IJCSS International Journal of Complex Systems in Science

Int. J. Complex Systems in Science
vol. **1(2)** (2011), pp. 119–123

Automata with memory on proximity graphs

Ramón Alonso-Sanz[1,†]

[1] *Universidad Politécnica de Madrid, Estadística-GSC*

Abstract. In proximity graphs to every pair of nodes it is assigned a certain vicinity, and the pair is connected if its vicinity is empty. In the automata studied here each node is characterized by a binary state and their updating is made according to a rule involving the neighborhood of each node. The effect of different types and degrees of memory of the past states embedded in nodes is assessed when considering a particularly active rule, namely the parity rule.

Keywords: Automata, Memory, Proximity Graphs
MSC 2000: 91E40, 05C10, 37N99

[†] **Corresponding author:** ramon.alonso@upm.es
Received: October 14, 2011
Published: October 24, 2011

1. Automata on proximity graphs

Given a set $\mathbf{P}$ of $\mathtt{N}$ nodes in the plane, in proximity graphs to every pair of nodes is assigned a certain neighbourhood, and the pair is connected if its neighbourhood is empty. Two kinds of proximity graphs will be considered in this study: β-skeletons and Delaunay triangulations.

In β-skeletons [6] the β-neighborhood is defined as: the intersection of two circles of radius $d(p,q)/2\beta$ that pass through p and q, if $\beta \in [0,1]$; the intersection of two circles of radius $\beta d(p,q)/2$ centered at the points $(1-\beta/2)p+(\beta/2)q$ and $(\beta/2)p+(1-\beta/2)q$, if $\beta \geq 1$. If β varies continuously from 0 to ∞, β-skeletons form a sequence of graphs extending from the complete graph to the (nearly) empty graph, i.e., with decreasing mean degree.

A Delaunay triangulation for a set $\mathbf{P}$ is a triangulation DT($\mathbf{P}$) such that no node in $\mathbf{P}$ is inside the circumcircle of any triangle in DT($\mathbf{P}$)[4].

In the automata on proximity graphs studied here, each node i is characterized by a binary state $\sigma_i \in \{0,1\}$. The updating of these states is made simultaneously in discrete time-steps (T), according to a common local transition rule (ϕ) involving only the neighborhood of each node $\mathcal{N}_i$. Thus, the site values evolve by iteration of the mapping : $\sigma_i^{(T+1)} = \phi\Big(\{\sigma_j^{(T)}\} \in \mathcal{N}_i\Big)$.

This article deals with the parity rule: $\sigma_i^{(T+1)} = \sum\limits_{j \in \mathcal{N}_i} \sigma_j^{(T)} \mod 2$. Despite its formal simplicity, the parity rule may exhibit complex behaviour [5].

2. Memory

In the Markovian approach just outlined, the transition depends on the configuration of the nodes only at the preceding time-step. Explicit historic memory can be embedded in the dynamics keeping ϕ unaltered, by featuring every node by a mapping of its states in the past [1]. Thus, $\sigma_i^{(T+1)} = \phi\Big(\{s_j^{(T)}\} \in \mathcal{N}_j\Big)$, $s_j^{(T)}$ being a trait state function of the series of states of the node j up to T [2]. In the particular case of the parity rule: $\sigma_i^{(T+1)} = \sum\limits_{j \in \mathcal{N}_i} s_j^{(T)} \mod 2$.

We will consider in the present study two kind of memories: The most frequent state (or *majority*), and decaying weighted memory.

With majority memory limited to the last τ time-steps:

$$s_i^{(T)} = mode\big(\sigma_i^{(T)}, \sigma_i^{(T-1)}, \ldots, \sigma_i^{(\top)}\big),$$

[1]A kind of memory-enrichment of the feed-back which is readily implementable in any discrete iterative system [1].

with $\top = \max(1, T-\tau+1)$. In the case of equality in the number of time-steps that a node was 0 and 1, the last state is kept, in which case memory does not really actuate. This lack of effect of memory induces a lower effectiveness of even size τ-memories.

Historic memory can be weighted by applying a geometric decaying mechanism based on a memory factor $\alpha \in [0,1]$. Thus, the trait state s of every node is the rounded weighted mean (m) of its previous states. Formally:

$$m_i^{(T)} = \frac{\sigma_i^{(T)} + \sum_{t=1}^{T-1} \alpha^{T-t} \sigma_i^{(t)}}{1 + \sum_{t=1}^{T-1} \alpha^{T-t}} \quad \rightarrow \quad s_i^{(T)} = \begin{cases} 1 & if \quad m_i^{(T)} > 0.5 \\ \sigma_i^{(T)} & if \quad m_i^{(T)} = 0.5 \\ 0 & if \quad m_i^{(T)} < 0.5 \ . \end{cases}$$

The choice α simulates the remnant memory effect: the limit case $\alpha = 1$ corresponds to $full$ memory (equivalent to unlimited trailing *majority* memory), whereas $\alpha \ll 1$ intensifies the contribution of the most recent states (short-term memory). If $\sigma \in \{0,1\}$, α-memory is effective only with $\alpha > 0.5$.

3. Results

In order to assess the effect of memory, the evolution of the changing rate (the relative Hamming distance between two consecutive patterns) and of the damage rate (the relative Hamming distance between patterns resulting from reversing the initial state of a single node) are shown in the long-term (average of the last five iteration rates) in Figs.1 and 2 .

Both changing and damage rates reach a 0.5 level in the conventional ahistoric model. Memory exerts an inertial effect which induces the depletion of both parameters. The depleting effect of memory is much more remarkable in skeletons with low connectivity, e.g., β=2.0 (Relative Neighborhood Graph), whereas skeletons with high connectivity, e.g., β=0.9, are hardly restrained with memory. Skeletons with connectivity so to say in the middle range, e.g., β=1.0 (Gabriel graph), are affected by memory somehow in between the extreme scenarios.

Memory notably restrains the changing rate, albeit turns out rather ineffective in the control of the damage in networks with high connectivity, i.e., β-skeletons with low β such as β=0.9 . In a simulation up to T time-steps, the maximum depleting effect of majority memory on the changing rate (Fig.1 left) is achieved with a memory length $\tau \simeq T/2$. With very high memory length, the initial oscillatory-like effect induced by majority memory in the changing rate is maintained too long to allow an effective depletion in the

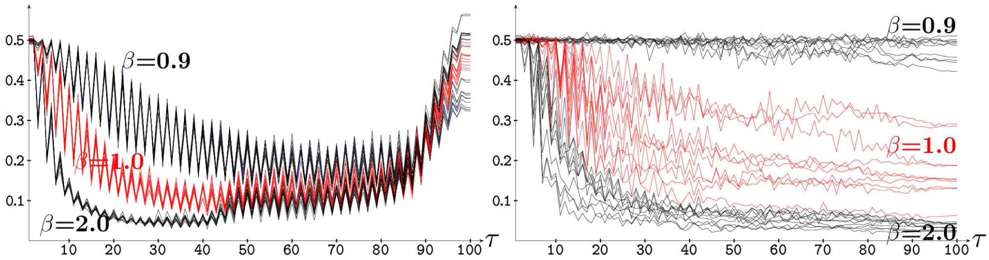

Figure 1: Long-term changing rate (left) and damage (right) in ten N=1000 parity β-skeletons with τ-majority memory run up to T=100 .

changing rate. That is so even in low-connected networks, i.e., skeletons with large β .

Weighted memory (Fig. 2) shows a smoother effect, with increasing effectiveness according to the increase of the memory factor α . Thus, increasing the memory factor implies restrain in the changing rate in the left panel of Fig. 2, though very weakly when low memory charge is implemented. The sharp peak in the full memory (α=1.0) scenario agrees with the increase observed with full length (τ=100) majority memory in Fig. 1 . The restrain of damage is achieved only with a memory factor greater than 0.7 .

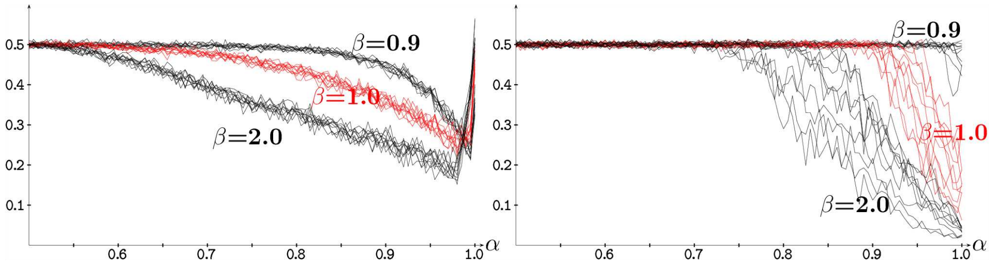

Figure 2: Long-term changing rate (left) and damage (right) in ten N=1000 parity β-skeletons with α-memory run up to T=100 .

In large DT the mean connectivity turns out to be $\overline{K}$=6 . Consequently, the effect of memory on the parity rule on DT is comparable to that on β-skeletons with high connectivity. The evolving dynamics of the changing rate and damage in parity DT automata are in fact reminiscent of that on the β=0.9 skeletons shown in Figs.1 and 2 . In particular, memory turns out rather ineffective in the control of damage on DT. Considering the triangles of a DT as *cells*, connected if they are adjacent, the scenario would be that of irregular (triangular) *cellular* automata. These triangular DT cellular automata will have mean degree equal to three, a low connectivity which allows for a higher effectiveness of memory in restraining the changing rate and damage spreading, much as in the β=2.0 skeletons shown in Figs.1 and 2 .

References

[1] Alonso-Sanz,R.,*Discrete Systems with memory*, World-Scientific (2011).

[2] Alonso-Sanz,R.,Adamatzky,A.,*Journal of Computational Science* **2,1**, 57-66 (2011).

[3] Adamatzky,A.,*Journal of Computational Science* **1**, 175-186 (2010).

[4] Berg,M. de,Cheong,O.,M.,Kreveld M. van,Overmans,O.(2008). *Computational Geometry*. Springer.

[5] Julian,P.,Chua,L.O., *Int. J. Bifurcation and Chaos* **12**, 479-494 (2002).

[6] Kirkpatrick,D.G,Radke,J.D., In G. Toussaint (ed.), Computational geometry, 217,248. North-Holland, 1985.