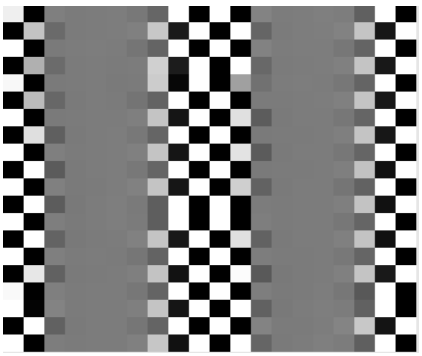
Moving Mass

A Nonlinear Dynamics Project

by Jeff Hutchinson (hutchinson@student.physics.ucdavis.edu)
& Matt Fletcher (fletcher@student.physics.ucdavis.edu)
UC Davis Physics Dept.
Spring 2008

Abstract:
Cellular automata systems with float values in [0.0,1.0], various neighborhoods and flows between cells are investigated. Flows range from a simple linear exchange on a von Neumann neighborhood to sinusoidally driven flow on a larger neighborhood. Accretion and repulsion were characterized, and exhibited ~exponential decay in their flow over time. An interesting checkerboard or band pattern was observed for the sinusoidally driven case. Original (we believe) methods of analysis are discussed.



Intro / Background:

Our motivation for the dynamical system under consideration was cluster formation in the early universe. The idea was to model the evolution of some initial mass distribution into clusters by the action of attraction between masses. This first idea established the initial dynamical system and the language in which we talked about the system thereafter--for instance, the quantity on which dynamical flow occurs we call the "mass". However, we quickly diverged from this idea into more interesting (in terms of results) dynamical flows. It should be noted that our final investigation was into a system with very minimal relation to this original idea, and that it became more just an abstract set of dynamics which we attempted to understand as fully as possible.

Dynamical System:

The dynamical flows modeled were all run on a cellular automata (typically ten cells by ten cells in size), with varying neighborhoods and flows. The mass was a float in the range [0.0,1.0] and flowed only between cells which shared a border, even when the neighborhood used to compute the flow was larger than these four cells. This model was created through instances of two original classes, coded in python: Wrapped2DArray, and Automata. Wrapped2DArray was a numpy array with an attached list of movable pointers to cells in the array and various methods for moving through the array safely. (See code comments in Wrapped2DArray.py for more details.) The Automata class provided methods for the seeding, clearing, and iteration of a Wrapped2DArray according to a specified flow. (See code comments in Automata.py for more details, and Basic_Flow.py for the implementation of the flows discussed below.)

The first system modeled ("model 1" hereafter) is an accretion model with a von Neumann neighborhood (the four cells sharing sides with the cell for which flow is being calculated). The exact equation is:

$$\text{flow_amount} = c * F1 * (T1-F1) / max$$

where c is a parameter, max is a cap imposed on the amount of mass present in a given cell, $T1$ is the amount in the current cell, and $F1$ is the amount in the neighboring cell (cell 1) from which the flow is coming.

The second system modeled ("model 2" hereafter) was just the reverse of model 1: literally just multiplying the flow_amount by -1. This models repulsive mass.

The third system modeled ("model 3" hereafter) was the same as model 1 except that is was modulated by a position and time dependent sine wave. (Note that this position dependence was in the horizontal direction only.) The exact equation is:

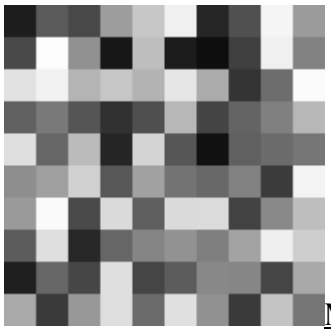
$$\text{flow_amount} = \sin(2*\pi*[k*x-w*t]) * c * F * (T-F) / max$$

The fourth and final system modeled ("model 4" hereafter) was the same as model 3 except for the definitions of T and F. Rather than being the value of mass in a specific cell, both T and F are now averages over a larger neighborhood, weighted by a $1/r^2$ dependence, as drawn below. For instance, $T1$ carries four times the weight of $T4$.

T3	T2	F2	F3
T4	T1	F1	F4
T5	T6	F6	F5

All of these models are mass preserving to at least floating point accuracy over the short term, and no notable change in the mass of the system was observed over more than 500 iterations.

The cellular automata were seeded with a uniform random distribution (reproducible if desired) over the range (0.0, max). (For all cases max was set to 1.0, and thus will be disregarded for the remainder of this report. One should not forget the important effect of a mass cap on behavior though.) An example of such an initial seed, with mass in the range above represented by a gray scale, is shown below:



Methods:

The full set of parameters among these models which can be varied in the search for interesting behavior and understanding are c, k, and w. The first question to arise is how to quantify behavior of a cellular automata which has 100 or more elements. Element by element comparison is meaningless for most questions, and so the focus of this research was on measures of the state (or change in state) of the array as a whole as a function of time. Measures that we looked at included:

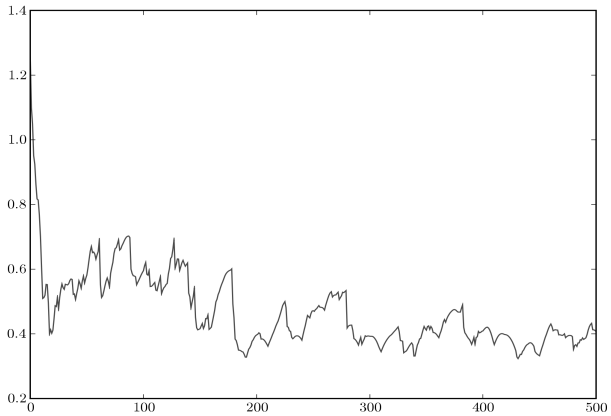
-the number of cells above and below given mass values after a set number of iterations, and how this varied across initial conditions and parameter settings,

-the number, area, and perimeter of "blobs" in the array. A blob is a group of cells in contact with each other which all have a mass value at or above a specified cutoff,

-a display of the state of the cellular automata as it iterates,

-a movie (playback after the fact) of the state of the automata as it iterates,

-the "settle" graph shows how much mass is moving between cells in the array from one iteration to the next. Consider the example below:



-an exponential fit to the settle graph,

-the average settle graph (average over many different settle graphs with the same parameter settings but different initial seed) See examples further below.

Of these, the analysis methods most commonly used were the simple settle graph and visual inspection of the state display as the automata iterated.

Results:

Models #1 and #2:

Models #1 and #2 consistently fell into static patterns. Model #1 accreted into small clusters of full cells, with no partial cells adjacent to each other. Model #2 quickly smoothed to a homogeneous mixture.

Both models exhibited exponential decay in their settle time graphs. Model #1 fit more poorly to an exponential curve per iteration (Fig #1), but approached a smooth exponential when averaged over initial conditions (Fig. #2). Note that fitting underestimates the long term flow. It is unknown whether this is due to errors in the fitting parameters, or deviations from an exponential model.

The functional dependence of the exponential parameters on the flow parameter, c, was investigated and summarized in Fig. #3. The values shown cover the entire range used to minimize overflow (producing cells over 1 or under 0) for Model #1. Linear dependence is evident.

Models #3 and #4:

Model # 3 fell into two typical patterns by ~500 iterations depending on the parameters. Investigations focused on variation in frequency, w, with fixed c (= 0.3) and k (= 0.1). For $w \approx 0.05$ and greater, with c and k as given above, simulations produced a smoothly varying field with some structure remaining at 500 iterations (Fig. 4). When run for 2000 iterations, this overall appearance remained but with a more smoothed appearance, suggesting that the structure may be fundamentally transient but persistent

beyond the timescales investigated.

For w ≈ 0.05 and less for these parameters, Model #3 produces a striped pattern, transverse to the direction of the wave propagation. The stripes alternate smooth and checkerboard patterns, with occasional defects in the checkerboard persistent to 2000 iterations (Fig. #5). At w ≈ 0.05, empty and full cells appear in small numbers and clustered together, eventually expanding into checkerboard stripes as w decreases.

Model #4 exhibited similar high frequency behavior with a shift in the transition point to w ≈ 0.01. Being more computationally intensive, model #4 was only investigated to 500 iterations. At low frequencies, however, the pattern falls to stripes of horizontal bands in the direction of propagation (Fig. #6) rather than checkerboards.

The settle time graphs of model #4 are shown in Fig. #7. There appear two main trends in the high frequency region (w ≈ 0.01 and greater). First, an exponential decay with a significantly longer half life than in models #1 and #2. Compare with the w =1 case, which devolves into a mixture of the first two models since it has no time dependence with a period of one iteration. The half lives for w ≈ 0.75 - 0.04 are on the order of 200 iterations. For w less than ≈0.04 the decay is lost in the structure, approaching a linear decay instead. Exponential fitting of these curves was not performed due to time limitations, leaving the exact functional form of the curve and its relationship to w unknown. To the quality of visual inspection, there appears no dependence on w in the decay.

Second, a near periodic oscillation modulating the decay grows increasingly structured as the frequency decreases. Besides the structure within each semi-periodic region, smoothly varying envelops on either the overall modulation or components of the internal structure appear, particularly within w ≈ 0.3 – 0.06.

In the low frequency region (w ≈ 0.01 and less), the periodicity disappears and the internal structure grows convoluted and apparently random. This region corresponds to the emergence of stripes in the structure. Note that the lowest frequency shown (w = 0.0025) was investigated for only slightly over one period. Therefore, it is unknown whether the long term behavior appears semi-periodic as well. The case of w = 0.01 was investigated to 2000 iterations (20 periods) with little or no periodic structure found.

Conclusion:

Models #1 &#2 showed the expected behavior and show the correct running of the code running the cellular automata. Uninteresting beyond this function.

Both Model #3 and #4 appear to fall into asymptotic states minimizing flow. There are two states, depending on the value of w. High frequencies appeared to limit to uniformity, while low frequencies limited to vertical stripes of uniformly structured and uniformly smooth regions. The structures maximize the flow in any given comparison (alternating adjacent full/empty cells for nearest neighbor, hence checkerboard, while alternating full/empty bands for the larger neighborhood).

The w dependence in Model #4 is surprisingly complex and requires more investigation. Current findings suggest a possible period doubling cascade leading to chaos around w ≈0.01 in Model #4.

In order to investigate this claim, sensitivity to initial conditions and aperiodicity over longer timescales need to be checked. Further investigation could also involve looking for attractors by plotting the state space and locating the boundary between the asymptotic states in (w,k,c) space.

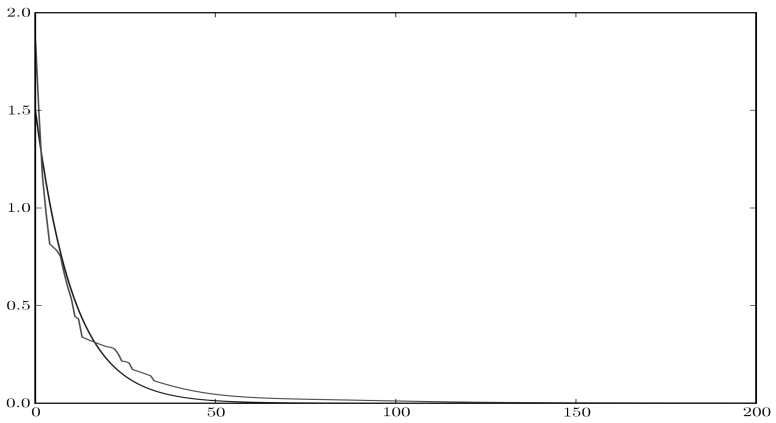


Figure 1. Model #1 settle graph (flow vs. time)

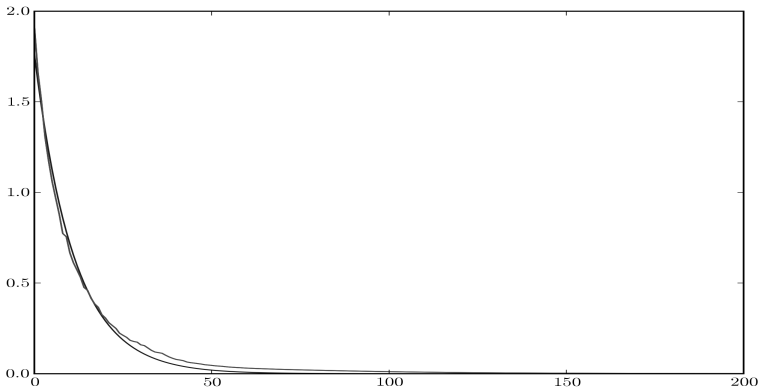


Figure 2. Same as figure 1 except averaged over many ICs

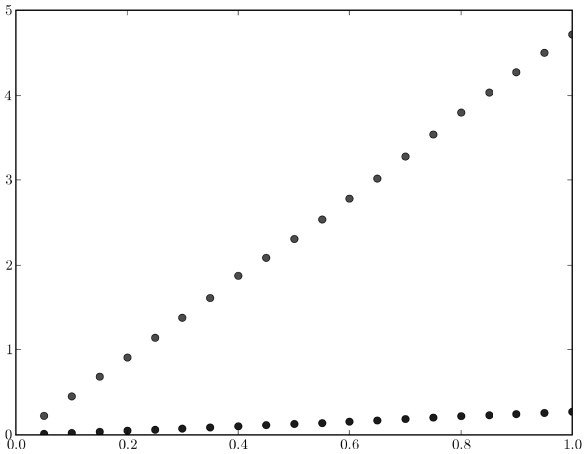


Figure 3. Exponential fit parameter dependence on c (parameters vs. c)
Red = Initial value. Blue = Time constant.

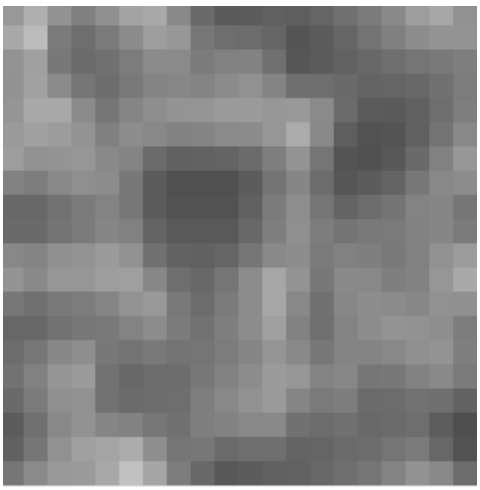


Figure 4. High frequency (w = 0.1) state after 500 iterations in Model #3.

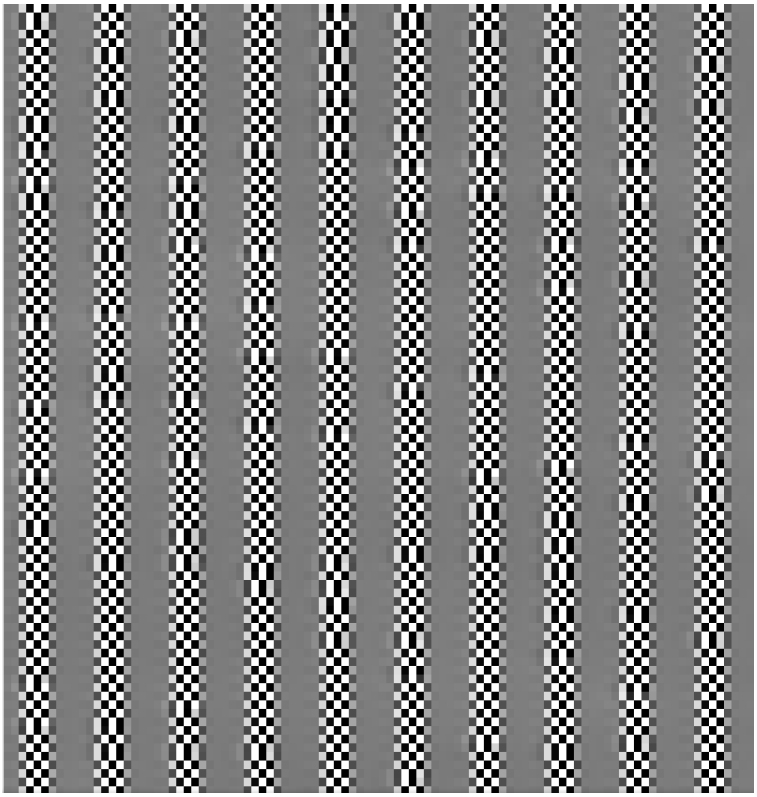


Figure 5. Low frequency (w = 0.01) behavior of Model #3 at 1800 iterations.

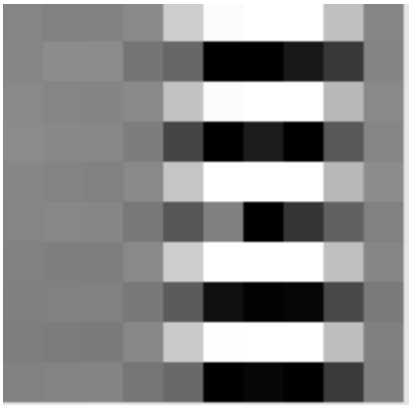
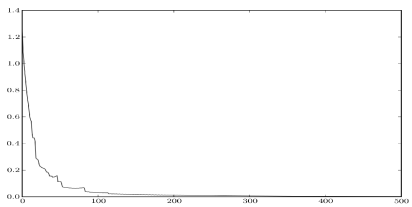


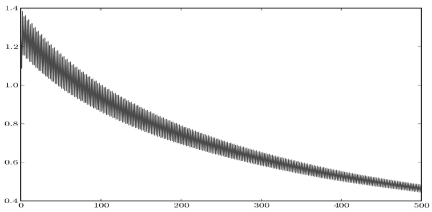
Figure 6. Low frequencies behavior of Model #4 at 500 iterations.

Figure 7. Various settle time graphs for Model #4 (k =0.1, c =0.3)

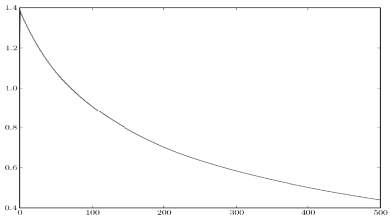
w = 1.0



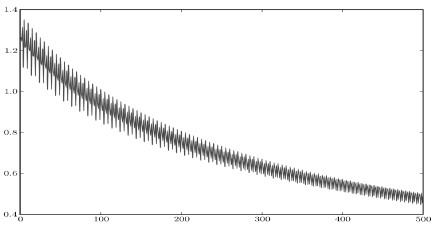
w = 0.75



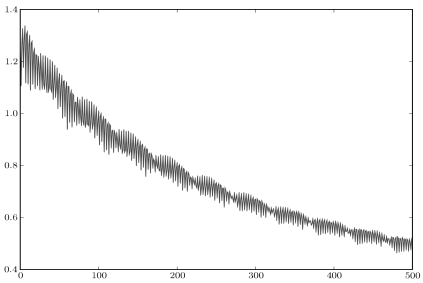
w = 0.5



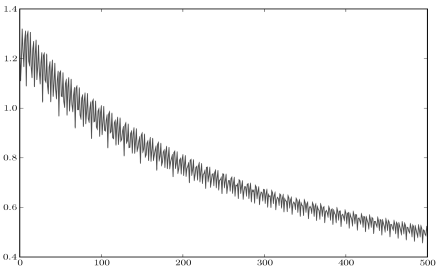
w = 0.3



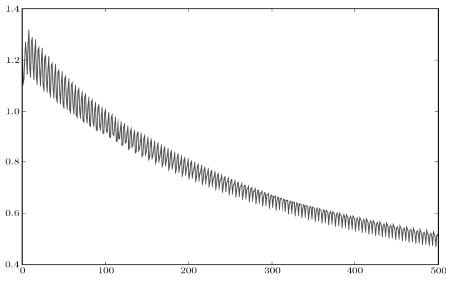
w = 0.17



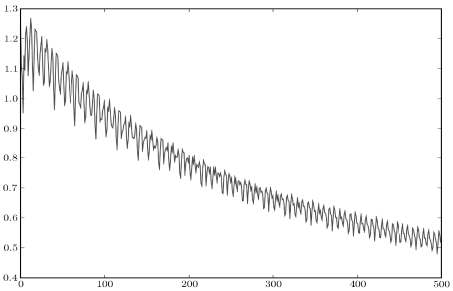
w = 0.15



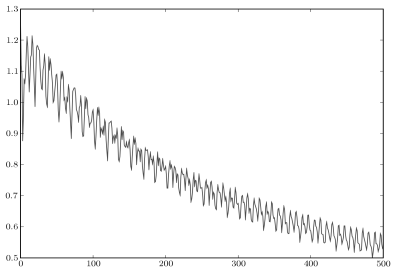
w = 0.125



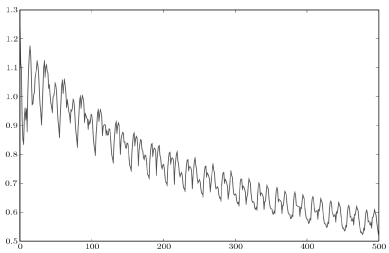
w = 0.08



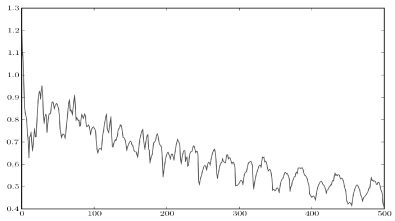
w = 0.06



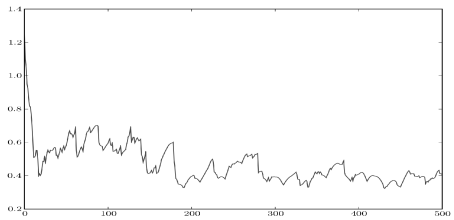
w = 0.04



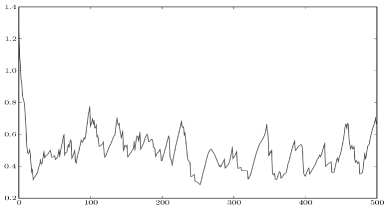
w = 0.2



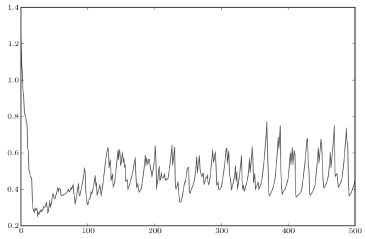
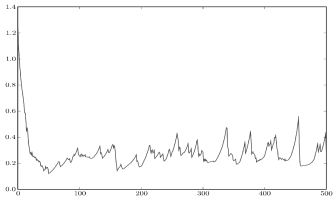
0.01



w = 0.0075



w = 0.005



w = 0.0025