# Simulating The Collective Behavior of Schooling Fish With A Discrete Stochastic Model

Alethea Barbaro , Bjorn Birnir, Kirk Taylor, 2006

Funded by
The National Science Foundation
and
The Research Fund of The University of Iceland

The goal in this project was to develop a mathematical model that can be used to simulate the behaviors of a school of fish, and thereby increase the general understanding of the behavior patterns of migrating schools. We used and expanded on a discrete model already well-known to researchers in this field. This document will attempt to explain how the model works, as well as describe our solutions.

Using what is known about how fish sense changes in their environment and use information to navigate (see the work of Partridge (1980, 1982)), we can use mathematics in a computer program to mimic certain known collective behaviors of a school of fish. Not all behaviors of fish are deterministic, especially those of individuals, but the unpredictability of living organisms can be simulated with random numbers in the program. If a model can be constructed in such a way as to leave room for data-driven parameters such as ocean currents, temperature data, food availability, etc. to be input, fisheries resource managers and scientists can better predict the sizes, directions, and spatial distributions of migrating or stationary schools of fish. Our model has this capability, but it is not discussed in this paper. We focused on some known patterns of fish behavior, and the required parameterizations for a successful application of the model to those behaviors. This model uses discrete equations to give solutions for localized behaviors. Similar models using continuous, ordinary differential equations can be used to study global behaviors, and three-dimensional adaptations are possible.

The values used for our parameterizations (in Frame 1, for example) are basically nondimensionalized, which allows them to be applicable to a variety of fish species. As an example, we considered the capelin fish of the north Atlantic, which ranges from 3 to 7 cm in length. The values, therefore of our zones of attraction, orientation, and repulsion could easily be interpreted in terms of centimeters for this particular fish.

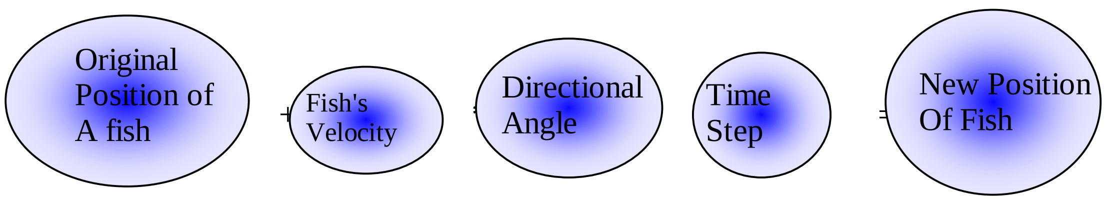
Our immediate goal was to achieve solutions of three basic types: migrating schools, swarming schools, and circulating schools, all using the same basic code but with different distance, velocity and angle parameters. The simulations produced short animations saved in .avi format. We ran them on a Hewlett Packard xw4300 computer using a Linux operating system. The computing program Matlab was used to write the code and run the simulations. The figures accompanying this article are snapshots of these animations, and the axes are scaled in nondimensionalized units.

## The Model

Our Model, in simplified form:

$$\begin{pmatrix} x_i(t) \\ y_i(t) \end{pmatrix} + v_j(t) \begin{pmatrix} \cos\theta_i(t) \\ \sin\theta_i(t) \end{pmatrix} \Delta t = \begin{pmatrix} x_i(t+\Delta t) \\ y_i(t+\Delta t \end{pmatrix}$$

Which simply means:




In the program itself, this equation looks like this:

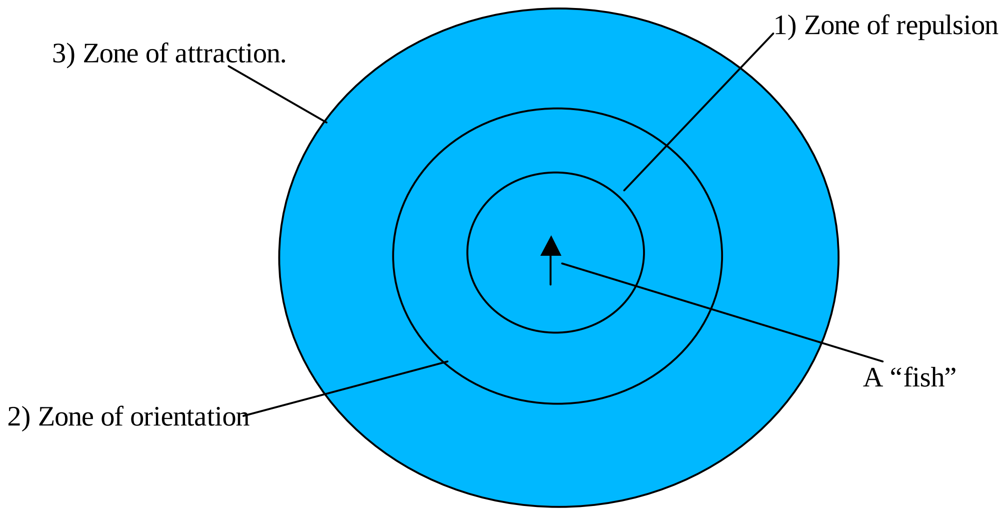
```
x(k) = x(k) + dt * v(k) * wVector(1) / norm(wVector);
y(k) = y(k) + dt * v(k) * wVector(2) / norm(wVector);
```

This is the final equation of a long series of code that defines and controls many parameters that determine the position, velocity and angle of each fish. Most of those parameters are initialized and defined with the block of code shown in Frame 1. Our code also included a program to define other features of the fish and their interactions, a program to draw each fish in a frame using the mathematics and geometry outlined in the code, and a program that linked frames to make the animations.

Before discussing the actual solutions, we should first explain some of the basic features of the program. We have based our model on the principles outlined by Czirok, Vicsek, Ben-Jacob, Cohen, and Shochet (1999). Our model uses a zone of attraction, a zone of orientation, and a zone of repulsion for each fish. See Frame 2 for the code relating these zones. Briefly, a zone of attraction is responsible for fish gathering, or coalescing into a school, a zone of orientation is responsible for fish becoming aligned with each other, and thereby producing parallelism, and a zone of repulsion helps keep fish from excessively colliding with one another.

In the “grammar” of our program's code, a “*k*” is the subject fish, and a “*j*” is an object fish. That is, in all calculations, *k* is at the center of the zones, and *j* is a neighboring fish within one of the zones. Referring to Frame 2 will help the reader to understand how the mathematics is structured after reading the following explanation.

**The Zones of Attraction, Orientation, and Repulsion,** and how they influence the behavior of a fish.




1) For any fish *j* that is within the radius of repulsion for fish *k* set in the initial conditions of the program, the code calculates the angle away from that fish and averages its present angle with the calculated angle. In this way a fish will turn away from another fish if it gets too close.

2) For any fish *j* that is within the radius of orientation for fish *k* set in the initial conditions of the program, the code simply takes a weighted average of the *j*'s angle and *k*'s angle. In this way a fish will tend to align with other fish within a specified radius of orientation.

3) Finally, for any fish *j* that is within the radius of attraction specified for fish *k*, the code calculates the angle towards that fish and averages its present angle with the calculated angle. In this way a fish will turn towards another fish within this radius.

*   *   *

These calculations are done at every time step interval for every fish usually taking in the positions of many fish at once and balancing all the weighted averages. Then, having “chosen” an angle, the fish moves according to a specified velocity. The velocity also has a random component, so that how fast a fish moves towards, away from, or to alignment with another fish is also subject to stochastic perturbations.

It should be emphasized here that the above-mentioned averages between angles are in fact weighted averages, and that the weights used for these averages are important controllable parameters of this program. In fact, it will be shown later that the relative sizes of these weights and indeed their very nature are crucial determinants of the desired solutions.

## Migrating Schools

A *migratory solution* is characterized by two basic phases. In the first phase the fish are observed coalescing from a randomly scattered configuration into a school. **Figure 1** shows a simulation after just a few iterations, and the fish are all oriented towards a centralized location. In

**figure 2** the fish are completing the coalescence phase and moving towards a parallel motion, but some fish are still outside of the school and the school's final direction has yet to be determined. Finally, in **figure 3** the school is formed, has assumed parallelism, and is beginning to migrate. Once the direction of migration has been determined the school will remain headed in that direction, or vary only slightly for as long as the simulation runs. These snapshots are from a simulation called "trueMigration.avi" and the salient parameters are initialized thusly:

```
numberOfFish = 250;            >this is how many fish are in the simulation
v = 5*ones(numberOfFish,1);    >a velocity coefficient
dt = 0.25;                     >the time step
radiusOfOrientation = 55;      >the sizes of the zones are in nondimensionalized
                              units
radiusOfAttraction = 150;
radiusOfRepulsion = 12;
attractionWeight = 1.5;        >weights are used in calculating angular direction
repulsionWeight = 0.7;
ampVNoise = 0.2;               >controls the amount of random velocity variation
ampAngNoise = 0.55;            >controls the amount of random angular direction
```

Frame 1.

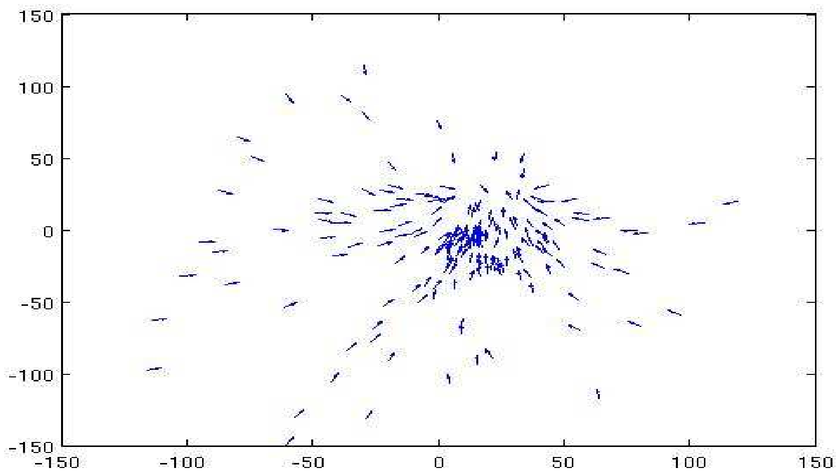



Fig.1 The beginning of the coalescence phase.

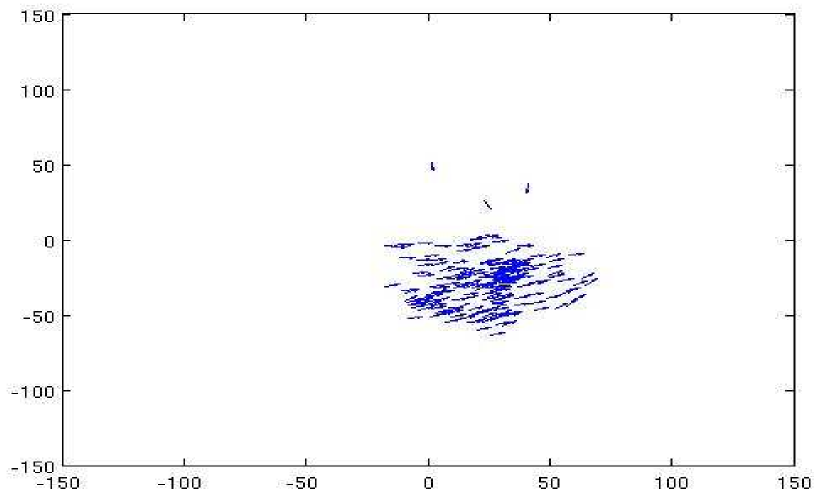



Fig.2 The school is nearly completely formed, most fish are aligned but a final direction has not been chosen.

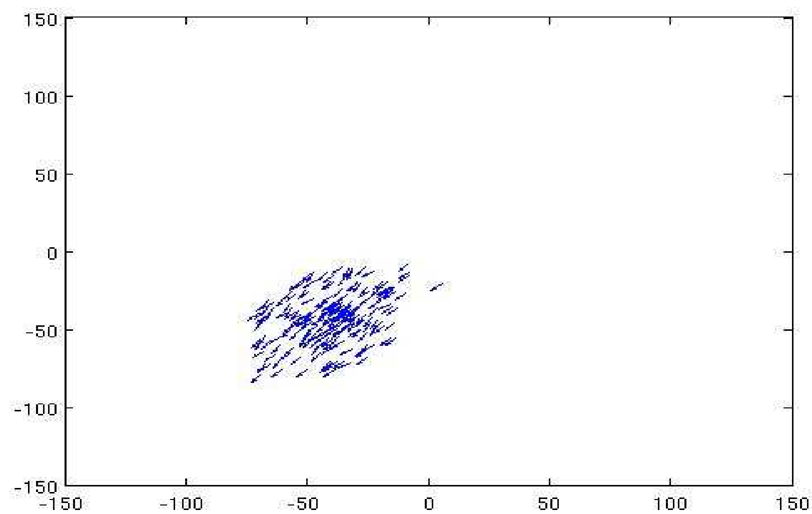



Fig. 3 The school is now completely parallel, and is migrating in a straight line.

The migratory solution, out of the three we studied, has the simplest components. A large zone of attraction ensures that most, if not all fish join the school, and a small zone of repulsion allows the school to be more or less homogenous without artificial internal structures. The angle noise in the example above (Frame 1) is relatively moderate; it is low enough so that the fish swim in a natural-looking manner, but high enough to introduce significant unpredictability. Small velocity and time step changes have little noticeable effect on the behavior of the school, but will slightly alter the simulation's appearance. Larger adjustments of velocity and time step produce simulations that do not look realistic and may not be useful.

**The migratory to swarming continuum**

In these, and all the simulations we have done, a balance has had to be struck between the sizes of the zones, their respective weights, the noise, and the total number of fish. Each parameter can be thought of as a continuum, with behaviors adjusting in logical ways as one moves along the continuum. Along these continuums there are "tipping points" where the school's behavior actually changes from one state to another. For example, decreasing the zone of attraction relative to the number of fish might force smaller, more numerous schools to form, or simply prevent any schools to form. Or, decreasing the zone of repulsion relative to the other zones might force small clumps of fish to stabilize within a school and create artificial internal structures. While examples like these abound, all parameters work in concert with each other and it is often difficult to pin a specific behavior to one particular parameter. Similar behavior changes can be achieved by adjusting different parameters, as will be seen.

To achieve a stable, swarming school of fish we introduced a new parameter, which we called "self weight", that described a fish's tendency to swim in the same direction from one time step to the next. This parameter appears to correlate with the number of fish in that the more fish are in the simulation the higher self-weight needs to be for the school to remain stable. In fact, we found self-weight needed to be approximately 1.5 to 2 times the total number of fish to achieve a swarming solution. This greatly outweighs any other of the zonal weights, sometimes by a factor of 100 or more. In exploring the effects of the self-weight parameter we found that a stable, swarming solution can be made migratory by adjusting down only the self-weight. Adjusting it a small amount will cause the swarm to drift around, but there is still no parallelism in the school. Adjusting it down more causes more drift, and some fish begin to align with each other. Further downward adjustment, approaching 25% or less of the number of fish, pushes the school over a tipping point where it assumes parallelism and travels in a well defined direction, in other words, it becomes in fact a migratory solution.

Alternatively, a stable swarming school can be made migratory by adjusting only the amount of angle noise, although the numbers are less dramatic. Our stable swarm used an angle noise coefficient of 0.72. When it is gradually adjusted downwards to around 0.12, the same continuum of behaviors is observed as when the self-weight is decreased, although there are some differences. By adjusting the angle noise, the school also achieves a parallel motion and migrates, but each fish is moving in a much smoother path.

Angle noise and self weight are two examples of parameters that control a continuum of behaviors, along which a tipping point exists where the school changes from migratory to swarming, and vice versa. Other parameters, such as the number of fish, and the size of the zones operate similarly. An interesting example of this balance-counterbalance interplay is in **SWmigration.avi**, in which a randomly scattered configuration begins to coalesce, but then hesitates and almost stops all movement for several seconds. Then a small stochastic perturbation alters the direction and the group continues to form and grow, changing direction and gathering speed. Other trials of this particular simulation resulted in the fish actually swimming backwards for more than half of the simulation. In this simulation the zone of attraction is only slightly larger than the zone of orientation (120 and 100, respectively), and the zone of repulsion is small (5), and a high degree of noise is introduced in both velocity and angular direction. Clearly, this is not a realistic simulation of fish behavior, but it helped us learn more about the zones.

Considering these continuums of behavior, some migratory solutions **could** more realistically be termed "*semi-migratory*" because the school does move in parallel, but does not strictly adhere to any particular direction. Instead the school wanders more or less, according to some internal stochastic perturbations (see Figure 4). It is interesting to note that for the more strictly migratory solutions, the final direction chosen by the school is predominantly determined by the position of the very last fish or group of fish to join the school, which illustrates two important ideas: 1) the way the three zones work together, and 2) the principle of collective behavior. The fish in the lead of the school "see" that fish once it enters their zone of attraction and move towards it, then they align themselves with it when it enters their zone of orientation, and all the other fish in the school follow suit. It works exactly the same for the outsider fish, of course, and when the school gets close enough to be within its zone of repulsion it turns around, thereby moving ahead of, but in the same direction as the school. This seems to be reasonable behavior for a school of fish. The other principle at work here is the idea of collective behavior. This interplay and balance, whereby the behavior of the whole group is determined by a complex combination of predictable responses to outside stimuli and stochastic perturbations, and no single member of the group is designated as a leader acting with its own volition, is at the essence of the entire simulation.

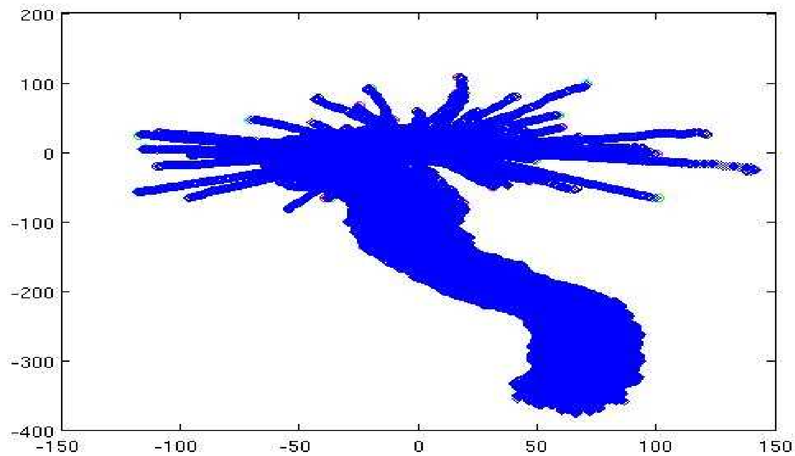



Fig. 4 A tracking diagram of a semi-migratory solution. Fish coalesce at top, then migrate in a randomly determined path towards bottom of diagram.

## Stable Swarming Solutions

In searching for swarming solutions, we were looking for a school to coalesce and stay primarily in one location with the fish continuously moving around in smooth, randomly directed paths. A mentioned earlier, the addition of the self-weight parameter, and by adjusting the angular noise we could easily make the fish swarm. The inclusion of self weight is achieved by letting the $j$ and $k$ identifiers be equal, and adding $j$'s angle multiplied by the self weight, as shown in Frame 2. Adding a self weight makes sense in that it is reasonable for a fish to have a certain directional or angular momentum, that is, a tendency to keep swimming in the same direction at any given time step. A good example of a swarming solution is **mediumSwarm.avi**, one frame of which is shown in Figure 5. In it one can see the fish coalesce within the radius of orientation, and simply swim around randomly. The only unifying characteristic of the fish is that they stay in the school; there is no parallel alignment whatsoever. The school drifts around slightly, and in repeated trials of this simulation the school sometimes drifted more and sometimes less, but there is no direction chosen, and the drifting appears to be random. When more fish are added to the school, the self-weight needs to increase, keeping at about 150% of the number of fish. By keeping this balance and making other small changes a swarming solution can be obtained with virtually any size school.

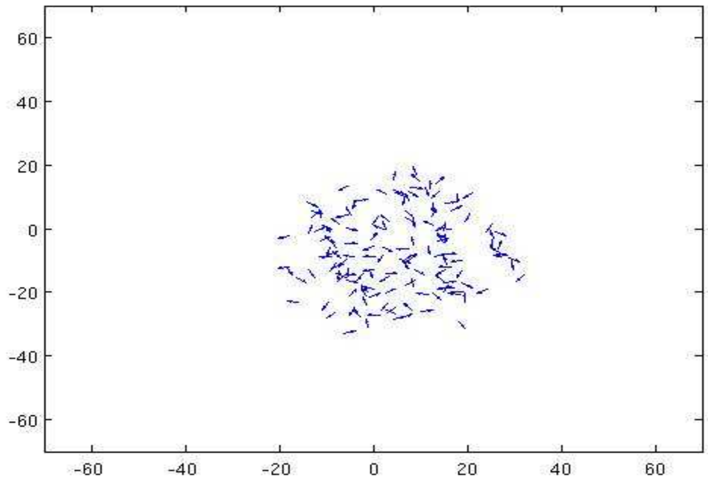



Fig.5 A frame from “mediumSwarm.avi”

Here is where “Self Weight” is added into the program. Just like the zones of attraction, orientation, and repulsion, 'self weight' is used to influence the angle of the fish. Notice also how the radius of repulsion subtly differs from the radius of attraction.

```
if(j==k)
    averageCos = averageCos + selfWeight*cos(angle(j));
    averageSin = averageSin + selfWeight*sin(angle(j));
    counter = counter + selfWeight;
elseif(dist <= radiusOfRepulsion)
    angleBetween1 = atan2(y(k)-y(j), x(k)-x(j));
    averageCos = averageCos + repulsionWeight*cos(angleBetween1);
    averageSin = averageSin + repulsionWeight*sin(angleBetween1);
    counter = counter + repulsionWeight;
elseif(dist <= radiusOfOrientation)
    averageCos = averageCos + orientationWeight*cos(angle(j));
    averageSin = averageSin + orientationWeight*sin(angle(j));
    counter = counter + orientationWeight;
elseif(dist <= radiusOfAttraction)
    angleBetween2 = atan2(y(j)-y(k), x(j)-x(k));
    averageCos = averageCos + attractionWeight*cos(angleBetween2);
    averageSin = averageSin + attractionWeight*sin(angleBetween2);
    counter = counter + attractionWeight;
end
```

Frame 2.

Below are the same initialization parameters (see Frame 1, also) for a swarming solution, for comparison with the migratory solution. Note also the inclusion here of orientation weight.

```
numberFish=150;
v=2.0*abs(randn(1,numberFish));
dt=.5;
radiusOfOrientation=40;
radiusOfAttraction=120;
radiusOfRepulsion=4.5;
repulsionWeight = 1.15;
attractionWeight=3.35;
orientationWeight=1.0;
selfWeight = 350;
ampVNoise=.050;
ampAngNoise=0.69;
ampPosNoise=0.15;
```

Frame 3.

## Circulating Solutions

The circulating solution, or torus solution, grew from the swarming. We began with a stable swarm, and found that by increasing only the radius of repulsion relative to the radius of orientation we could take a swarm solution and give it ring formation, but the difficulty was in forcing the fish to swim *around* the circle. In all of our initial attempts the school either assumed parallelism and began a migratory path, moving as a rigid circle, or the fish just moved within a small section of the circle, maybe 60 to 180 degrees of arc, and occasionally crossed over to another part of the circle. See CircleMovie1.avi for an example of this as well as Figure 6. It seemed logical that the problem was in the angles chosen by the fish, so focusing on the orientation weight and the angular noise made sense. Increasing the orientation weight slightly affected the circulation, but not significantly nor consistently. The same could be said of lessening the angular noise. Orientation weight, as well as the other weights had up to this point always been constant parameters. We could change them for any trial, of course, but during any trial they were always constant. We wondered what would happen if randomness were introduced into this part of the simulation, and so multiplied the orientation weight by a random number. This produced a stable, circulating school of fish, a snapshot of which is shown in Figure 7. Repeated trials, and trials of much longer duration always showed the same stable circulating school. Furthermore, if instead of a random number used as an orientation weight a constant value from within the same range was used, the school quickly assumed parallelism and moved into a directly migratory solution, albeit an unrealistic-looking one.

In Frame 4 are the initializations for one version of a circulating school. However we found successful solutions using a rather wide range of values for some of these parameters. For example, the attraction weight can vary from about 3.5 up 8, and the solution persists, although the thickness of the ring increases with the lower attraction weights, and more fish cross over or temporarily leave the circle. Repulsion weights should scale with attraction weights. Self-weight can vary widely, from about 200 to 400, without changing the number of fish. The absolute sizes of the three zones can change, but should remain basically within these proportions. The question of the scaling relationships between these parameters is interesting, deserving further study.

```
numberOfFish = 300
v=abs(3.2*(randn(1,numberOfFish)));
dt = 0.45;
radiusOfOrientation = 40;
radiusOfAttraction = 60;
radiusOfRepulsion = 32;
selfWeight = 295;
attractionWeight = 4.55;
repulsionWeight = 2.85;
ampVNoise = 0.65;
ampAngNoise = .03;
```

Frame 4.

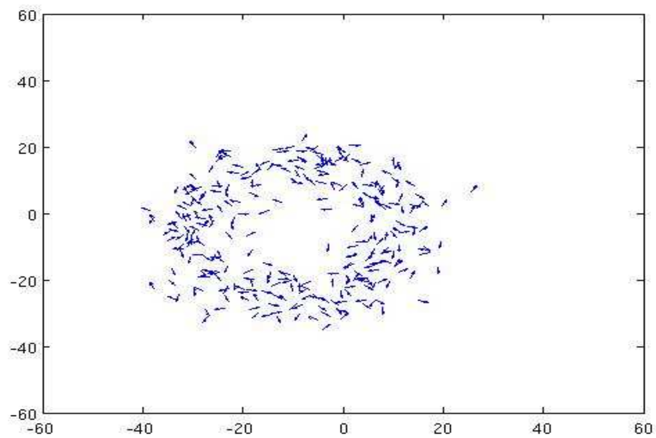



Fig.6 The non-circulating circle of fish. Note that although fish are configured in a ring, some fish swim outside or inside the ring, and fish are positioned at random angles around the ring.

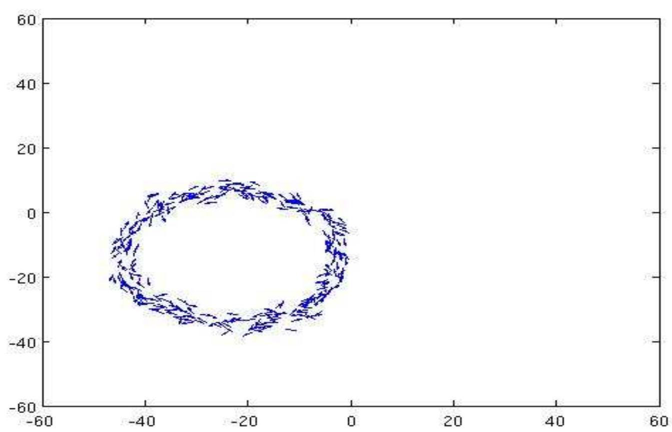



Fig. 7 A stabilized circulating school. Note the small amount of angular deviation.

Why did this happen? In viewing CircleMovie2.avi**,** or in the even more dramatic example of **stableCircle.avi** , one immediately notices the effects of a strong attraction weight and the small difference between the zones of orientation and repulsion. The school immediately coalesces into a ring formation out of the initial randomly scattered configuration, and the size of the ring is consistent with the radii of orientation and repulsion. The next observation is that at first, fish are behaving much like in **CircleMovie1.avi**, in that they are in a ring but are not necessarily circulating around the ring. The angular variation is controlled somewhat by the small difference between the zones of orientation and repulsion, but not enough to force them around the circle. Then the behavior begins to change. After a few more iterations, the fish gradually reduce their angular variation, and begin to circulate, while occasionally straying outside the ring, but never changing direction. The longer the simulation runs, the tighter the control on the angular direction of the fish. In a sense, the fish are caught between the zones of repulsion and orientation, and the average angles are approaching a limit within these boundaries over time. It appears that because the orientation weight is variable it is allowed to adjust to within the set zones.

These circular solutions appear to be stable, and in fact increase stability over more iterations of the program, and are characterized by fish circulating in both directions simultaneously around the circle. Unidirectional solutions appear unstable, or at least the solution of a stable, unidirectional circular solution has yet to be achieved with this program. They can be made metastable, however, so we believe they represent transient solutions. Moreover, exactly how the discrete solutions achieved in this model translate into three dimensions has yet to be explored.

## Conclusions and Opportunities

We have shown that using one discrete model a variety of solutions are possible, and that the great many combinations of adjustments and balancing of this set of parameters creates a very rich field of possibilities. Some of these parameterizations operate as continuums, with tipping points of state change. We have focused on three solutions: migratory, stable swarm, and stable circle. Especially interesting are the stable solutions, because their facilitation required the introduction of the self-weight, and stochastic weighting, which both add another layer of richness to the model. The investigation into the scalability of parameter sets presents an opportunity for further investigation, as does a three dimensional version of the model. Further work is also possible on transient solutions such as unidirectional circular solutions and ones that incorporate elements of both stable and migratory solutions. It is our hope that this work will be some contribution to the already significant body of knowledge aiding fisheries resource managers and scientists, in that it gives another perspective from which to analyze collected data.

**Cited Works:**

[1] B.L. Partridge. The structure and function of fish schools. *Scientific American,* 245:90-99, 1982.

[2] B.L. Partridge and T. J. Pitcher. The sensory basis of fish schools: relative role of lateral lines and vision. *Journal of Comparative Physiology,* 135: 315-325, 1980

[3]T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen, and O. Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters,* 75(6): 1226-1229, 1995