Cyclic Cellular Automata : A Tool for Self-organizing Sleep Scheduling in Sensor Networks

K. J. Kwak
Electrical Engineering Dept.
Columbia University
New York, NY 10027
kjkwak@ee.columbia.edu

Y. M. Baryshnikov
Bell Labs
600 Mountain Ave.
Murray Hill, NJ 07974
ymb@research.bell-labs.com

E. G. Coffman
Electrical Engineering Dept.
Columbia University
New York, NY 10027
egc@ee.columbia.edu

INTRODUCTION: Cyclic Cellular Automata (CCAs) have been found to provide a natural, beguilingly simple, and elegant infrastructure for the design of sensor systems with sleep-wake scheduling to maximize system lifetime. The Greenberg-Hastings model (GHMZ) [3, 2] defined on the integer lattice $\mathbb{Z}^2$ is particularly appropriate and is described as follows. Each grid square of the integer lattice is a cell with a set of $k > 1$ states and a *neighborhood* N; the neighborhoods of interest here are the *von Neumann* neighborhood and the *Moore* neighborhood. The von Neumann neighborhood N_x of cell x consists of just those cells to the north, east, south, and west of x, whereas the Moore neighborhood expands to that 3×3 array of cells with x at its center, i.e., all cells that touch x at a side or vertex. All cells change state synchronously step by step according to a clock cycle and transition function common to all. The local rule for state changes is little more than a counter: The state $\xi_{t+1}(x)$ of cell x at time $t + 1$ is a simple mod k increment: $\xi_{t+1}(x) = \xi_t(x) + 1$ if $\xi_t(x) > 0$. But if $\xi_t(x) = 0$, the state is incremented to 1 if and only if it has at least one neighbor in N_x which is currently in state 1.

In a discrete sensor system, we would consider a finite version of this automaton where we identify cells with sensors, and the 0 state with the wake state. But a better model of reality requires a continuous relaxation of this model, one that we define on $\mathbb{R}^2$. Instead of discrete cells, we take a Poisson pattern contained in a square area with an intensity chosen so as to yield the desired expected number of sensors. With cells (sensors) now replaced by points, we then redefine a neighborhood of a sensor x as a circle of *communication* radius r_c, a given parameter, with the circle centered at x. A sensor y is in the neighborhood of sensor x if and only if y is inside this circle. With this basic model we are done: the transition function is the same as before and yields the continuous GHMR model of synchronous sensor systems . With the number of states, sensor density, and r_c properly chosen, we can start the automaton in a

random state and create thereby a remarkably effective sleep-wake sensor system. It will be a system of "sweep" type, in which waves of wake sensors sweep the entire field periodically in a metastable equilibrium in which each sensor cycles endlessly through the states 0,1, ..., $k - 1$. (See [5] for another sweep system based on a far more complicated mechanism.)

THE DEMO: The demo that we have available will provide simulations illustrating the evolution to equilibrium. In the demo, states are identified with colors where the wake state 0 is always taken to be black. Figure 1 gives snapshots of the evolution of the system at times 0, 10, 15, and 50 with parameter values: unit sensor density, 200 by 200 sensor field (an average of $40,000$ sensors), a communication radius of 1.5, and a number of states $k = 20$ (sensors sleep roughly $\frac{k-1}{k}$ (95%) of the time). Full coverage is virtually guaranteed and intruder-detection delay is at most 8 clock cycles. The scaling of the figure blurs the discrete sensor locations, as can be seen. Recall that it is only the black in the figure that represents wake (0) states. The initial state, often called *primordial soup* in the CCA literature, is a sample from the product measure on the uniform law on $\{0, 1, \ldots, k - 1\}$. To other than CCA mavens, the highly developed structure that truly simplistic local rules will produce from primordial soup can be quite startling. Note that GHMR rather quickly enters states with a large number of sensors resting temporarily in the wake (black) state. Soon, structure in the form cyclically expanding curves of circular shapes appears and reaches an equilibrium where only about 5% of the sensors are awake; the wake sensors form narrow waves sweeping the field and missing no intruders. The points where the circular shapes originate are called *nucleating centers*.

Before further discussion, we digress briefly to mention another distributed sleep-wake technique, again much more complicated than the one here. This is the domatic
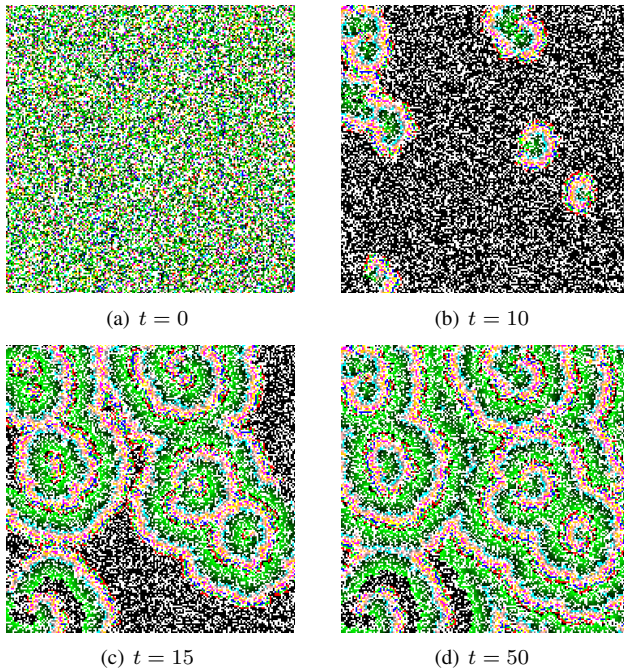
IEEE computer society

(a) $t = 0$ (b) $t = 10$

(c) $t = 15$ (d) $t = 50$

Figure 1. Greenberg-Hastings Model in $\mathbb{R}^2$

Figure 2. Wake Sensor Waves

partition technique [4] that cycles wake states through a sequence of disjoint sensor-field covers. This approach is quite different in that it doesn't provide the same ease of trading off time-to-detection with energy consumption (the fraction of sensors that are awake). For other methods, see the references in [1]. More detailed comparisons are the subject of a paper to appear.

PROPERTIES: The GHMR has the following critically important behavior: *it is scalable, effective against intruders with knowledge of sensor locations, fault tolerant, and seamlessly works around obstacles effectively, and has a high performance/cost ratio.* It is striking that these additional properties essentially 'come for free,' as artifacts of the self-organizing protocol that clearly were not objectives that influenced such a simplistic design. Naturally, however, these properties do vary with parameter values. The first two properties are readily seen. Fault tolerance is similar to self-healing and allows constant fractions of the sensors to malfunction or have communication failures. The near perfect accommodation of obstacles is quite unexpected and will be a feature of the demo, as will fault tolerance.

We leave engineering details to [1], recalling only that sensing intruders, and sensing neighbors broadcasting a state-1 occupancy signal both take place in state 0; broadcasting the state-1 occupancy signal is done in state 1; and the remaining sleep states involve no sensing or communication and simply make the deterministic state change.
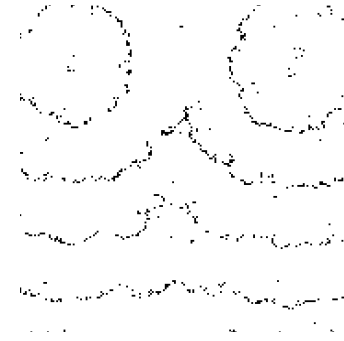
A problem with the GHMR (and CCAs generally) is that for k large, the process may well die out; the CCA is said to *fixate*, meaning that, in our case, all sensors eventually go into the wake state and stay there. In practice this is easily avoided by the placement of *seeds* which consist simply of a cluster of sensors in states $0, 1, \ldots, k - 1$ such that the sensor in state i is within communication range of the sensor in state $(i + 1) \mod k$. These guarantee "liveness" and are illustrated in Figure 2 for the case of two seeds, whose locations are obvious from the figure. The parameters are unit sensor density in a 200 by 200 field, $k = 30$, and $r_c = 1.5$. This figure gives a snapshot suppressing all but the black state, and showing the wake-sensor waves sweeping the sensor field. (The demo will show the actual motion with an appropriately chosen clock cycle.) Note that with seeds planted, the initial state outside of the seeds can be arbitrary.

References

[1] Y. M. Baryshnikov, E. G. Coffman, and K. J. Kwak. High performance sleep-wake sensor systems based on cyclic cellular automata. In *Proceedings, IPSN '08*, 2008.

[2] R. Fisch, J. Gravner, and D. Griffeath. Metastability in the Greenberg-Hastings model. *The Annals of Applied Probability*, 3(4):935–967, 1993.

[3] J. Greenberg and S. Hastings. Spatial patterns for discrete models of diffusion in excitable media. *SIAM Journal of Applied Mathmetics*, 34:515–523, 1978.

[4] S. V. Pemmaraju and I. A. Pirwani. Energy conservation via domatic partitions. In *Proceedings, MobiHoc '06*, pages 143–154, 2006.

[5] P. Skraba, Q. Fang, A. Nguyen, and L. Guibas. Sweeps over wireless sensor networks. In *Proceedings, IPSN '06*, pages 143–151, 2006.