Journal of Computational Science 2 (2011) 57–66

Contents lists available at ScienceDirect

Journal of Computational Science

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

On beta-skeleton automata with memory

Ramón Alonso-Sanz [a,*], Andrew Adamatzky [b]

[a] Universidad Politecnica de Madrid, ETSI Agronomos (Estadistica, GSC), C. Universitaria, 28040 Madrid, Spain
[b] University of the West of England, Bristol Frenchay Campus, Bristol BS16 1QY, UK

ARTICLE INFO

Article history:
Received 20 August 2010
Received in revised form 1 December 2010
Accepted 2 December 2010
Available online 24 December 2010

Keywords:
Beta-skeletons
Automata
Memory

ABSTRACT

A β-skeleton is a proximity undirected graph whose connectivity is determined by the parameter β. We study β-skeleton automata where every node is a finite state machine taking two states, and updating its states depending on the states of adjacent automata-nodes. We allow automata-nodes to remember their previous states. In computational experiments we study how memory affects the global space–time dynamics on β-skeleton automata.

1. Introduction

In computational geometry and geometric graph theory, a β-skeleton or beta skeleton is an undirected proximity graph defined from a set of points in the Euclidean plane. Two points p and q are connected by an edge whenever their β neighborhood is empty [24]. In lune-based β-skeletons, 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 $\beta \geq 1$.

Fig. 1 shows three β-skeletons with increasing β parameter value, based on the same 10 nodes. The figure shows also the β-neighborhoods of the node labeled 1 (upper-right). In the transition from $\beta = 0.9$ to $\beta = 1.0$, the 1-node loses its 1–6 and 1–8 links, whereas from $\beta = 1.0$ to $\beta = 1.5$ it loses the 1–5 link.

β-Skeletons belong to a family of proximity graphs. which are monotonously parameterised by the parameter β. The structure of proximity graphs represents a wide range of natural systems and is applied in many fields of science. Few examples include geographical variational analysis [20,27,33], evolutionary biology [26], simulation of epidemics [36], study of percolation [18] and magnetic field [35], design of ad hoc wireless networks [25,28,31,34,40]. Thus developing and analysing computational models of spatially extended systems on proximity graphs will shed a light onto basic mechanisms of activity propagation on natural systems.

Automata on β-skeletons were originally introduced in [5] and studied in a context of excitation dynamics. In the present paper we develop ideas of [5] along lines of memory-enriched automata and global dynamics. The paper is structured as follows. In Section 2 we define automata on β-skeletons. Global dynamics on automata for $\beta \geq 1$ (planar graphs) are studied in Sections 2.1 and 2.3, whereas the $\beta < 1$ (non-planar graphs) is studied in Section 2.2. Section 4 demonstrates the effects of weighted memory on the behaviour of automata networks. Automata networks with non-parity rules are briefly tackled in Section 5. Possible applications of our computational findings are discussed in Section 6.

2. Automata on beta-skeletons

In the automata on beta-skeletons studied here, each node is characterized by an internal state whose value belongs to a finite set. The updating of these states is made simultaneously (*à la* cellular automata) according to a common local transition rule involving only the neighborhood of each node [5]. Thus, if $\sigma_i^{(T)}$ is taken to denote the state value of node i at time step T, the site values evolve by iteration of the mapping: $\sigma_i^{(T+1)} = \phi(\{\sigma_j^{(T)}\} \in N_i)$, where N_i is the set of nodes in the neighborhood of i and ϕ is an arbitrary function which specifies the automaton *rule*. This article deals with two possible state values at each site: $\sigma \in \{0,1\}$, and the parity rule: $\sigma_i^{(T+1)} = \sum_{j \in N_i} \sigma_j^{(T)} \bmod 2$. Despite its formal simplicity, the parity rule may exhibit complex behaviour [23].

In the Markovian approach just outlined (referred as *ahistoric*), the transition function depends on the neighborhood configuration of the nodes only at the preceding time step. Explicit historic memory can be embedded in the dynamics by featuring every node by

* Corresponding author.
E-mail addresses: ramon.alonso@upm.es (R. Alonso-Sanz), andrew.adamatzky@uwe.ac.uk (A. Adamatzky).

1877-7503/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.jocs.2010.12.001

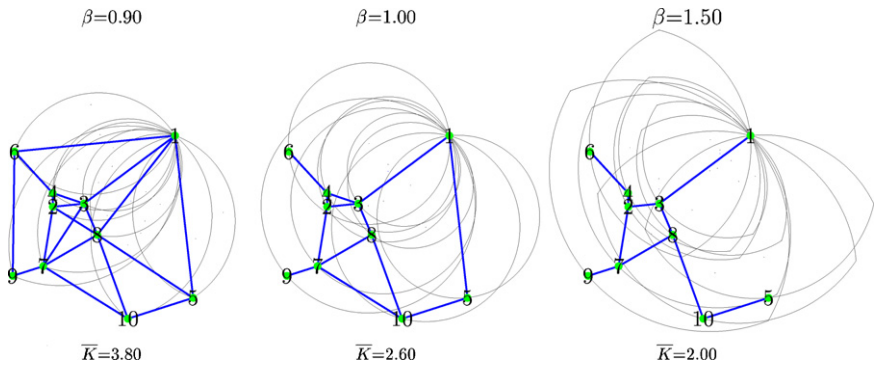

Fig. 1. Three β-skeletons on the same 10 nodes.

Table 1
The core of the FORTRAN progam.

```
DO it=1,maxit
   call MEMORY(MODE,NEW,ISIG,n,it,maxit,itau)
   DO i=1,n;iadd=0
      do j=1,n;if(ADJ(i,j)==0)cycle
             iadd=iadd+MODE(j)
      enddo
      NEW(i)=mod(iadd,2)
   ENDDO
ENDDO

subroutine MEMORY(MODE,NEW,ISIG,n,it,maxit,itau)
   integer MODE(n),NEW(n),ISIG(n,maxit)
   ISIG(:,it)=NEW;MODE=NEW
   itin=max(1,it-itau+1);itaux=min(it,itau)
   do i=1,n; inn=sum(ISIG(i,itin:it))
         if(2*inn>itaux)MODE(i)=1
         if(2*inn<itaux)MODE(i)=0
   enddo
END
```

a mapping of its states in the previous time steps. Thus, what is here proposed is to maintain the transition function ϕ unaltered, but make it act on the nodes featured by a trait state obtained as a function of their previous states: $\sigma_i^{(T+1)} = \phi(\{s_j^{(T)}\} \in N_j)$, $s_j^{(T)}$ being a state function of the series of states of the node j up to time-step T. We will consider here the most frequent state (or *majority*) memory implementation. Thus, with unlimited trailing memory: $s_i^{(T)} = mode(\sigma_i^{(T)}, \sigma_i^{(T-1)}, \ldots, \sigma_i^{(1)})$, whereas with memory of the last τ state values: $s_i^{(T)} = mode(\sigma_i^{(T)}, \sigma_i^{(T-1)}, \ldots, \sigma_i^{(T)})$, with $T = \max(1, T-\tau+1)$. 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 explains the lower effectiveness of even size τ-memories found in the results presented below.

A FORTRAN code working in double precision has been implemented to perform computations. The core of the code is given in Table 1. The wiring of the n nodes is encapsulated in a $n \times n$ adjacency matrix named ADJ, so that the iadd variable adds the states of the nodes linked to the generic node i. Thus, the parity rule is implemented by means of the `mod 2` operation on the iadd variable, generating `NEW`(i), i.e., $\sigma_i^{(T+1)}$. Previously, the `MEMORY` subroutine implements the itau-majority memory computing. `MEMORY` provides the `MODE`(i) trait states, i.e., $s_i^{(T)}$, on which the iadd is calculated.

2.1. The $\beta = 1$ case

Fig. 2 shows the initial evolving patterns of a simulation of the parity rule on a $\beta = 1$ skeleton (or Gabriel maps) with $N = 10^2$ nodes distributed at random in a unit square. Red squares denote node state values equal one, black squares denote zero state values. The effect of endowing nodes with memory of the *majority* of the last three states is shown at $T = 4$. The encircled node exemplifies the initial effect of memory. Dynamics of perturbation spreading looks unpredictable and quasi-chaotic due to existence of long-range connections.

Fig. 3 shows the evolution of the changing rate (the Hamming distance between two consecutive patterns) in 11 different $\beta = 1$ simulations based on 1000 nodes distributed at random in a unit square. The red curves correspond to the ahistoric simulations, in which case the parity rule exhibits a very high level of changing rate, oscillating around 0.5. Fig. 3 shows also the effect on the changing rate of endowing nodes with memory of the last τ state values (blue lines). The inertial effect of memory tends to reduce the changing rate compared to the ahistoric model, particularly when β is odd. With high memory charges, such as $\tau = 19$ in the lower left panel, the changing rate tends to vary in the long term in the [0.1,0.2] interval, after an initial *almost-oscillatory* behaviour which ceases by $T > \tau + 1 = 20$. With unlimited trailing memory (lower right panel), this *oscillatory* pattern is never truncated, so that a rather unexpected quasi-oscillatory behaviour turns out with full memory. With no exception, the proportion of node states having one given state value (*density*), oscillates near to 0.5 regardless of the model considered.

Fig. 4 shows the evolution of the damage rate, i.e., the relative Hamming distance between patterns resulting from reversing the initial state value of a single node, referred to as damage (or perturbation) spreading. The damage propagates very rapidly without memory (*butterfly effect*), so that by $T = 30$ the red curves already oscillate around 50% of the cells. When memory is introduced to the system, the spread of damage is depleted, albeit the restraining effect of memory is very low if the charge of memory is low ($\tau = 3,4$). With higher memory lengths, e.g., $\tau = 9$, the depletion in the advance of the damage becomes apparent, though by $T = 150$ the damage rate also reaches the 0.5 level in every simulation in Fig. 4. Similar evolution of damage is found with higher limited trailing memories, such as $\tau = 11$ and $\tau = 13$ (not shown in the figure). On the contrary case, unlimited trailing memory appears very effective in the control of damage, as shown in the lower-right panel.

Fig. 4 also shows the relative Hamming distance of the ahistoric patterns to the corresponding historic ones. After a very short transition period, this distance reaches a fairly permanent plateau level, oscillating around 0.5 regardless of the charge of memory endowed.

2.2. A $\beta < 1$ case

Fig. 5 shows a simulation up to $T = 4$ of the parity rule on a $\beta = 0.9$ skeleton based on the same nodes and initial states as in Fig. 2. In correspondence with the lower β value, there are more links