DIVISION OF THE HUMANITIES AND SOCIAL SCIENCES

CALIFORNIA INSTITUTE OF TECHNOLOGY

PASADENA, CALIFORNIA 91125

ON INCENTIVES AND UPDATING IN AGENT BASED MODELS

Scott E Page

SOCIAL SCIENCE WORKING PAPER 1001

February 1997

ON INCENTIVES AND UPDATING IN AGENT BASED MODELS

Scott E Page*

1 Introduction

An economy consists of agents: firms, consumers, governments, etc..., endowed with interests, abilities, and information who interact in real time. These agents behave in response to their incentive structures subject to certain constraints. The importance of incentives and the ability of human actors to adapt their behavior – to change their decisions or decision making rules – distinguishes social systems from physical systems. This paper demonstrates how the interplay between incentives and the timing of updating can alter findings significantly in economic models with neighborhood effects. The analysis considers cellular automata models in which the timing of updating is varied from synchronous, to random asynchronous, to incentive based asynchronous. Significant and interesting differences in the dynamics and steady states are found and explained under each updating rule.

Cellular automata are unfamiliar to many economists. Some background information helps to place the findings in context. Cellular automata (CA) are dynamical systems defined on lattices in which time, states, and spatial relationships are discrete. The state of each cell on the lattice updates at each time step according to a local rule. This local rule depends not only on the current state of the cell but also on the states of cells in a neighborhood around the cell. Cellular automata provide a simple, adaptable framework in which to analyze complex dynamical systems. Even one dimensional cellular automata can exhibit a wide range of dynamic behavior including chaos (Wolfram 1992). The potential impact of cellular automata on economics is not insignificant. Cellular automata allow for models in which economic and social behavior occur through local interactions including models in which economic agents possess only localized knowledge of prices and opportunities. One implication of an agent based modeling perspective is that inefficiences may persist in an economy (Tesfatsion 1994).

CAs can be used to model many physical, biological, and social systems. Recently, there has been work on extending CAs to allow for a continuum of states and continuous

---

*Division of Humanities and Social Science, California Institute of Technology, Pasadena, CA 91125.
Bob Axelrod, Ann Bell, Thad Brown, Colin Camerer, Bernardo Huberman, Mike McBurnett, and students in a computational economics class at UCLA helped to refine the ideas put forth in this paper. Forthcoming in "Computational Economics."

time.[1] In the latter research agenda, the assumption of the discrete synchronous clock has come under scrutiny. Huberman and Glance (1993) re-analyze Nowak and May's (1992) model of the prisoner's dilemma being played on a cellular automata using *random asynchronous updating*. In random asynchronous updating, cells are ordered randomly, and they update their states sequentially in that order. Huberman and Glance find that the interesting patterns exhibited in Nowak and May's work disappeared when the updating of cells became asynchronous. More generally, Chate and Manneville (1991) state that "some of the apparent self–organization in cellular automata is an artifact of the synchronization of clocks," and Bersini and Detours (1994) in analyzing modified versions of the game of life and the immune network model find that random asynchronous updating induces stability rather than long transients.

Random asynchronous updating picks cells at random and updates their states. In a model of a physical system, this may be an appropriate assumption. Yet, this is just one of many possible asynchronous updating schedules. For the purposes of elaboration, consider a one dimensional circular lattice where each cell represents an economic actor and each state represents a strategy. Many types of asynchronous updating can be imagined in this arrangement. The sequencing could be based on geography: the agent located at twelve-o-clock might update first with the updating proceeding either clockwise or counter–clockwise. The sequence might be state based: those agents in state α might update first. Such an assumption would make sense if some states have greater flexibility and require less time to change. The sequence might even be determined by a function of both location and state. Any deterministic or biased sequencing is possible. The modelling challenge is to choose the updating sequence which best applies to the phenomenon being studied.

In economic contexts, the order of sequencing might depend on the incentives to updating. This can formalized as an *incentive based asynchronous updating* rule. The agents (cells) who benefit most from changing their states update first. To determine which agents benefit most, entails extrapolating from the local updating rule and impute a utility function. The order in which agents update can be determined by the agents' relative increases in utility from state changes. The idea that those who benefit most by updating change states earliest accords with reality: all other effects being equal, the first people in line for general admission concert tickets most want to attend, and the people with the most exciting evening plans leave work earliest. Such an assumption is also defensible on theoretical grounds with strategic agents, but the focus here is on CA models in which the automata follow rules rather than act strategically.[2]

To gain insight into incentive based asynchronous updating matters, two models, the *game of life* and the *conformity CA*, are analyzed. The game of life (Berkelamp, Conway, and Guy 1982) has generated widespread interest among professional and amateur

[1] When the cell state space is continuous, the system is called a a bi-coupled lattice. See Bell (1994) for an example of a bi–coupled lattice in an economic context. An advantage of the bi–coupled lattice is that states can update synchronously but at different rates which can be interpreted as measuring different incentives to update.

[2] An interesting research question arises if a CA model is contemplated from a game theoretic perspective. This could be accomplished by letting the per period payoff at a site equal one if the cell is in the correct state and zero if is not.

scientists for decades. Its popularity stems from its capacity to generate cyclic patterns and long transients which can be visualized on a rectangular grid. In the game of life, cells assume one of two states: dead or alive. Whether a cell is alive or dead depends upon the how many of its eight immediate neighbors are alive. If the number of living neighbors is less than two then the cell dies of starvation. If it is more than three, the cell dies of suffocation. If the number of neighbors lies between the thresholds, and the cell is currently alive it remains so. If a cell is dead, it comes to life if exactly three of its neighbors are living.

Modern technology has transported the game from the tiled floors of mathematics departments to computer screens which has increase its popularity. As mentioned, the game of life offers an ideal testbed because with synchronous updating it creates interesting dynamics: patterns, cycles, and long transients as well as extreme sensitivity to initial conditions. The patterns include gliders which float along the grid. With random asynchronous updating, the system quickly stabilizes and exhibits little sensitivity to initial conditions (Bersini and Detours 1994). In this paper, incentive based asynchronous updating is shown to also induce stability. This is not surprising. What is surprising is that it generates greater sensitivity to initial conditions than synchronous updating.

The second model, the conformity CA, generates much simpler dynamics than the game of life. In fact, in an appendix it is shown to converge to a steady state in finite time. Though less well known than the game of life, the conformity CA has greater economic relevance. In the conformity CA, each cell assumes one of two states: zero – one, Mac – IBM, or democrat – republican. Here, neighborhoods contain an odd number of cells.[3] The state of a cell at time period $t+1$ is determined by majority rule over neighboring cells: each cell assumes the state of the majority of its neighbors at time t. Given this interpretation of the states, the conformity CA can be utilized as a model of preference formation in which agents begin to conform to the preferences of their neighbors (Brown, Pfiefer, and McBurnett 1993, Bell 1994). For the conformity CA, incentive based asynchronous updating differs from synchronous and random asynchronous updating in the dynamics it generates, the distribution over steady states it induces, and in the sensitivity to initial conditions it creates.

The remainder of this paper is organized as follows. In the next two sections, the conformity CA is defined, and the updating rules are characterized formally. In section 4, an analysis of the dynamics, steady states, and sensitivity to initial conditions of the conformity CA under the various updating rules is presented. In section 5, the game of life is defined and analyzed, and in section 6, the idea of geographic based updating is applied in two forms to the game of life. The discussion at the end of the paper includes additional comments on the relevance of these results for computational theory and discusses possible future work.

[3] In the conformity CA, whenever referring to a cell's neighbors, the cell itself is included in this set. Recall that in the game of life, the cell is not included.

2 The Conformity Cellular Automata

A cellular automata (CA)is a D-dimensional lattice with a finite state automaton at each cell (Wolfram 1992). In this paper, attention is restricted to square two dimensional CAs of finite width oriented as a torus. To accomplish this, the top of the lattice is connected to the bottom and the right side connected to the left side.

Def'n The *cells* $L = N \times N$

At each cell, the automaton assumes a value from a finite set of states. In the two models considered here, the cells assume only two states.[4]

Def'n The *set of states*, $S = \{0, 1\}$, the state of the automaton at cell (i, j) is denoted by s_{ij}.

The global state of a CA can also be defined.

Def'n The set of *global state* of the cellular automaton, $\mathcal{S} = \{s : s = \{s_{11}, s_{12}, ...s_{NN}\}, s_{ij} \in \{0, 1\}\}$

Each automaton's input consists of the states of the automata in a neighborhood of the cell. For the conformity CA, square neighborhoods of various sizes are considered.

Def'n The *neighborhood of size* α *of cell* (i, j), $N_\alpha(i, j) = \{(\hat{i}, \hat{j}) \ \mid \mid i - \hat{i} \mid \leq k$ and $\mid j - \hat{j} \mid \leq \alpha$

A neighborhood of size one contains nine cells, while neighborhoods of size two contain twenty five cells. In the conformity CA, each automaton wishes to conform its state to the state of a majority of the automata in its neighborhood.

Def'n The *conformity updating rule given neighborhoods of size* α, $C^\alpha : N \times N \times \mathcal{S} \rightarrow \{0, 1\}$ according to the following rules:

$$C^\alpha(i, j, s) = 0 \qquad \text{if} \sum_{(k,l) \in N_\alpha(i,j)} s_{kl} < \frac{(2\alpha + 1)^2}{2}$$

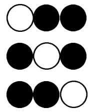
$$C^\alpha(i, j, s) = 1 \qquad \text{if} \sum_{(k,l) \in N_\alpha(i,j)} s_{kl} > \frac{(2\alpha + 1)^2}{2}$$

The figure below shows a neighborhood of size one. Including the cell itself, there

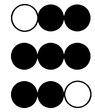
[4] Hereafter, the state of an automaton at a particular cell shall be referred to as the cell's state.

are a total of nine cells. According to the conformity rule, if five or more cells' neighbors assume the state 1 (denoted by a filled in circle), as is the case here, then the cell in the center should also be in state 1. In the figure below, the center cell is in state 0, the incorrect state prior to updating and in the correct state after updating.

Before



After



3 Updating Rules

The *synchronous*, *random asynchronous*, and *incentive based asynchronous updating* rules can now be defined. Let s^t denote the global state of the CA at time t, and let $\mathcal{C}$ denote the space of all possible local updating rules.

Def'n Given $C \in \mathcal{C}$ and $s^0 \in \mathcal{S}$, the *synchronous updating* rule $SY : \mathcal{S} \times \mathcal{C} \rightarrow \mathcal{S}$ as follows:

1. $s_{i,j}^{t+1} = C(i,j,s^t)$ for all (i,j)

Def'n Given $C \in \mathcal{C}$ and $s^0 \in \mathcal{S}$, the *random asynchronous updating* rule $ASY : \mathcal{S} \times \mathcal{C} \rightarrow \mathcal{S}$ as follows:

1. Let $s^* = s^t$
2. For $i = 1$ to $N \times N$ do
   (a) Randomly choose (i,j)
   (b) Let $s_{ij}^* = C(i,j,s^*)$
3. $s_{i,j}^{t+1} = s_{i,j}^*$ for all (i,j)

To define incentive based asynchronous updating requires extrapolating a 'utility from updating' function from the local updating rule. This is the gain in utility that a cell obtains from changing its state. In the conformity CA, a cell wants to conform to be like its surrounding cells. If a majority of the surrounding cells are in a different state than the cell, the cell wishes to change to the other state. A natural assumption is that the more neighboring cells that are in the opposite state, the more the cell wishes to change. The largest utility increases would occur if all of a cell's neighbors were in the opposite state of the cell.[5] This rule can be formalized as follows:

[5] Jonathon Bendor has suggested allowing the order to be determined by the utility differences between cells. In the case of the conformity CA, these rules would have similar characteristics

Def'n Given α the *utility from updating function* for the conformity CA, $U^\alpha : N \times N \times \mathcal{S} \to \mathcal{R}$ as follows:

$$U^\alpha(i,j,s) = \left(\frac{(2\alpha+1)^2}{2} - \sum_{(k,l)\in N_\alpha(i,j)} s_{kl} \right) \cdot (s_{ij} - C^\alpha(i,j,s))$$

Suppose that $\alpha = 1$, so that each cell has nine cells in its neighborhood. The first term in the brackets equals the difference between 4.5 and the number of neighbors that have value one. The second term equals zero if the cell's state would not change during updating, and plus or minus one (depending upon whether it should update to 0 or 1) if the cell's state would change. Thus, if a cell currently has six neighbors in state 1 (as in the figure at the end of the previous section) and is in state 0, then it receives a utility from updating equal to 1.5. Incentive based asynchronous updating can now be defined.

Def'n Given $C \in \mathcal{C}$ and $s^0 \in \mathcal{S}$, the *incentive based asynchronous updating* rule $IBA : \mathcal{S} \times \mathcal{C} \to \mathcal{S}$ as follows:

1. Let $s^* = s^t$
2. For $i = 1$ to $N \times N$ do
   (a) Compute $U^\alpha(i,j,s)$ for all (i,j)
   (b) Choose $(\hat{i},\hat{j}) \in \{\ (i,j)\ :\ U^\alpha(i,j,s) \geq U^\alpha(i',j',s)$ for all $(i',j\prime)\}$
   (c) Let $s^*_{\hat{i}\hat{j}} = C(\hat{i},\hat{j},s^*)$
3. $s^{t+1}_{i,j} = s^*_{i,j}$ for all (i,j)

In incentive based asynchronous updating, a random cell from among those cells which have the highest utility from updating is selected to be updated.

4 Computational Analysis of the Conformity CA

In order to compare the outcomes of the conformity rule under various updating rules, some additional characteristics of CA must be defined. A global state is said to be *stable with respect to the conformity map* if it is a fixed point of the conformity map applied to each cell.

Def'n A global state, $s = \{s_{11}, s_{12}, ...s_{NN}\}$, is *stable with respect to* C^α if $s_{ij} = C^\alpha(i,j,s)$ for all (i,j).

The set of steady states is identical for synchronous, random asynchronous, and incentive based asynchronous updating because the local updating rule is the same for each. This does not imply that given an initial configuration they will lead to the same steady state, nor does it imply that given a distribution over initial configurations that they will generate similar distributions over steady states.

The steady states do vary with α, the neighborhood size. For example, figure one shows a global state which is stable with respect to the conformity map given neighborhoods of size one, and figure two shows a global state which is stable with respect to the conformity map given neighborhoods of size three.

Place figures one and two here

A distinguishing feature between these two global states is that the second consists of larger "chunks" of automatons in identical states. To capture this feature of CAs a measure of *linear disparity* is introduced. Linear disparity equals the average number of times the state changes on average in each row of the cellular automata.

Def'n The *linear disparity*, D, of a global state, $G = \{s_{11}, s_{12}, ...s_{NN}\}$, is defined as follows:

$$D(G) = \frac{1}{N} \sum_{i=1}^{N} (\mid s_{iN} - s_{i1} \mid + \sum_{j=1}^{N-1} \mid s_{ij} - s_{i(j+1)} \mid)$$

The extent to which the various updating rules alter the sensitivity to initial conditions is also of interest. There are several ways to measure sensitivity to initial conditions. The approach followed by Browne, Pfiefer, and McBurnett (1992) is to generate a time series of the average state of the cellular automata and to compute the Lyapunov exponent. A statistical test of the Lyapunov exponent then allows the determination of the likelihood that the underlying process is chaotic. A simpler, equally compelling measure of the sensitivity to initial conditions is to ask how much the equilibrium global state changes when a small change is made in the initial global state. Therefore, the measure of sensitivity to initial conditions employed is constructed as follows: change the state of one, two, three, or four randomly chosen cells in the initial global state and then count the number of cells whose states differ in the resulting equilibrium global state. If only five of ten cells have new states, then the system is not very sensitive to initial conditions. If several hundred cells have different states, then the system is sensitive to initial conditions. Exactly what constitutes *extreme sensitivity to initial conditions* is not addressed here.

Before describing the computational experiments performed using the conformity CA, a comment is in order about its stability. The conformity CA settles down to an equilibrium and does so quickly. In an appendix at the end of the paper, a proof of the convergence of the conformity CA is included.

4.1 Observations

For the computational experiments conducted here, the initial configurations of cells were randomly distributed with cells equally likely to be in state 1 or state 0. The conformity map is applied to neighborhoods of size one, two and three. Table 1 shows linearity disparity measures from fifty trials on a fifty by fifty cellular automata for the non–incentive based updating rules.

Table 1

Linear Disparity

Nbhd size	*Sync*	*(s.d.)*	*Async*	*(s.d.)*	*Incen*	*(s.d.)*
1	13.143	(0.55)	11.510	(0.59)	10.342	(0.68)
2	7.265	(0.39)	7.102	(0.40)	3.673	(0.732)
3	4.735	(0.29)	3.918	(0.31)	0.00	(0.00)

Several observations about the linear disparity under the various updating rules can be made.

Observation 1: *Linear disparity decreases as neighborhood size increases for all three types of updating rules.*

This observation has an intuitive explanation. As the neighborhood size increases the size of the "blocks" of conformity should grow. Eventually, when the entire array forms one neighborhood, then the only equilibrium is when all cells are in the same state.

Observation 2: *Linear disparity varies only slightly between synchronous and random asynchronous updating. Incentive based asynchronous updating creates less linear disparity.*

Though the dynamics differ appreciably, there is little difference between the distribution of equilibrium states for synchronous and random asynchronous updating of cells. This finding suggests further inquiry into the conditions under which the move from synchronous to random asynchronous updating effects the distribution over end states. In the conformity, CA the choice of updating rule, at least between these two, does not seem to matter.

On average, incentive based asynchronous updating creates larger "blocks" of identical states as measured by linear disparity. An explanation for this phenomenon is that the first cells to update tend to complete small blocks. These blocks then grow into larger blocks. This difference in linear disparity may have implications for empirical studies of neighborhood effects as discussed later in this paper. The following observation is implied by the linear disparity findings.

Observation 3: *Incentive based updating exhibits greater variance in average state value.*

The various updating rules also effect sensitivity to initial conditions.

Table 2

Sensitivity to Initial Conditions: Synchronous

\# cells changed

Nbhd size	*One*	*(s.d.)*	*Two*	*(s.d)*	*Three*	*(s.d)*	*Four*	*(s.d)*
1	2.12	(0.49)	7.08	(0.89)	12.38	(1.22)	18.68	(1.48)
2	7.50	(1.38)	14.30	(1.74)	26.12	(2.70)	28.52	(2.61)
3	16.04	(3.43)	21.58	(3.19)	36.08	(4.48)	28.38	(3.50)

Table 3

Sensitivity to Initial Conditions: Random Asynchronous

\# cells changed

Nbhd size	*One*	*(s.d.)*	*Two*	*(s.d)*	*Three*	*(s.d)*	*Four*	*(s.d)*
1	1.84	(0.60)	3.98	(1.02)	8.44	(1.11)	10.52	(1.28)
2	8.82	(2.97)	14.48	(3.23)	23.92	(4.24)	27.22	(4.64)
3	14.68	(5.17)	16.04	(5.13)	34.50	(6.72)	40.70	(7.54)

Observation 4: *Sensitivity to initial conditions increases with neighborhood size for synchronous and random asynchronous updating.*

As neighborhood size increases there are two counteracting effects. First, the probability that a cell switches its state as a result of one cell in its neighborhood switching decreases as there are more cells. This would decrease the sensitivity to initial conditions. Second, the number of cells who are neighbors of a given cell increases, which increases the sensitivity to initial conditions. This second effect dominates.

Observation 5: *Random asynchronous updating exhibits similar sensitivity to initial conditions as synchronous updating.*

This suggests that random asynchronous updating does not generate greater or less stability. Note that in the game of life, random asynchronous updating generates much greater stability. In the data shown, the updating of cells occurred in the same random order in each experiment. If in addition to changing a bit, the order in which the cells update their states also changed, then there would be a massive change in the global state. Table 4 below shows the sensitivity to initial conditions given incentive based asynchronous updating.

Table 4

Sensitivity to Initial Conditions: Incentive based

cells changed

Nbhd size	*One*	*(s.d.)*	*Two*	*(s.d)*	*Three*	*(s.d)*	*Four*	*(s.d)*
1	5.82	(1.42)	11.38	(2.06)	12.54	(2.36)	16.76	(2.45)
2	32.94	(18.83)	37.46	(14.36)	77.10	(18.77)	72.04	(17.60)

Observation 6: *Incentive based asynchronous updating exhibits significantly greater sensitivity to to initial conditions than synchronous updating and random asynchronous updating.*

The differences in linear disparity may be a partial explanation for the increased sensitivity to initial conditions. In many cases, changing one cell initially change a 'chunk' of cells in the final distribution. The linear disparity measure shows that the chunks are larger under the incentive based updating rule, which might contribute to the increased sensitivity to initial conditions.

5 The Game of Life

As has been mentioned, with synchronous updating, the game of life creates complex dynamics and extreme sensitivity to initial conditions, but with random asynchronous updating these interesting dynamics disappear: the CA quickly settles into a stable global state (Bersini and Detours 1994). Here incentive based asynchronous updating also generates stability, though not nearly as quickly. Surprisingly, this stability does not imply less sensitivity to initial conditions. According to the measures used, incentive based asynchronous updating leads to even greater sensitivity to initial conditions than synchronous updating. Some insights into why this occurs are provided later in this section.

This section begins with a description of the game of life and then motivates two utility from updating functions: one biased towards population growth and another unbiased. Differences in these utility from updating function generate predictable changes in the dynamics of the CA. Finally, another class of updating sequencing, *geographic based asynchronous updating*, is considered. Examples from two updating rules within this class are discussed.

5.1 Description

The story behind the game of life is that if fewer than two of a cell's neighbors are live, the cell dies from starvation, and if more than three of its neighbors are alive, the cell dies from suffocation. In order for a cell to be come to life exactly three of the neighboring cells must be alive.

Def'n The *game of life*, $C^L : N \times N \times \mathcal{S}times\{0,1\} \to \{0,1\}$ according to the following rules:

$$C^L(i,j,s,1) = 0 \qquad \text{if} \sum_{(k,l)\in N_1(i,j)} s_{kl} \notin \{2,3\}$$

$$C^L(i,j,s,1) = 1 \qquad \text{if} \sum_{(k,l)\in N_1(i,j)} s_{kl} \in \{2,3\}$$

$$C^L(i,j,s,0) = 0 \qquad \text{if} \sum_{(k,l)\in N_1(i,j)} s_{kl} \neq 3$$

$$C^L(i,j,s,0) = 1 \qquad \text{if} \sum_{(k,l)\in N_1(i,j)} s_{kl} = 3$$

Rather than just having one threshold like the conformity CA, the game of life has two thresholds which differ depending upon the state of the cell. This more complicated updating rule permits greater flexibility in formulating a utility from updating function. One approach is to bias the updating towards cells that were poised for birth – cells which are dead but which have exactly three live neighbors. Other approaches are to bias the updating toward death by making birth the lowest priority, or to balance the priorities for births and deaths. Biasing the updating function so that cells which die update first leads to uninteresting dynamics. Such a rule usually generates entirely dead CAs; thus, the analysis here is restricted to the other two approaches.

By constructing two different utility from updating functions, their features can be compared. Not surprisingly, dynamics, steady states, and sensitivity to initial conditions depend upon the updating function. Thus, for some classes of CA, the utility from updating function used in incentive based updating also effects characteristics of outcomes. An implication of this unfortunate fact is that modellers should consider multiple utility from updating functions and either report invariant outcomes or motivate their choice of particular rule to the exclusion of others.

The two utility from updating functions rely on the fact that there are only eight situations in which a cell has positive utility from updating: one where the cell is about to come to life, two where the cell starves to death, and five where the cell suffocates. The first utility from updating function is biased in favor of dead cells which would become alive if updating were to occur. For convenience, it is called the *growth biased* utility from updating function.

Def'n The *growth biased utility from updating function* for the game of life, U^G

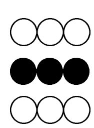
Utility From Updating		
state	*# Neighs alive*	U^G
0	3	6
1	8	5
1	7	4
1	0	4
1	6	3
1	1	3
1	5	2
1	4	1

The other utility from updating function kills off live cells with seven or eight live neighbors and with no live neighbors, before bringing dead cells with three neighbors to life. It is called the *unbiased* utility from updating function.

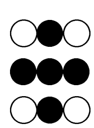
Def'n The *unbiased utility from updating function* for the game of life, U^U

Utility From Updating		
state	*# Neighs alive*	U^U
1	8	6
1	7	5
1	0	5
0	3	4
1	6	3
1	1	3
1	5	2
1	4	1

These two utility from updating functions generate quite different dynamics. To see why, begin with a global state in which three live cells are arranged in a horizontal row on a twenty cell by twenty cell CA. Denote live cells by discs and dead cells by circles.



Under the growth based updating rule, either of two cells could "come to life." These two cells have been filled in the figure below:



When either of these cells comes to life, the two cells on either side are now poised to come to life as well. Simple calculations with pencil and paper show that this initial configuration will lead to a huge region of entirely live cells. Under the unbiased updating rule, the growth of live cells will be mitigated by the fact that whenever a cell is surrounded by seven or eight live cells (which will happen quickly given this configuration), it will die. An analysis of incentive based asynchronous updating can now be undertaken with each of these two utility from updating functions.

5.2 Computational Analysis

As before, the CAs begin with a random distribution of live and dead cells with each occurring with equal probability. In agreement with previous studies, here synchronous updating leads to complex, often unstable dynamics. As linear disparity was used to compare steady states, it is omitted in the analysis. Sensitivity to initial conditions becomes the primary focus. Table 6 below shows the sensitivity to initial conditions for a twenty by twenty CA which was iterated for twenty cycles.

Table 6

Sensitivity to Initial Conditions: Game of Life

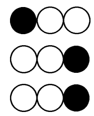
cells changed

Updating Rule	*One*	*(s.d.)*	*Two*	*(s.d)*	*Three*	*(s.d)*	*Four*	*(s.d)*
Synch.	85.22	(6.50)	99.96	(4.51)	110.92	(3.25)	109.56	(3.28)
Ran. Asy.	12.74	(1.53)	18.94	(1.92)	22.42	(1.83)	23.44	(1.76)
Growth I.B.	107.44	(5.02)	104.98	(4.75)	108.66	(5.10)	114.76	(4.62)
Unbias I.B.	102.92	(4.86)	107.32	(5.80)	115.44	(5.12)	113.92	(5.05)

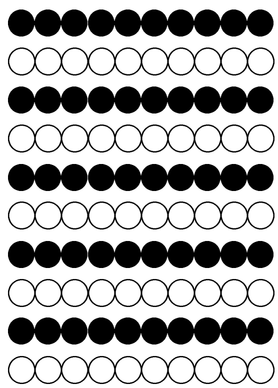
With synchronous updating, if one cell changes its state, then on average a little more than one-fifth of the cells are in opposite states after twenty cycles. This is extreme sensitivity to initial conditions. In contrast, for random asynchronous updating, if one cell's state is changed, then on average less then twenty cells are in opposite states after twenty cycles. Once again, random asynchronicity leads to less sensitivity to initial conditions and to stability.

With incentive base asynchronous updating, although obtaining stability – usually after three or four passes through the CA, the system has settled into an equilibrium – the sensitivity to initial conditions is greater than under synchronous updating. Moreover, this is true for each of the two utility from updating rules. Both observations merit further attention. The first finding suggests that all forms of asynchronous updating do not necessarily lead to less sensitivity to initial conditions. In other words, interesting dynamics exists. The second finding is both surprising and encouraging – surprising in that the growth base rule did not lead to much greater sensitivity to initial conditions, and encouraging in that the level of sensitivity is roughly invariant to the choice of updating rule.

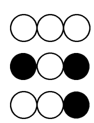
An example from a ten by ten CA using the growth biased updating rule demonstrates how the increased sensitivity to initial conditions can occur. Suppose that the initial configuration of live cells looks as follows:



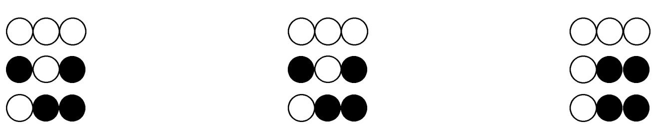
The cell in the center is the unique cell which can come to life. After it comes to life, other cells come to life and the equilibrium that obtained consists of alternating rows of live cells as shown below:



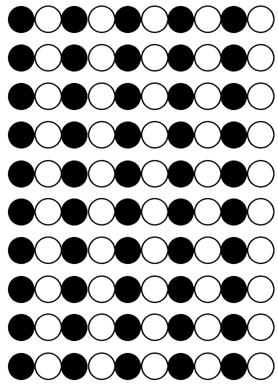
It is easy to see that this configuration is stable with respect to the rules of the game of life. Changing the state of two cells in the initial configuration forms the following configuration:



In this configuration there are two cells which may come to life. Suppose that the cell in the center of the bottom row comes to life. The next three states of the CA are as follows:



If the center cell comes to life first, then the population of live cells will grow quickly. In one experiment, the CA settled into the following stable configuration:



This configuration disagrees with the earlier configuration of alternating rows of live and dead cells on exactly half of the cells. Similar sorts of effects occur when starting from random initial configurations.

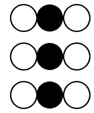
6 Geographic Updating

There are a multitude of rules which the cells in a CA could use to determine the updating order. So far, this paper has focused on random and incentive based asynchronous updating. In some contexts, geographic based asynchronous updating, in which the order of updating is determined by location, may be appropriate.[6] Economic examples include cases in which the cells represent agents and the agents are arranged according to some criterion, such as seniority or expertise as is often the case in hierarchies.

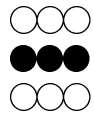
In geographic updating, cells update asynchronously according to an order which is determined by their ranking. In this example, cells are ranked lexicographically first by row and then by column. The cell $(1,1)$ has the highest rank followed by $(1,2)$ and so on through cell (n,n). Once a ranking has been established, there are at least two ways to proceed with with updating. In each, the cells are ranked ordinally. The first method, *iterated geographic updating*, updates the highest ranked cell, then the next highest, and so on until all cells have been updated. At that point, it begins again with the highest ranked cell. Notice that this is different from an updating rule which the next cell updated is the highest ranked cell in the 'wrong' state. This second method shall be called *strict geographic updating*. In this geographic ranking system, this method calls for the cell $(1,1)$ to be the next cell updated whenever $(1,1)$ is in the wrong state.

To better understand the differences between these two types of geographic updating rules, consider a simple initial configuration and apply the game of life updating rule. In this and other representations of the CA, a three cell by three cell section is used to represent a much larger CA

[6] One dimensional CAs where the updating is geographic often are called filter automata.

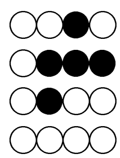


Consider first, the standard synchronous updating. This initial configuration creates a blinker between two global states. In the next iteration, the CA looks as follows:

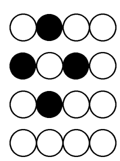


A straightforward computation demonstrates that in the next iteration, the CA returns to the initial global state. Therefore, the dynamic behavior is a row of live cells of length three which alternates between a horizontal and a vertical orientation.

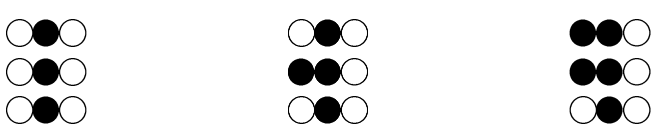
With iterated geographic updating, the dynamics differ. Again, begin with three live cells arranged vertically. The cells are updated sequentially moving first across the CA and then down the rows. After one pass through the CA, a global state with five live cells is reached.

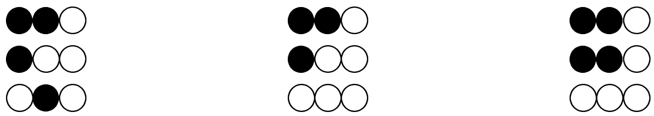


After a second pass through the CA, the CA is in a stable global state.



Taking the same initial global state and applying strict geographic updating generates a different sequence of global states. Because there are no complete passes through the CA with strict geographic updating, the CA is shown after each cell updating:





After five cell updates, the CA is in a stable global state, which is different from that reached by iterated geographical updating.

7 Conclusions

This paper contains experiments with two classes of CA and demonstrates differences between synchronous, random asynchronous and incentive based asynchronous updating. For the conformity CA, incentive based asynchronous updating appears to bias the distribution towards global steady states with less linear disparity. It also leads to greater sensitivity to initial conditions. For the game of life, incentive based asynchronous updating does not induce the stability found with random asynchronous updating. In the game of life, incentive based asynchronous updating also creates sensitivity to initial conditions which disappears with random asynchronous updating. Though there is no formal proof for why this occurs, one guess is that a small change in the initial conditions changes the order in which cells are updated, which starts the system on an entirely different path. Under incentive based asynchronous updating, the ordering is more deterministic than under random asynchronous updating, therefore the divergence of paths might be greater. But, this is all conjecture based upon watching systems evolve on the computer screen.

While these findings do not imply that for all updating rules that incentive based asynchronous updating will alter relevant characteristics of the dynamical system, they suggest that great care should be taken when defining the order in which updating takes place. More generally, these findings demonstrate the need to be careful in drawing inferences about social phenomena from such simple models. In light of the analysis presented here, others may wish to revisit existing CA models to test their susceptibility to changing the ording of cell updating. Among the models which might merit a fresh look is Schelling's (1978) famous tipping model of racial segregation. When considering neighborhood effects in economic models (Durlauf 1995), whether they are endogenous or exogenous, the role of timing of updating which might have seemed too subtle to be of much importance should now have a more prominent role. To give just one example, in trying to estimate the size of influence neighborhoods in a world accurately modelled by the conformity CA. Incentive based updating leads to much greater linearity disparity for a given neighborhood size. This increase in linear disparity could be mistakenly interpreted as a larger influence neighborhood when in fact it is the result of incentive based asynchronous updating.

This paper also suggests future experiments with asynchronous updating in which the order is biased towards those cells with the higher utility to updating. Intuitively, the dynamics and distributions over stable states should lie in between those generated by random asynchronous updating and incentive based updating. Whether in fact there is a phase transition for some levels of updating would be of particular interest.

Appendix: Convergence of the Conformity CA

The proof that the conformity CA converges to a steady state rests on three facts: aggregate conformity increases with each cell updating; aggregate conformity is bounded from above; and the size of each increase is bounded from below. Together these imply that the process of cells changing their states must stop in finite time.[7] To reduce the amount of notation, two changes are made in the formulation of the conformity CA. First, the state space is renormalized so that a cells now take either the state minus one or plus one. Second, cells shall be referred to by a single variable i indexed up to $m = n \cdot n$, rather than with the coordinate pair (i, j). Let cell i's state be denoted by $t_i \in \{-1, 1\}$, and let the sum of the states of the cells in i's neighborhood of size α be denoted by T_i^α. Given this notation, the total conformity of the CA's state relative to the neighborhood size can be characterized. Call it the $\alpha-$ *conformity.*

Def'n The α–*conformity*,

$$, _\alpha = \sum_{i=1}^{m} t_i \cdot T_i^\alpha$$

The maximal α–conformity equal $m \cdot (2\alpha + 1)^2$ which occurs when all m cells are in the same state. Claim 1 states that when a cell update it increases α–conformity by an amount bounded away from zero.

Claim 1 *When cell i updates its state according to the conformity updating rule given neighborhoods of size α, the α–conformity increases by at least eight for any $\alpha \geq 1$.*

pf: Without loss of generality assume that cell i updates its state from -1 to 1, which implies that $T_i^\alpha \geq 1$. Hereafter, abbreviate T_i^α as T_i. The α–conformity prior to updating equals:

$$, _\alpha^0 = \sum_{j \notin N_{i\alpha}} t_j \cdot T_j \; + \; \sum_{j \in N_{i\alpha}} t_j \cdot T_j \; + \; t_i \cdot T_i$$

where $N_{i\alpha}^-$ consists of cell i's neighborhood of size α minus cell i itself. After updating cell i the α–conformity equals

$$, _\alpha^1 = , _\alpha^0 \; + \; 2 \cdot \sum_{j \in N_{i\alpha}^-} t_j \; + \; 2T_i \; + \; 2$$

The difference in α–conformity, which is denoted by Δ, can be written as:

$$\Delta = 2 \cdot \sum_{j \in N_{i\alpha}^-} t_j + 2T_i \; + \; 2$$

[7]The proof presented was derived independently and is simplified as much as possible. An alternative proof is contained in Bersini and Detours (1994)

By construction, $T_i - t_i = \sum_{j \in N_{i\alpha}^-} t_j$, and by assumption $t_i = -1$; therefore, we can rewrite Δ as:

$$\Delta = 2T_i + 2 \; + \; 2T_i + 2$$

By assumption $T_i \geq 1$, which implies $\Delta \geq 8$, which completes the proof.

Given that α–conformity is bounded from above, Claim 1 implies that the process of cell updating stops. This claim can be used to prove other characteristics of the conformity CA. For example, a simple calculation shows the maximal number of cells which must be updated before a stable global state is attained. Finally, note that incentive based asynchronous updating increases the α–conformity by the maximal amount given the configuration at each time step.

Bibliography

Bell, A. 1994. "Dynamically Interdependent Preferences in a General Equilibrium Environment." Working paper, University of Wisconsin–Madison.

Berkelamp, E., J.H. Conway, and R. Guy. 1982. "Winning Ways For Your Mathematical Plays." Academic Press. New York.

Bersini, H. and V. Detours. 1994. "Asynchrony Induces Stability in Cellular Automata Based Models." in *Artificial Life IV*. MIT Press, Cambridge.

Brown, T., P. Pfiefer, and M. McBurnett. 1992. "Political Life on a Lattice: Toward an Elementary Theory of Nonlinear Politics." unpublished manuscript, University of Missouri.

Buvel, R.L. and T.E. Ingerson. 1984. "Structure in Asynchronous Cellular Automata." *Physica D*. 1:59-68.

Chate, H. and P. Maneville. 1991. "Evidence of Collective Behavir in Cellular Automata." *EuroPhys Lets*. 14:409-413.

Durlauf, S. 1995. "Some Economics of Sorting" mimeo Santa Fe Institute.

Gutowitz, H. 1991. "Transients, Cylces, and Complexity in Cellular Automata." *Physical Review A*. 44(12): 7881-7884.

Huberman, B. A. and Glance, N. S. 1993. "Evolutionary Games and Computer Simulations" *Proceedings of the National Academy of Sciences*, Vol 90, pp 7716-7718.

Nowak, M.A. and May, R. M. 1992. *Nature* (London) Vol 359, pp 826-829.

Schelling, T. 1978. *Micromotives and Macrobehavior*, Norton, New York.

Tesfatsion, L. 1993. "How to Get A-Life" unpublished manuscript. Iowa State University.