$ leads to the same result.

Lemma 1.4. Let *S* be the successor function for a one-dimensional CA, σ_0 an initial configuration, and *p* an integer, such that $I[S(\sigma_0)]=I(\sigma_0)+p$. Then, $I\{S[(\sigma_0)^k]\}=I[(\sigma_0)^k]+kp$.

Proof. As our CA's are toroidal, $S[(\sigma_0)^k]=(\sigma_1)^k$. Then, $I\{S[(\sigma_0)^k]\}=I[(\sigma_1)^k]=k*I(\sigma_1)=k*I(\sigma_0)+kp=I[(\sigma_0)^k]+kp$.

Lemma 1.5. Let *S* be a successor function of a perfect one-dimensional density classifier, and let *r* be the radius of the CA. For any configuration σ_0, if $I[S(\sigma_0)]=I(\sigma_0)+p$ then $-4r\leqslant p\leqslant 6r$.

Proof. Let σ_0 be a configuration such that $I[S(\sigma_0)]=I(\sigma_0)+p$. Define a configuration υ_0, such that $\upsilon_0=0^{m_0}R_1\sigma_0 R_2 1^{m_1}$, where $R_2=\sigma_0^{(0,r)}$ and $R_1=\sigma_0^{(n-r,n)}$ and $m_0,m_1\geqslant 2r+1$.

Then, given Lemma 1.2 and our definition of R_1 and R_2, we conclude that $S(\upsilon_0)=C_1 0^{m_0-2r}C_2\sigma_1 C_3 1^{m1-2r}$, where C_1 is the $2r$-bit-long configuration obtained at the border of $1^{2r}0^{2r}$, C_2 is the *r*-bit-long configuration obtained at the border of $0^{2r}R_1$, and C_3 is the *r*-bit-long configuration obtained at the border of $R_2 1^{2r}$.

From Lemma 1.3 we know that we can define m_0, m_1 such that $D(\upsilon_0)>\rho$, and that if we decrease m_1 by 1 and increase m_0 by 1, $D(\upsilon_0)<\rho$. (Note that we can increase both m_0 and m_1 by $2r+1$ so that $m_0,m_1\geqslant 2r+1$ as required above.) Then, as $D(\upsilon_0)>\rho$, we know that $D(\upsilon_1)>\rho$ (Lemma 1.1), which, given the chosen values of m_0, m_1, implies that $I(\upsilon_1)\geqslant I(\upsilon_0)$. Expanding $I(\upsilon_1)$ and $I(\upsilon_0)$, we can derive that $I(C_1)+I(C_2)+I(C_3)+p-2r-I(R_1)-I(R_2)\geqslant 0$.

Analogously, if we define m_0, m_1 such that $D(\upsilon_0)<\rho$ and if we decrease m_0 by 1 and increase m_1 by 1, then $D(\upsilon_0)>\rho$. Then, as $D(\upsilon_0)<\rho$, we know that $D(\upsilon_1)<\rho$ (Lemma 1.1), which, given the chosen values of m_0, m_1, implies that $I(\upsilon_1)\leqslant I(\upsilon_0)$, from which we derive that $I(C_1)+I(C_2)+I(C_3)+p-2r-I(R_1)-I(R_2)\leqslant 0$.

Hence, we know that $I(C_1)+I(C_2)+I(C_3)+p-2r-I(R_1)-I(R_2)=0$, meaning that *p*—the variation of number of 1*s* between σ_0 and σ_1—is exactly equal to $I(R_1)+I(R_2)-I(C_1)-I(C_2)-I(C_3)+2r$. Given the lengths of R_1, R_2, C_1, C_2, and C_3, we compute that $-4r\leqslant p\leqslant 6r$.

We are now able to prove Theorem 1.

Proof of Theorem 1. We will proceed by contradiction.

Assume there exists a configuration σ_0, such that $I[S(\sigma_0)]=I(\sigma_0)+p$, p being a nonzero integer. From Lemma 1.4 we know that we can create a configuration $\tau_0=(\sigma_0)^k$, such that $I[S(\tau_0)]=I(\tau_0)+kp$. However, if we set $k=7r$, where r is the radius of the CA in question, then we have a configuration τ_0, wherein $I[S(\tau_0)]=I(\tau_0)+7rp$, which contradicts Lemma 1.5, since $p\neq 0$.

Hence $p=0$, and thus, for all configurations σ_0, $I[S(\sigma_0)]=I(\sigma_0)$.

To avoid a lengthy proof, we provide an informal argument as to the validity of Theorem 1 to d-dimensional CA's. Lemmas 1.1 and 1.2 straightforwardly hold for any dimension. Lemma 1.3 can be extended to any dimension if we define the blocks 0^{m_0} and 1^{m_1} to be n-dimensional blocks stacked up along the same dimension on each side. To extend Lemma 1.4 to d dimensions, we note that if $I[S(\sigma_0)]=I(\sigma_0)+p$, a configuration υ_0 can be defined as the d-dimensional vector of k stacking up of σ_0 along any one dimension; then, $I[S(\upsilon_0)]=I(\upsilon_0)+kp$. Finally, taking into account the aforementioned modification for Lemma 1.3, we would obtain a bounded value for p in Lemma 1.5 (albeit different from the one for the one-dimensional case) but still independent from the size of the chosen configuration. Thus, having proved both the necessity of a bounded variation of 1s and the possibility of creating a configuration with as large a variation of 1s as desired, theorem 1 holds for d dimensions.

IV. A PERFECT DENSITY CLASSIFIER'S RULE MUST EXHIBIT A DENSITY OF 0.5

Having obtained a necessary condition on the global successor function S, we prove in this section a theorem relating to the local transition rule, s, namely, it must exhibit a density of 0.5.

Theorem 2. Let s be the transition rule of a perfect, two-state, toroidal density classifier of any dimension. Then, for any density threshold of 1s, ρ, $D(s)=0.5$.

The proof of this theorem involves five lemmas and a result on consecutive-l graphs proved by Ref. [11].

A *consecutive-l graph*, $G(l,n,q,h)$, is an n-node directed graph, wherein there exists an edge, (i,j), if and only if $j\in\{qi+k(\text{mod n}):h\leqslant k\leqslant h+l-1\}$. Du *et al.* [11] proved that such a graph contains a Hamiltonian cycle [13] if $q=l$, $h=0$, and $l\geqslant gcd(n,q)\geqslant 2$.

Lemma 2.1. For any radius r, one-dimensional, two-state toroidal CA, there exists a configuration σ_0 of length 2^{2r+1}, such that all 2^{2r+1} neighborhoods are present once and only once.

Proof. Consider the directed graph G, whose vertices are the 2^{2r+1} binary numbers $0,\ldots,2^{2r+1}-1$, defined as follows: there is an edge from vertex v_n to vertex v_m, if and only if the last $2r$ bits of v_n are identical to the first $2r$ bits of v_m. Then, finding a Hamiltonian cycle in G is equivalent to finding an input configuration σ_0 satisfying the conditions of the lemma.

The set of edges of G can be defined as follows: $i\rightarrow j$ if $j\in\{2i+k(\text{mod n}):0\leqslant k\leqslant 1\}$. We thus obtain a *consecutive-2* directed graph, $G(l,n,q,h)$, with $q=l=2$ and $h=0$. As the number of nodes n is a power of 2, we have $q=l,h=0$ and $h\geqslant gcd(n,q)\geqslant 2$. Thus, following the results of Ref. [11] G contains a Hamiltonian cycle, thereby proving the lemma.

Lemma 2.2. Let σ_0 be a d-dimensional configuration of length 2^{2dr+1}, such that all 2^{2dr+1} neighborhoods of a *d-dimensional* CA are present once and only once. Then, $\overline{\sigma}_0$, the bitwise inversion of σ, is also such a configuration.

Proof. Consider any two of the 2^{2dr+1} possible neighborhoods of $\overline{\sigma}_0$: $\overline{a}$ and $\overline{b}$. Then, by definition, there exist a,b, the two corresponding neighborhoods of σ_0. As each neighborhood is present once and only once, $a\neq b$, and thus $\overline{a}\neq\overline{b}$. Given that there are only 2^{2dr+1} neighborhoods in $\overline{\sigma}_0$, and given that there are 2^{2dr+1} possible different neighborhoods for a d-dimensional CA, then all neighborhoods are present once and only once in $\overline{\sigma}_0$.

Lemma 2.3. For any constant r, there exists a one-dimensional, two-state configuration σ_0 of length 2^{2r}, such that for any 2 blocks a and b of length $2r+1$ in σ_0, $a\neq b$.

Proof. One may see that the proof of Lemma 2.1 still holds for even powers of 2. Thus, we know that there exists a configuration of length 2^{2r}, such that any $2r$-long block $a_i\ldots a_{(i+2r-1)mod\,2^{2r}}$ is different from any other $2r$-long block $a_j\ldots a_{(j+2r-1)mod\,2^{2r}}$, $i\neq j$. In such a configuration, any $2r+1$-long block $a_i\ldots a_{(i+2r)mod\,2^{2r}}$ is thus different from any other $2r+1$-long block $a_j\ldots a_{(j+2r)mod\,2^{2r}}$, $i\neq j$.

Lemma 2.4. For any d-dimensional, 2-state toroidal CA, and for any radius r, there exists a configuration wherein all 2^{2dr+1} possible neighborhoods are present once and only once.

Proof. We will prove this lemma by induction.

The base of the induction, $d=1$, is proved by Lemma 2.1.

Induction step—Assume a d-dimensional configuration σ_0 that includes all 2^{2dr+1} possible neighborhoods, each present once and only once.

We next construct $\beta_0=\alpha_1\ldots\alpha_{2^{2r}}$, the $(d+1)$-dimensional configuration, by "stacking up" along the $(d+1)$th dimension 2^{2r} $\alpha' s$, where $\alpha\in\{\sigma_0,\overline{\sigma_0}\}$. We construct the sequence $\alpha_1\ldots\alpha_{2^{2r}}$, such that any block $\alpha_i\ldots\alpha_{(i+2r)mod\,2^{2r}}$ is different from any other block $\alpha_j\ldots\alpha_{(j+2r)mod\,2^{2r}}$, $i\neq j$. One can see this is possible: if we denote the case $\alpha=\sigma_0$ by 0 and the case $\alpha=\overline{\sigma_0}$ by 1, we can then invoke Lemma 2.3.

From the induction assumption and from Lemma 2.2, we know that along each hyperplane α_i there are 2^{2dr+1} different neighborhoods. Each of these neighborhoods includes along its $(d+1)$th dimension the sequence of bits $b_{(i-r)mod\,2^{2r}}\ldots b_{(i+r)mod\,2^{2r}}$. We know that this sequence is different for each hyperplane from the construction constraint that any block $\alpha_i\ldots\alpha_{(i+2r)mod\,2^{2r}}$ is different from any other block $\alpha_j\ldots\alpha_{(j+2r)mod\,2^{2r}}$, $i\neq j$. Thus, all 2^{2dr+1} different neighborhoods on hyperplane α_i are different from all 2^{2dr+1} different neighborhoods on hyperplane α_j, $i\neq j$. Then, we know that we have $2^{2r}*2^{2dr+1}=2^{2(d+1)r+1}$ different neighborhoods in β_0, which is also the maximum number of possible neighborhoods. Thus, β_0 is a configuration of dimension $d+1$, in which all $2^{2(d+1)r+1}$ possible neighborhoods are present once and only once. This proves the induction step d to $d+1$.

Lemma 2.5. Let σ_0 be a d-dimensional configuration, such that all 2^{2dr+1} possible input states are present once and only once, in any dimension d. Then, $D(\sigma_0)=0.5$.

Proof. The density of all neighborhoods, i.e., the density of all the numbers from 0 to $2^{2dr+1}-1$ is 0.5. When "moving" along σ_0 to collect all neighborhoods, each bit is counted exactly the same number of times, namely, $2dr+1$ times. Thus, the density of σ_0 is the same as the density of all possible neighborhoods, i.e., 0.5.

We are now able to prove Theorem 2.

Proof of Theorem 2. Assume configuration σ_0 contains all 2^{2dr+1} possible neighborhoods once and only once. From Lemma 2.4 we know that such a σ_0 exists. From Theorem 1 we deduce that—given that S is a perfect density classifier successor function—$D[S(\sigma_0)]=D(\sigma_0)$, which, from Lemma 2.5, we know to be 0.5. Moreover, as all 2^{2dr+1} possible neighborhoods are present once and only once, then $D[S(\sigma_0)]=D(s)$, and hence $D(s)=0.5$.

V. CONCLUSION

We have shown that a perfect CA density classifier must conserve the density in time of the initial configuration, and its rule table must exhibit a density of 0.5. Thus, nondensity-conserving CA's [such as the GKL rule of Fig. 1(a)], or, indeed, any specification of the problem that involves density change, precludes the ability to perform perfect density classification. These two necessary conditions might thus aid in the search for locally interacting systems that compute the global density property.

ACKNOWLEDGMENTS

The authors thank Eric Tannier from the Leibniz Laboratory at Imag, Grenoble, France, for his crucial remark on the use of graphs.

[1] M. Sipper, IEEE Computer **32**, 18 (1999).
[2] N. H. Packard, in *Dynamic Patterns in Complex Systems*, edited by J. A. S. Kelso, A. J. Mandell, and M. F. Shlesinger (World Scientific, Singapore, 1988), pp. 293–301.
[3] M. Land and R.K. Belew, Phys. Rev. Lett. **74**, 5148 (1995).
[4] M. Mitchell, P.T. Hraber, and J.P. Crutchfield, Complex Syst. **7**, 89 (1993).
[5] M. Sipper, *Evolution of Parallel Cellular Machines: The Cellular Programming Approach* (Springer-Verlag, Heidelberg, 1997).
[6] H. Juillé and J. B. Pollack, in *Genetic Programming 1998: Proceedings of the Third Annual Conference*, edited by J. R. Koza (Morgan Kaufmann, San Francisco, 1998), pp. 519–527.
[7] M.S. Capcarrère, M. Sipper, and M. Tomassini, Phys. Rev. Lett. **77**, 4969 (1996).
[8] K. Nagel and H.J. Hermann, Physica A **199**, 254 (1993).
[9] V. Belitsky and P.A. Ferrari, J. Stat. Phys. **80**, 517 (1995).
[10] S. Wolfram, Rev. Mod. Phys. **55**, 601 (1983).
[11] D.Z. Du, D.F. Hsu, and F.K. Hwang, Math. Comput. Modelling **17**, 61 (1993).
[12] In theory, we could add an additional time constraint to enable the use of $s(0^{2r+1})=1$ and $s(1^{2r+1})=0$. Note that Theorem 1 would still hold if we were to include it. However, we will avoid this unnecessary complication.
[13] A graph G is said to have a Hamiltonian cycle, if and only if there exists a cycle going through every node, once and only once.