One Dimensional Model of Sea Ice Movement

Edward Suh

May 1, 2004

Abstract

The goal of this project is to approach the behavior of sea ice by using a one-dimensional cellular model that simulates the movement of discrete ice floes. These blocks experience inelastic collision, where momentum is totally conserved and only velocity changed. First, this model is compared to a similar 1-D continuum model. Later, we introduce a time and position-varying stress that will simulate the effect of ocean surface currents.

1 Comparison of two models

This project was originally motivated by the 1-D Lagrangian PDE system describing sea ice movement created by Helga Schaffrin. Our approach was to create a similar model that treated the movement of sea ice as the interaction of discrete bodies in order to shed some light upon her simulation.

In our initial model, there were no external forces exerted on the ice. Without any drag, the velocity of an ice floe only changed upon collision with another, which was considered to be totally inelastic. It was possible, after making the appropriate transformations between Eulerian and Lagrangian coordinates, to make a comparison between this collision model and the PDE model. To do so, we calculated from the results of both models the distribution in the density of ice coverage that resulted when the same initial conditions were entered into each.

1.1 Continuum Model

This model approximates sea ice as a fluid of variable density, where the density is related to the percentage of ocean covered by ice. It is based on the equations governing the conservation of mass, conservation of momentum, and the inability of the ice coverage to exceed 100% of the ocean surface. Schaffrin gives these equations, in Eulerian coordinates, in one dimension, and with no sources or sinks of mass as:

$$(ch)_t + (chu)_x = 0$$

$$u_t + (\frac{u^2}{2})_x = \frac{F}{ch}$$

where c = concentration, h = height, u = velocity, and F is the sum of all forces acting on the ice. The product ch acts as the density in traditional fluid equations. Furthermore, height is taken as a constant, $h = 1$. In addition to these two equations, there is an additional criteria that $c < 1$, i.e., that the concentration of ice cannot exceed 100%.

Schaffrin then performed a Lagrangian transformation, using the Lagrangian variables $\tau = t$ and $\xi = \int\limits_0^x c\,d\hat{u}$. After further derivation, she derived a way of treating pressure as a Lagrangian multiplier, and concluded with a system of three equations,

$$\begin{cases} u_\tau = -p_\xi \\ \rho_\tau = u_\xi \\ \rho > 0 \end{cases}$$

where $\rho = \frac{1}{c} - 1$.

She used a finite differencing technique to create a code that would produce data corresponding to velocity, u, pressure, p, and the variable ρ, given a space-time grid and initial velocity curve. Setting the initial velocity as $u = \sin 2\pi\xi$, with $d\xi = 0.025$ and $d\tau = 1/100$, until $\tau = 1$, yields the curve in figure 1.

1.2 Collision Model

It was desirable to create a model where the behavior of sea ice was simulated on a molecular basis, i.e., where the interactions between individual floes could be simulated. In order to do this, we set up a model that simply advanced individual particles that had given (equal) widths and masses and

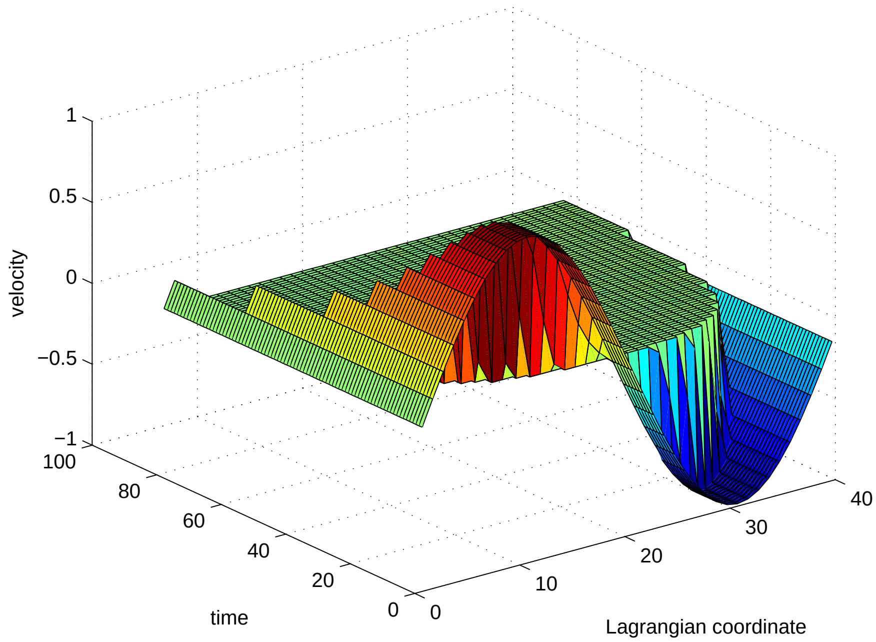


Figure 1: velocity curve vs. time and coordinate, given by continuum model

an initial velocity distribution. The system is without any frictional forces, and the velocity of individual particles remains unchanged until collision.

The velocity of particles after collision was given using just the equations for conservation of momentum. After a collision, the velocities of all particles involved in it are equal and given by

$$v_f = \frac{m_1 v_1 + m_2 v_2}{m_1 + m_2}$$

Advancing this through a initial particle distribution that starts with 40 blocks evenly spaced between 0 and 1, $dt = 0.100$, initial velocity given by $u = \sin 2\pi x$, and until $T = 1$, yields the curve in figure 2.

1.3 Density Comparisons

The similarity of the two graphs describing the evolution of a similar initial velocity distribution over time seems to indicate an agreement between the two models. In order to quantitate the agreement of the models, we ran

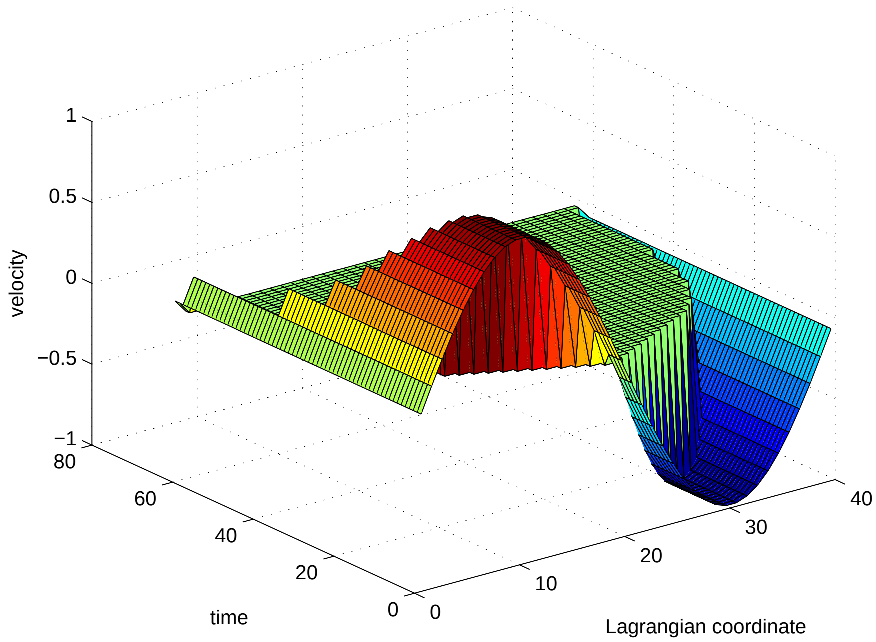


Figure 2: velocity curve vs. time and coordinate, given by collision model

simulations using equivalent initial conditions, and made a comparison based on the distribution of ice density over space.

To create equivalent conditions, it was first necessary to convert the Lagrangian coordinates used by the continuum model to their corresponding Eulerian ones. Since $\xi = \int_0^x c\, d\hat{u}$, it was necessary to make the blocks in the collision model evenly distributed from 0 to $1/c$, and the initial velocity curve $u(x) = \sin(2\pi cx)$. The number of blocks was set to 40 ($dx = 1/40$) and the time step was set as $dt = 1/50$. Furthermore, since the concentration c was set at $2/3$, it was necessary to set the block width in the collision model to appropriately reflect a uniform ice density given the number of blocks simulated.

Because the PDE model kept track of only the Lagrangian coordinates, to recover the Eulerian coordinate positions of points, it was necessary to integrate the velocities of these points. On the other hand, it is possible to read the positions and densities directly from the output of the collision model. Figures 3,4, and 5 show the correspondence of density given these initial conditions at 3 points in time - the initial density distribution, at the point of first collision (or first non-zero pressure in the continuum model, at $t = .0800$), and after 0.5000 time had passed. The graphs show close similarity between the two simulations.

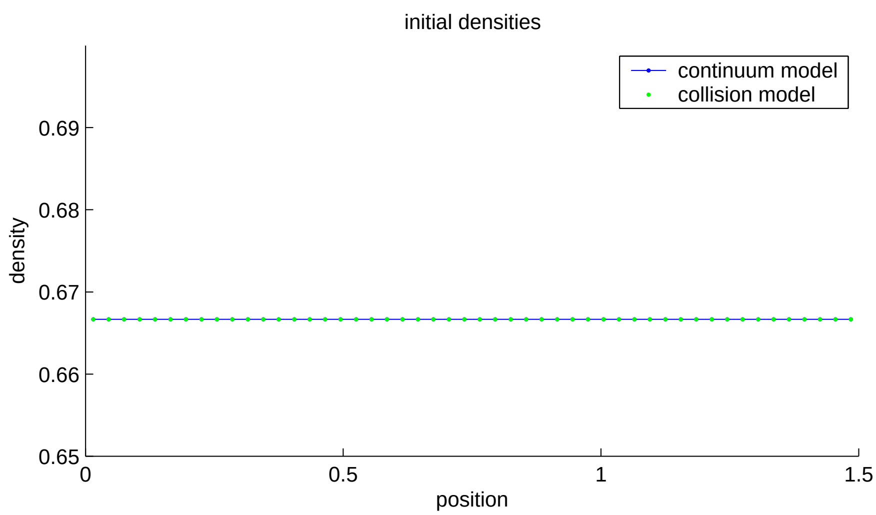


Figure 3: initial density

Both models converge very well. When both dx and dt are halved, the time to first collision changes less than 0.32% (from .7991 at $dx = 1/25$, $dt = 1/125$ to .79659 at $dx = 1/50$,$dt = 1/250$) in the cellular model and apparently not at all in the PDE one (remaining at 0.80). Calculation of mass initially and after collisions seems also to indicate that both models are well-behaved. We were able to compute mass by integrating the density curves. The continuum model has an initial mass of 1.0000, and a mass of 1.0110 at time $t = 0.5000$. Similarly, the collision model has an initial mass of 1.0000, and a final mass of 1.0001. The slight inaccuracy of mass conservation is likely an artifact of the rough integration technique.

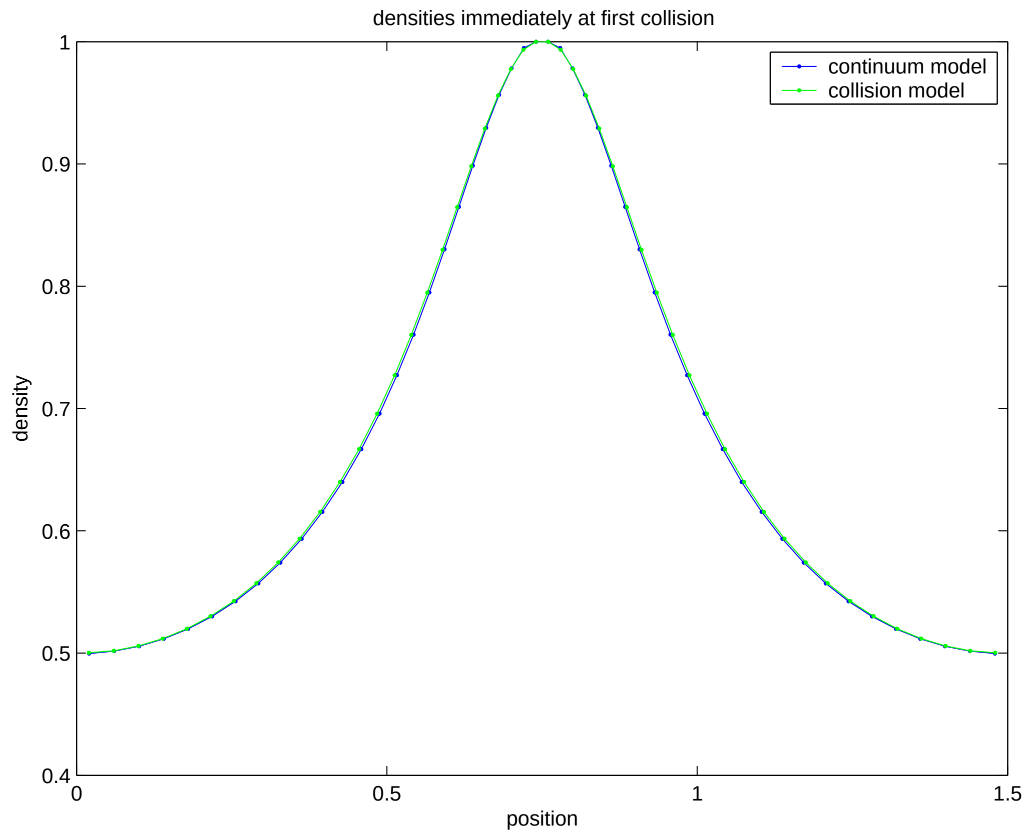


Figure 4: density after t=0.0800

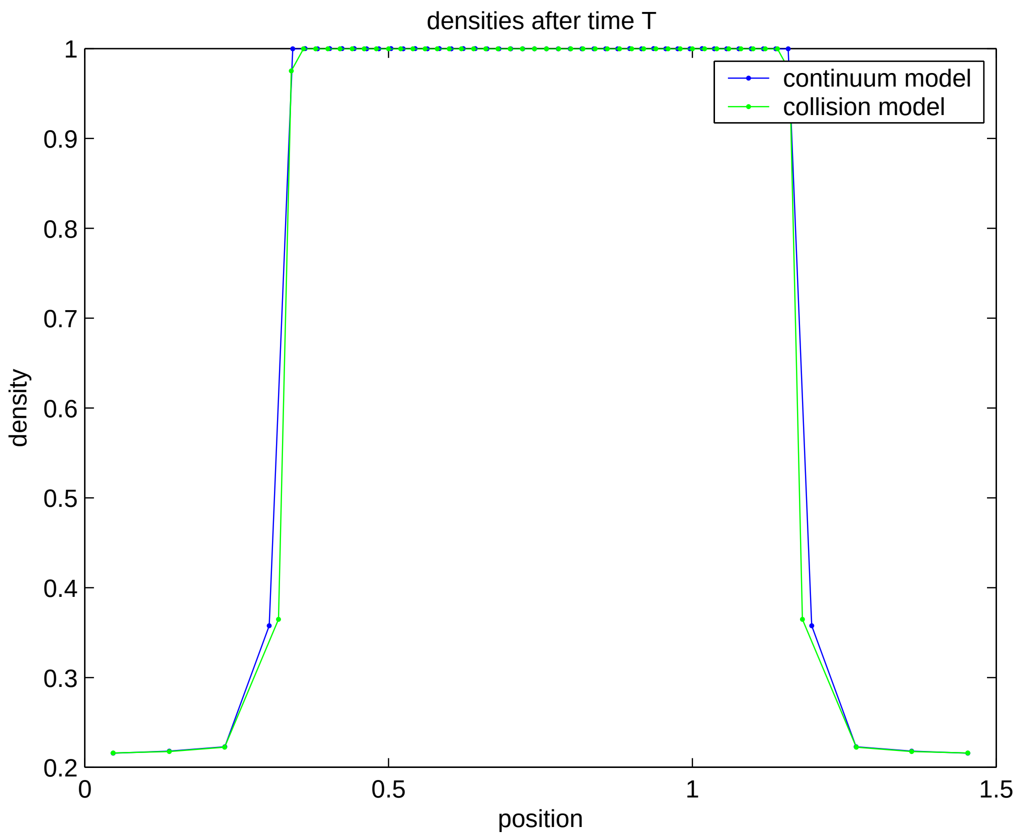


Figure 5: density after t=0.5000

2 Surface currents and velocity relaxation

We then introduced the force of the ocean surface currents on sea ice. The water's velocity does not act immediately upon the ice, but rather, the intrinsic velocity of the ice relative to the water experiences relaxation. With relaxation coefficient α, the velocity of the ice changes according to the relationship

$$u_t = -\alpha(u - w)$$

In our model, this was which was implicitly advanced using the formula

$$u^{n+1} = \frac{u^n - \alpha \Delta t w^n}{1 + \alpha \Delta t}$$

We used standing waves for the motion of the water, i.e., waves of the form $w = \cos(\Omega t)\sin(kx)$. In particular all of the figures shown use values of $\Omega = 0.35\pi$ and $k = \pi$. This produces a standing wave with stable and unstable nodes at every integer position, with the stability reversing over the oscillation. The effect of this wave on individual particles, given a significantly strong relaxation coefficient α, is shown in figure 6 to be an attraction towards the nodes. Although initially particles often oscillate between nodes, they seem to invariably fall into one node or another, provided a large enough time is taken.

Examining the numerical convergence of this model seems to indicate that it is relatively well behaved over short periods (figure 7). However, figure 8 demonstrates that observation over lengthy time periods indicates that the behavior is actually chaotic.

When multi-block systems are simulated, very interesting paths emerge. Because of the rapid changes in velocity that result after collisions, floes of ice actually can actually undergo ballistic motion. Clusters of ice that travel ballistically may eventually experience enough reduction in velocity to fall into oscillatory patterns around nodes. Conversely, clusters of ice in oscillatory patterns around nodes seem to experience enough difference in velocity to eventually collide and break out of these patterns. Because the effect of the water velocity on ice velocity is dependent on the relaxation coefficient, we examined the effect of changing α on one set of initial conditions.

Using $w = \cos(0.35\pi t)\sin(\pi x)$, $dt = 1/250$, and 20 blocks linearly spaced from 0 to 1 with an initial velocity $u = 0$, we ran simulations on varying values of α from 0 to 100. A selection of these results are displayed in the following figures.

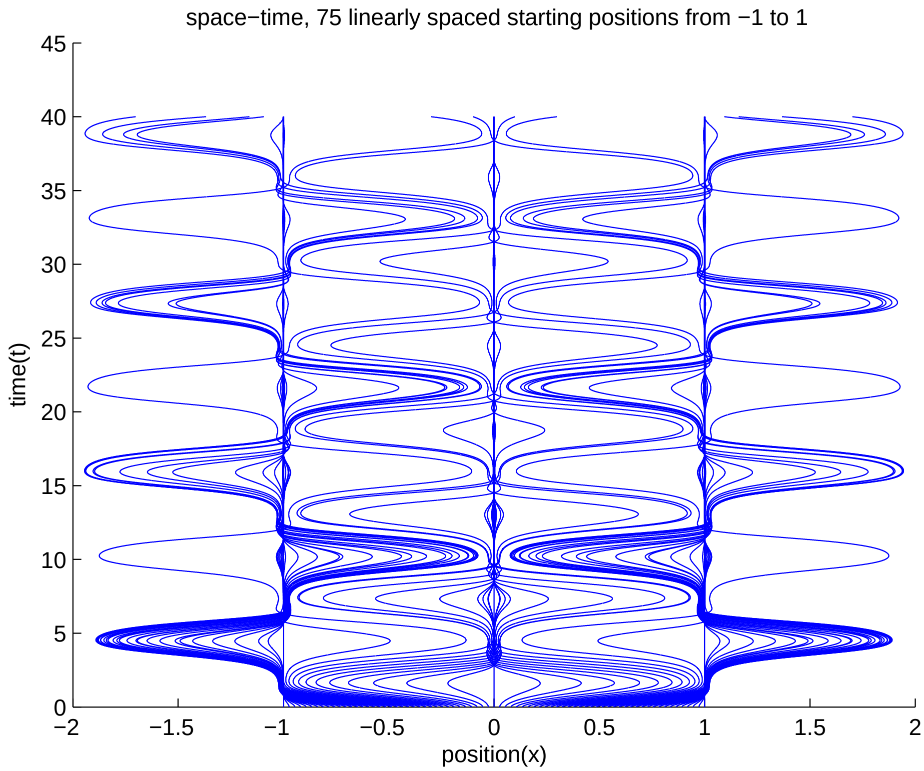


Figure 6: Space-time diagram for individual floes with relaxation

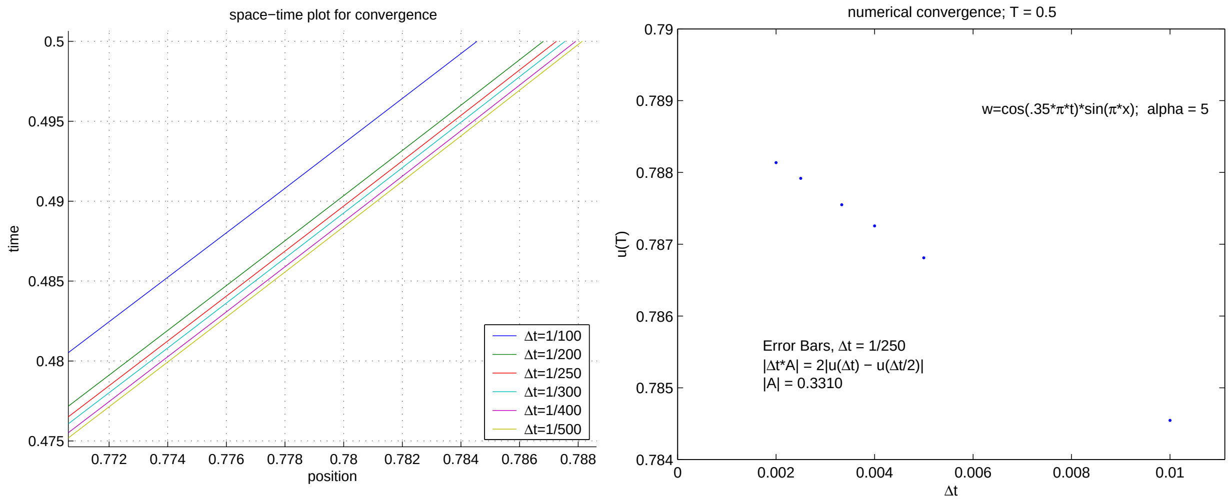


Figure 7: Numerical convergence for small time period

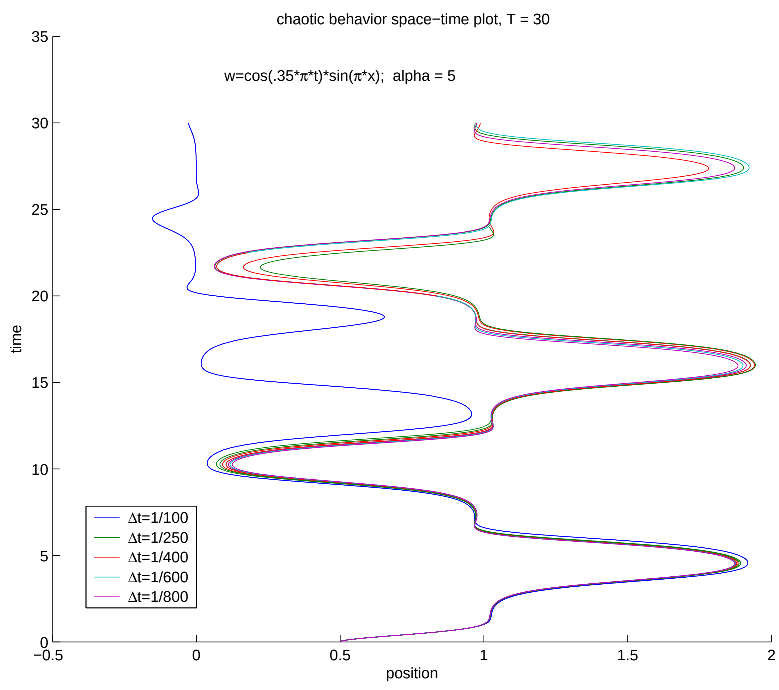
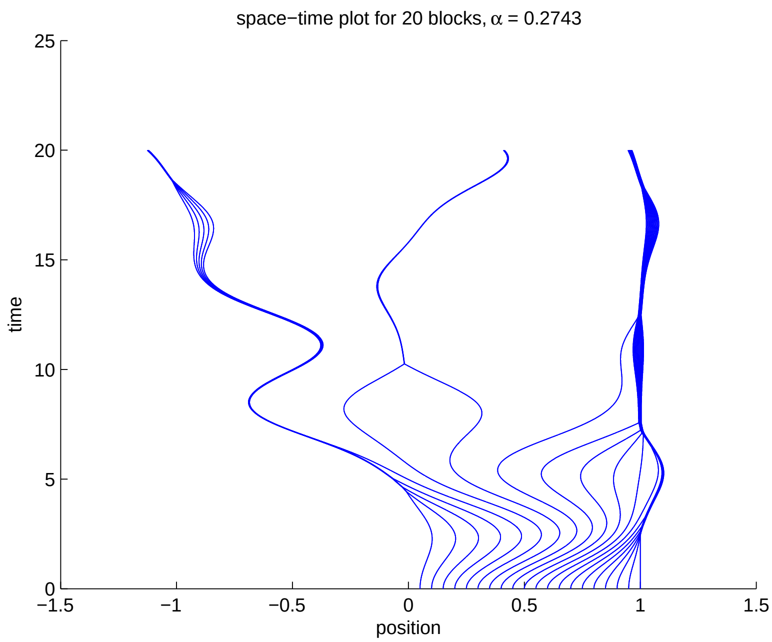
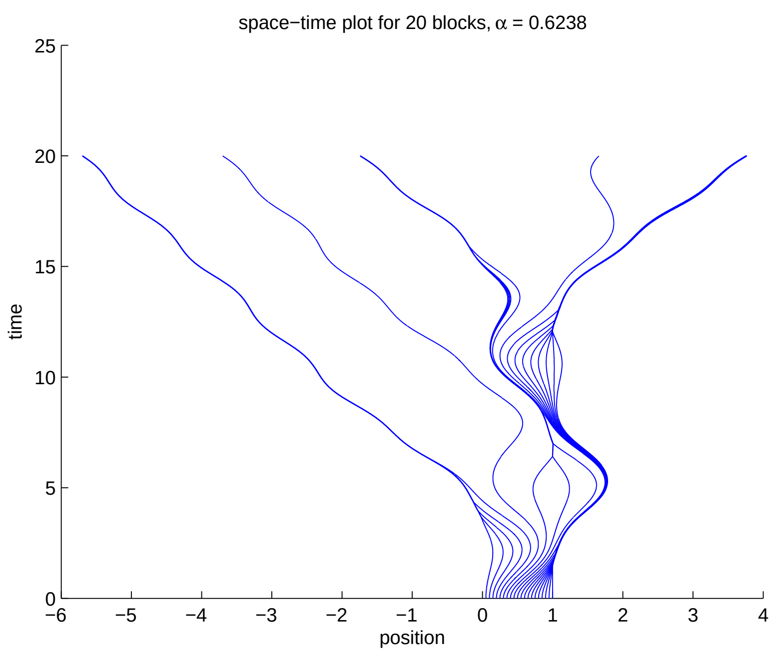
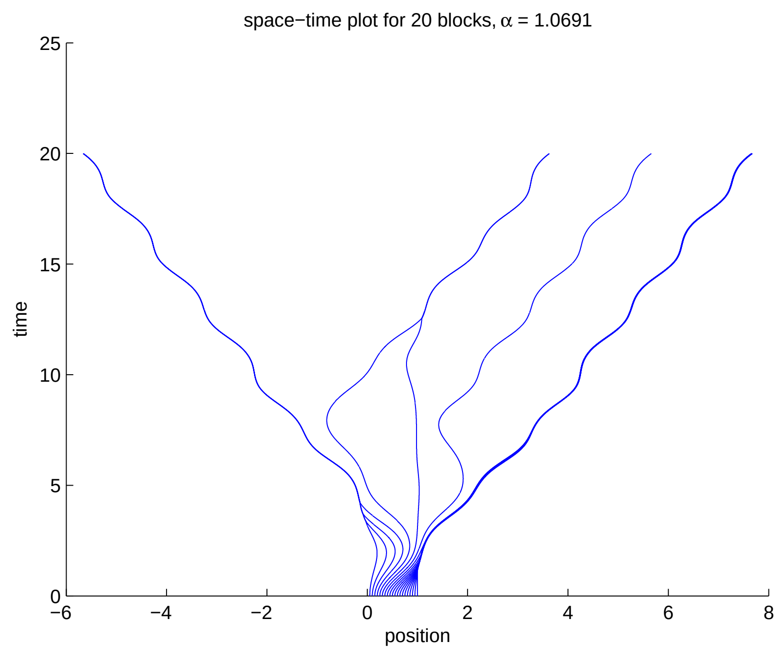
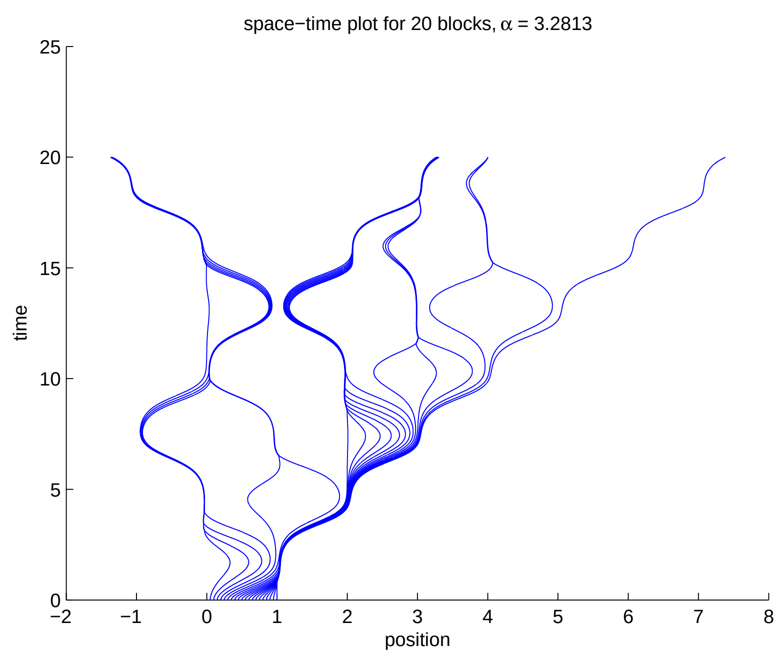


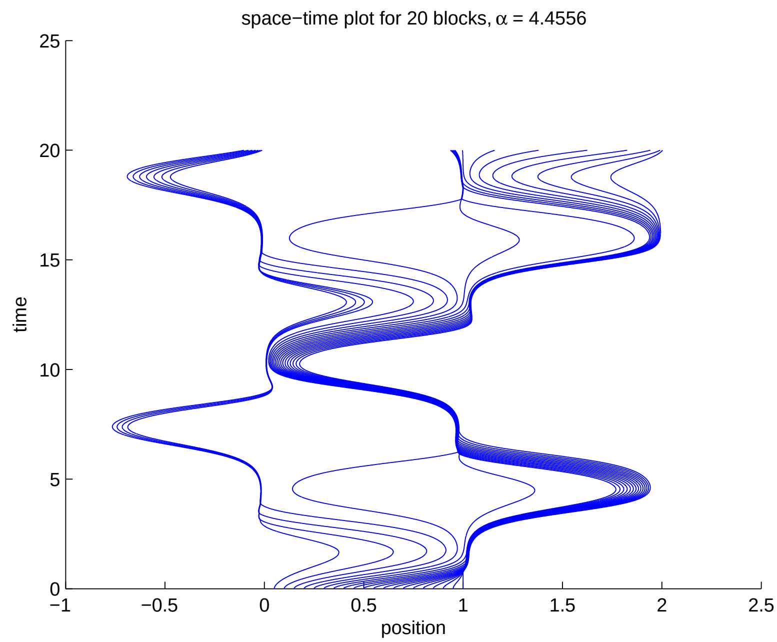
Figure 8: Chaotic behavior over large time periods

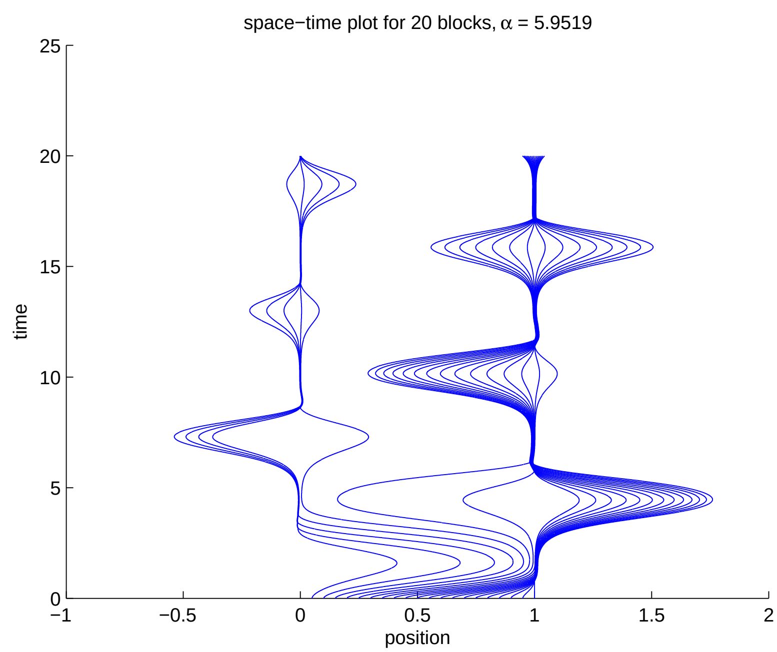

space−time plot for 20 blocks, α = 0.2743
time
position
25
20
15
10
5
0
−1.5
−1
−0.5
0
0.5
1
1.5

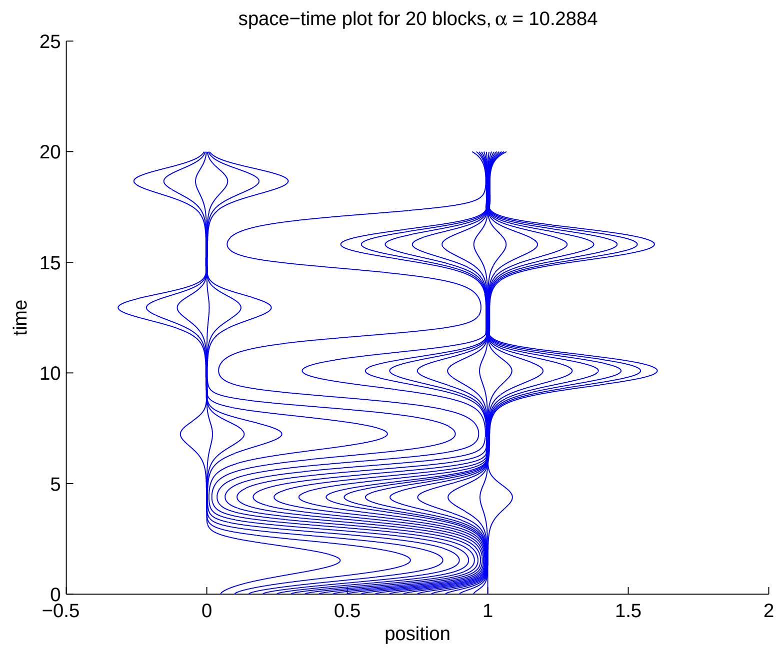

space−time plot for 20 blocks, α = 0.6238
time
position
25
20
15
10
5
0
−6
−5
−4
−3
−2
−1
0
1
2
3
4


space−time plot for 20 blocks, α = 1.0691
time
position
25
20
15
10
5
0
−6
−4
−2
0
2
4
6
8

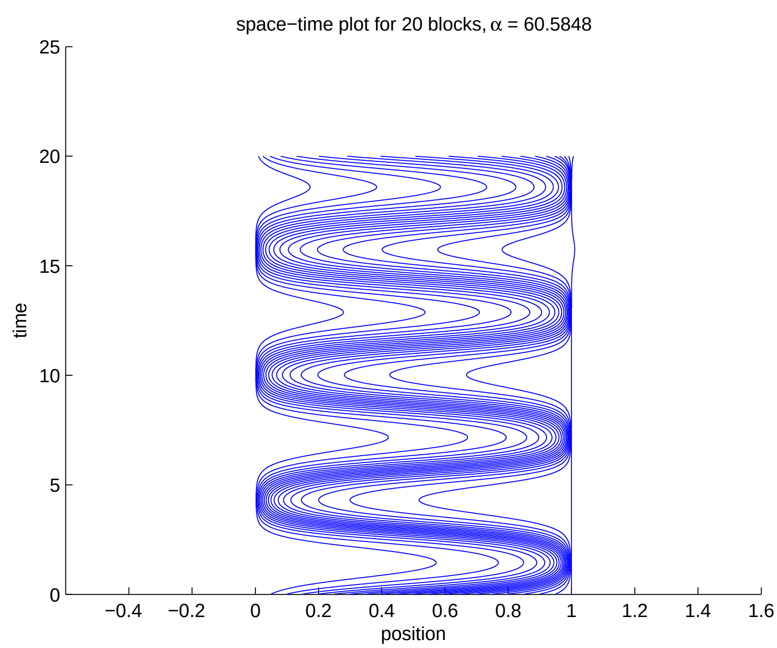

space−time plot for 20 blocks, α = 3.2813
time
position
25
20
15
10
5
0
−2
−1
0
1
2
3
4
5
6
7
8


space−time plot for 20 blocks, α = 4.4556
time
position
25
20
15
10
5
0
−1
−0.5
0
0.5
1
1.5
2
2.5

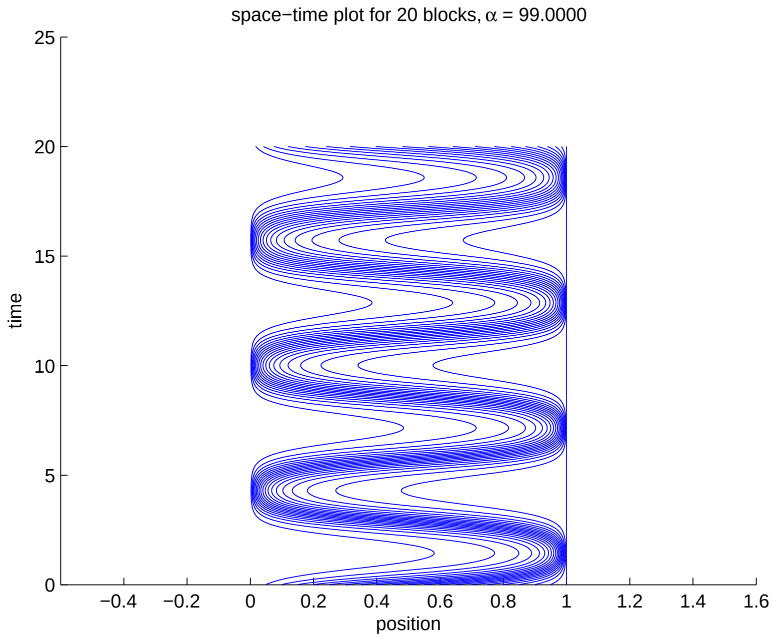

space−time plot for 20 blocks, α = 5.9519
time
position
25
20
15
10
5
0
−1
−0.5
0
0.5
1
1.5
2



space−time plot for 20 blocks, α = 10.2884
time
position



space−time plot for 20 blocks, α = 60.5848
time
position



It seems that even at high values of α, ballistic motion may be possible. Following the pattern of oscillations at higher values shows that although the relaxation becomes stronger, there is still enough freedom that ballistic motion may develop if a large enough time period is examined. Even observation of the last plot, at $\alpha = 99.00000$ shows that the highly compressed region spreads at a noticeable rate. It seems that a steady state may not ever be reached.

3 Conclusion

The comparison of Helga Schaffrin's PDE model and the collision model created from a cellular approach was very successful, in that it demonstrated that the movement of sea ice in the former seems to be consistent with the behavior of discrete floes undergoing inelastic collision. It may eventually be desirable to test different initial velocities at higher resolutions, but the data that we have gathered does not indicate anything else should be expected. As shown, the two models yield ice distributions that are remarkably close.

Introducing a velocity field to the underlying water had very interesting results as well. With a standing wave, the natural tendency of the particles was to move towards the stable nodes. Because of the oscillation of nodes between stable and unstable states, ice had a tendency to oscillate between or around nodes, until their velocity reached small enough magnitude that it was quickly dominated by the water velocity and sucked into a node. When ice was allowed to collide, however, fascinating behavior arose. The velocity-changing collisions prevented particles from reaching a steady state. Interactions between particles could even result in ballistic motion of clusters of ice.

Furthermore, changing the value of the relaxation coefficient α does increase the oscillatory motion, but cannot guarantee that particles will not break free and produce more complex patterns of motion. Even at high values, it is evident that non-trivial behavior will eventually result, if the system is given enough time to evolve.

4 The code

The entirety of the collision model, with both density calculations and water motion is contained in the function code `icemodel.m`. The density calculations, however, have been commented out. For density calculations without water motion, either run `comp_icemodel.m`, or run `compareflow.m`, which runs a comparison with the continuum model. It is a script, and calls on `ice_simplex.m`, `simplex.m`, and `pivot.m`, as well as `comp_icemodel.m`. To run the collision model with the relaxation for a single ice block or for multiple non-interacting blocks, (without collision), run `nod_icemodel.m`. This and `icemodel.m` both call on `wv.m`, which is the function for the velocity of water. To change the water velocity, edit this file.