CHAOS

Chaos is a program that explores data analysis. A sequence of points $x_0, x_1, x_2, x_3, \cdots$ is created which can be analyzed via one of the following five modes:

1. Time Series Mode, which plots a time series graph, that is, the points (n, x_n).
2. Delay Mode, which plots a delay graph, that is, the points (x_n, x_{n+1}).
3. Close Pairs Mode, which plots a close pairs map for a sequence of values.
4. IFS Mode, which constructs a plot based upon the Chaos Game, but where the rules (corners) are chosen according to a predetermined sequence of values.
5. Kelly Mode, which creates Kelly diagrams.

Upon execution of **Chaos**, the user is first asked to choose a mode. After that a user can change modes by selecting the appropriate option under the Mode menu. In each of these modes a recursively defined sequence of numbers is generated, where an initial seed x_0 is specified and given the current number x_n in the sequence, the next number x_{n+1} is generated by a certain rule. Upon initial execution of the program the tent map rule $T(x) = \frac{s}{2} - s\left|x - \frac{1}{2}\right|$ is operative so that $x_{n+1} = T(x_n) = \frac{s}{2} - s\left|x_n - \frac{1}{2}\right|$, where *s* is a parameter initially set to the value 1.5. Other pre-defined rules may be selected from the Rule menu. They are

1. the logistic rule: $L(x) = s \cdot x \cdot (1 - x)$
2. the uniform random rule,
3. the normal random rule,
4. the Voss rule, and
5. the Brownian rule.

In the logistic rule the parameter *s* is initially assigned the value 2.5. The uniform random rule uses a random number generator to determine the next term in the sequence, which will always be between 0 and 1. The normal random mode uses the same random number generator, but then calculates the value of *x* such that, in a normal distribution with mean 0.5 and standard deviation 3, the probability that the generated number will be less than *x*. The Voss rule is a crude simulator that approximates $1/f$,or pink, noise. The Brownian rule generates a sequence according to Brownian motion. Given the current term x_n in the sequence the next term is a randomly generated number in the interval $\left[x_2 - \frac{s}{2}, x_2 + \frac{s}{2}\right]$, where *s* is a pre-defined parameter. You can also make up your own rule through the "Homemade" option in the Rule menu.

Upon choosing the Time Series mode your screen should look like Figure 1. This mode works much like the Time Series mode in the **Iterate** program. The status window at the right side of the screen can be used to change settings. The first setting that can be changed is the parameter *s*. There are limits on the possible values of the parameters. For example in the tent map rule *s* must be no less than 0 and no greater than 2. In the logistic rule *s* must be between 0 and 4 and in the Brownian rule *s* must be between 0.01 and 1. In the other pre-defined rules the parameter *s* is not used in the creation of the sequence.

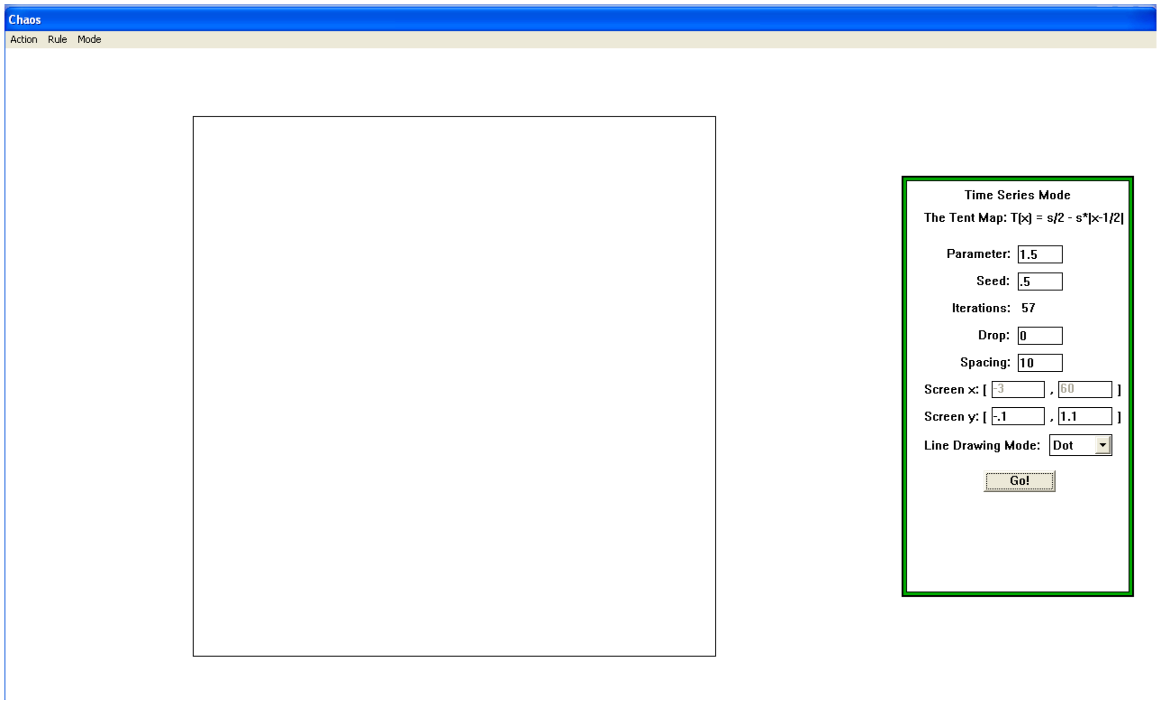

Figure 1

The seed edit box in the status window is the value of the first term of the sequence being generated. **Chaos** automatically determines how many time series points it can plot without running over the right side of the graph window. That number cannot be changed. The drop edit box indicates the number of iterations initially calculated before anything is plotted. When the entry in the "Spacing" edit box is *k*, then the vertical spacing between two successive time series points and $(n+1, x_{n+1})$ is *k* pixels. The default spacing is 10. Note that the screen *x*-interval cannot be changed, but the screen *y*-interval can be changed. The left endpoint of the screen *y*-interval gives the smallest value of x_n that will show up in the graph window, and the right endpoint gives the largest value. The "Line Drawing Mode" selection box offers three options:

1. Dot mode, where a small dot is plotted at each time series point $(n, \ x_n)$,
2. Single mode, where two successive time series points are connected by a straight line, and
3. Knot mode, which combines the Dot and Single modes. That is, a small dot is plotted at each time series point and two successive time series points are connected by straight lines.

Chaos automatically determines how many time series points it can plot without running over the right side of the graph window. When that number of points has been plotted, you are given three choices: 1) readjust the left and right boundaries of the graph window and plot more points, 2) trace the points currently showing, and 3) stop. Figure 2 shows a time series graph for the Brownian map with parameter 0.1.

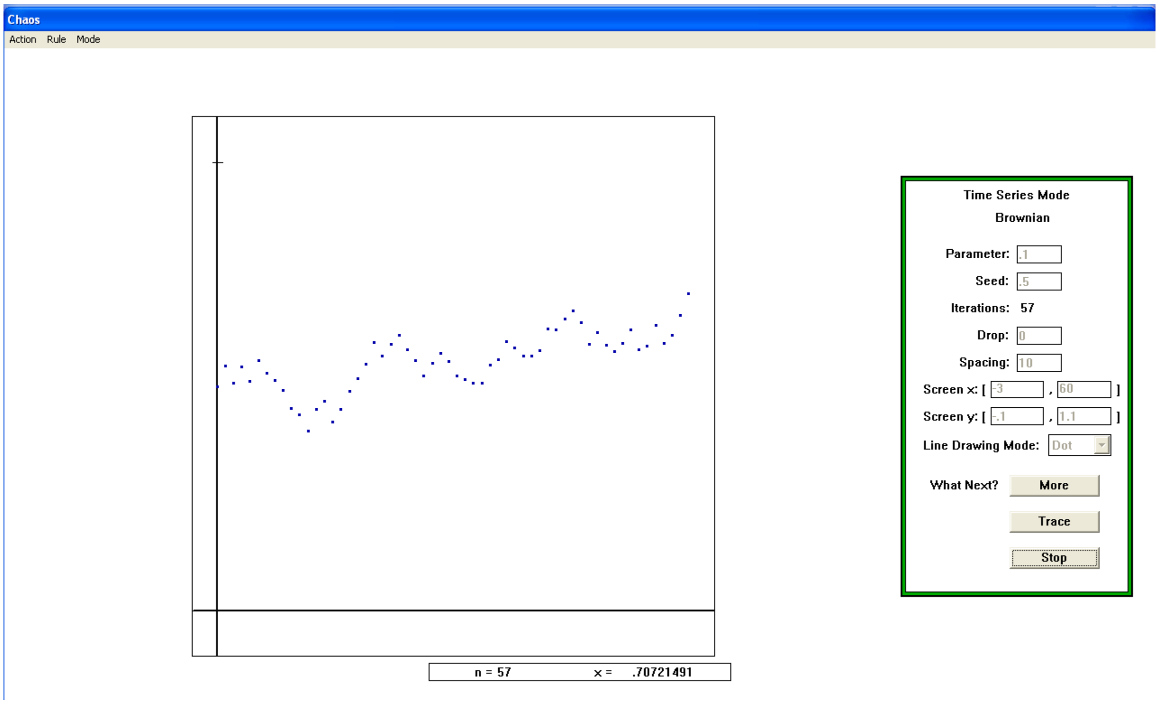

Figure 2

Next let's move on to the Delay mode. The graph that is plotted is a delay graph, where the points being plotted are $(x_n,\ x_{n+1})$. When this mode is selected your screen should look like Figure 3. The meanings of the options in the status window are the same as in the time series mode. Note that the "Spacing" and "Line Drawing Mode" options are no longer available. Also, unlike the time series mode, the number of iterations and the screen *x*-interval can be edited. Figure 4 shows a delay plot for the tent map running 200 iterations. Note that clearly all of the points plotted lie on the graph of $T(x) = \frac{s}{2} - s\left|x - \frac{1}{2}\right|$, so it is apparent from this delay graph that the sequence of points is not randomly generated.

In the Close Pairs mode, once a sequence of points has been generated, a point is plotted at the location (i, k) if $|x_i - x_{i+k}| < d$, where *d* is a pre-specified filter. Note that if x_i is near a *k*-cycle, then x_i is close to x_{i+k} and x_{i+1} will be close to x_{i+k+1}, x_{i+2} will be close to x_{i+k+2}, etc. The number of iterations performed is determined by the size of the point (dot) plotted. This leaves a recognizable signature in the Close Pairs plot; dots plotted at $(i, k), (i + 1. k), (i + 2, k), \cdots$ That is, an approximate *k*-cycle reveals itself as a horizontal band of dots. The dot size can range from 1 x 1 to 5 x 5 pixels. The number of iterations plotted is the number after the drop size. So for example, if the drop size is 50 and the dot size is 2 x 2 (hence, the number of iterations is 200), then iterations 51 through 250 will be plotted. The minimum *x* and maximum *x* values are only used in the Brownian rule and are the lowest and highest possible values in the generated sequence. Figure 5 shows a close pairs plot for the logistic map $L(x) = s \cdot x \cdot (1 - x)$ where the parameter s has the value 3.5 and with a 2 x 2 dot size. The scale marks on the axes are

20 iterations apart, and there are five horizontal lines between each scale mark. Therefore this sequence consists of (approximately) 4-cycles.

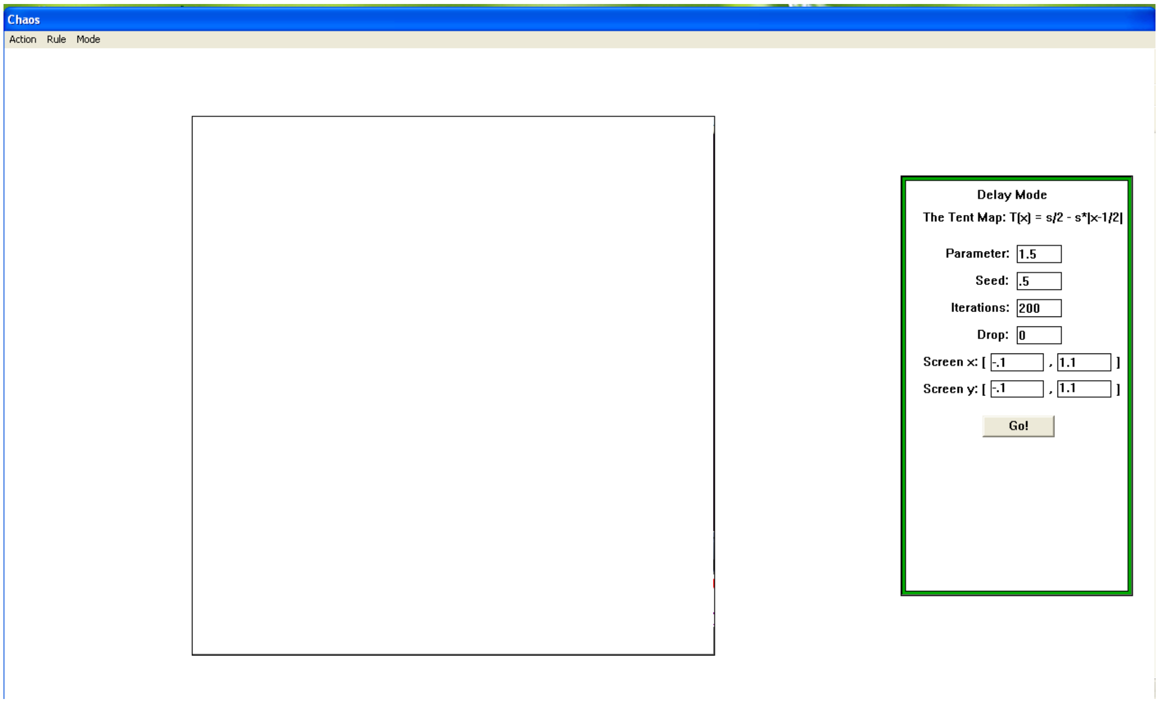

Figure 3

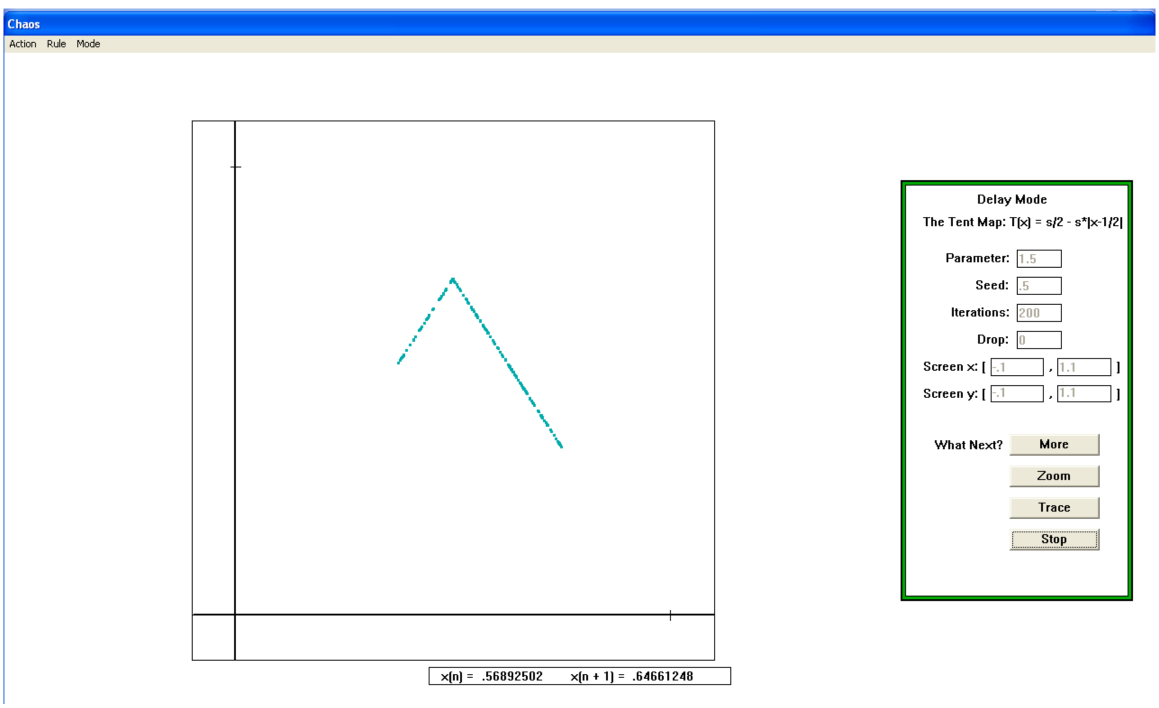

Figure 4

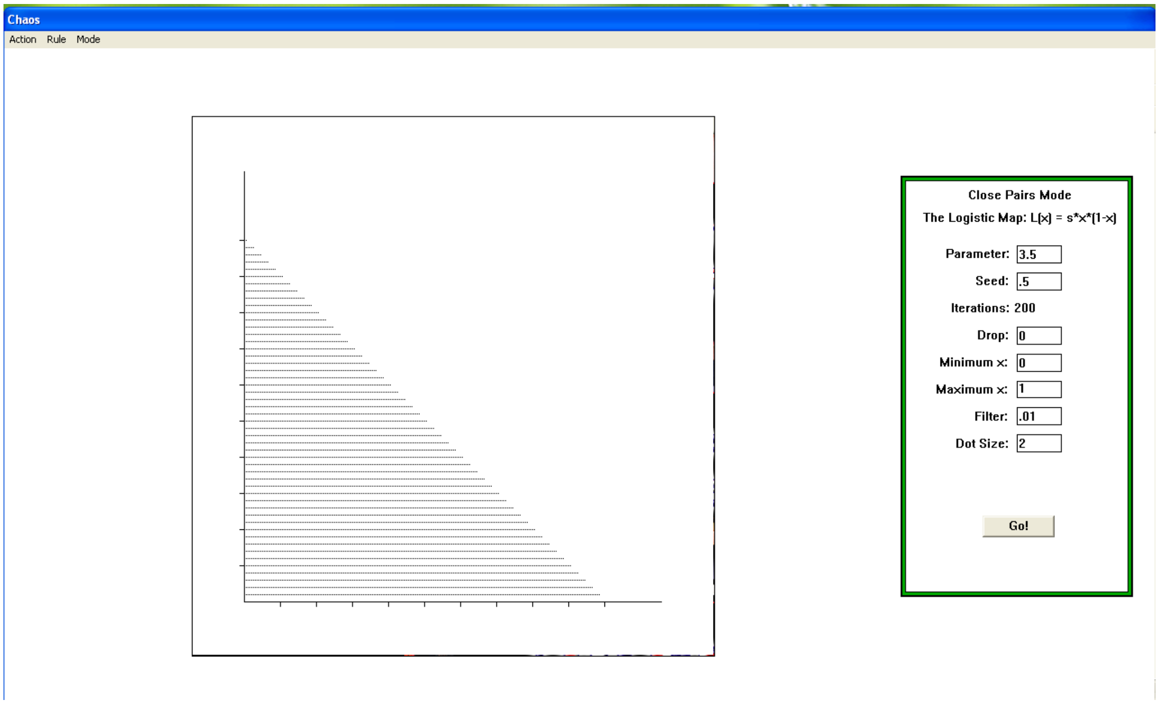

Figure 5

The next mode is IFS. The graph generated is related to the one created by the Four Corners Chaos Game in the **IFS** program. Here is how the graph is generated using default values of the parameters. Use one of the rules to generate a sequence of values, all between 0 and 1. For every entry in the original sequence, create a new sequence by the following rule:

- If the original entry value is equal to or greater than 0 but less than 0.25, the entry in the new sequence is 1, that is, the original entry value is located in Bin 1.
- If the original entry value is equal to or greater than 0.25 but less than 0.5, the entry in the new sequence is 2, that is, the original entry value is located in Bin 2.
- If the original entry value is equal to or greater than 0.5 but less than 0.75, the entry in the new sequence is 3, that is, the original entry value is located in Bin 3.
- If the original entry value is equal to or greater than 0.75, the entry in the new sequence is 4, that is, the original entry value is located in Bin 4.

Label the upper right corner of a unit square 1, label the upper left corner 2, the lower left corner 3, and the lower right corner 4. Start with a seed point in the center o the square. Graph a new point midway from the corner determined by the first entry in the new sequence to the seed point. Use the point so constructed as a new seed point and repeat the last step with the next entry in the new sequence. Keep on repeating these steps until 1000 points have been plotted.

When you select IFS mode your screen will look like Figure 6. The rule parameter (if applicable), the seed, iterations, drop, minimum *x*, and maximum *x* are all defined as in earlier modes and can all be modified by use of their edit boxes. In addition the boundaries between the bins can also be modified. Figure 7 shows an IFS picture made using the Normal Random (Gaussian) rule.

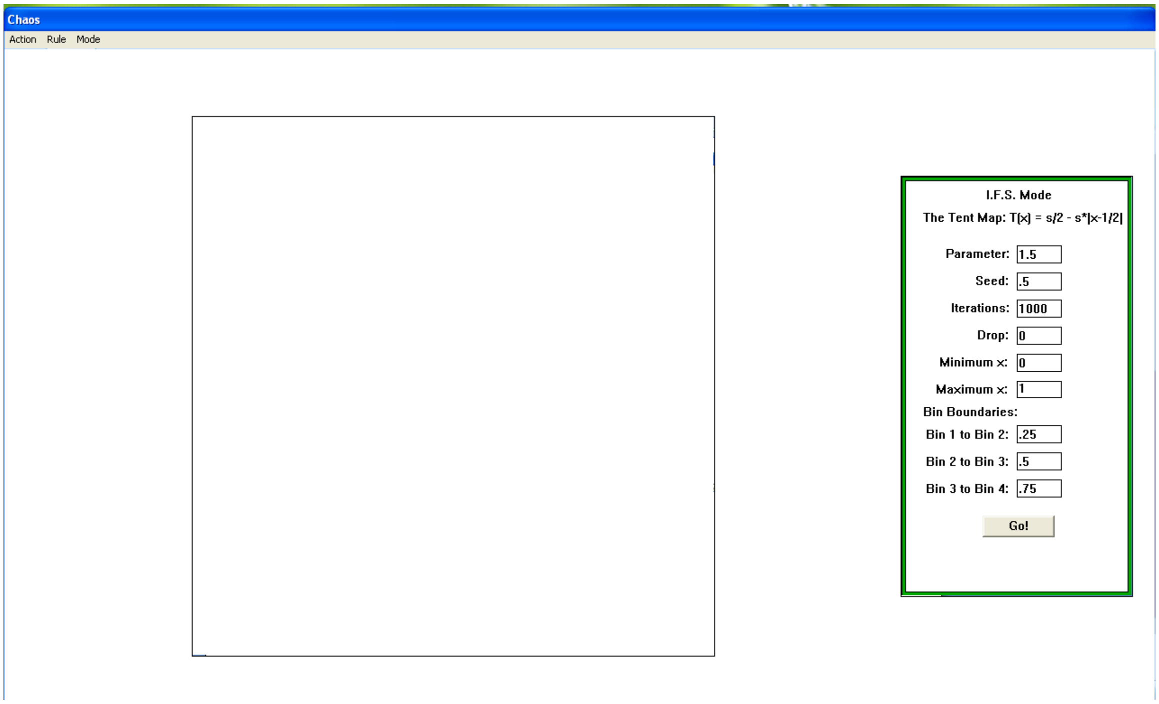

Figure 6

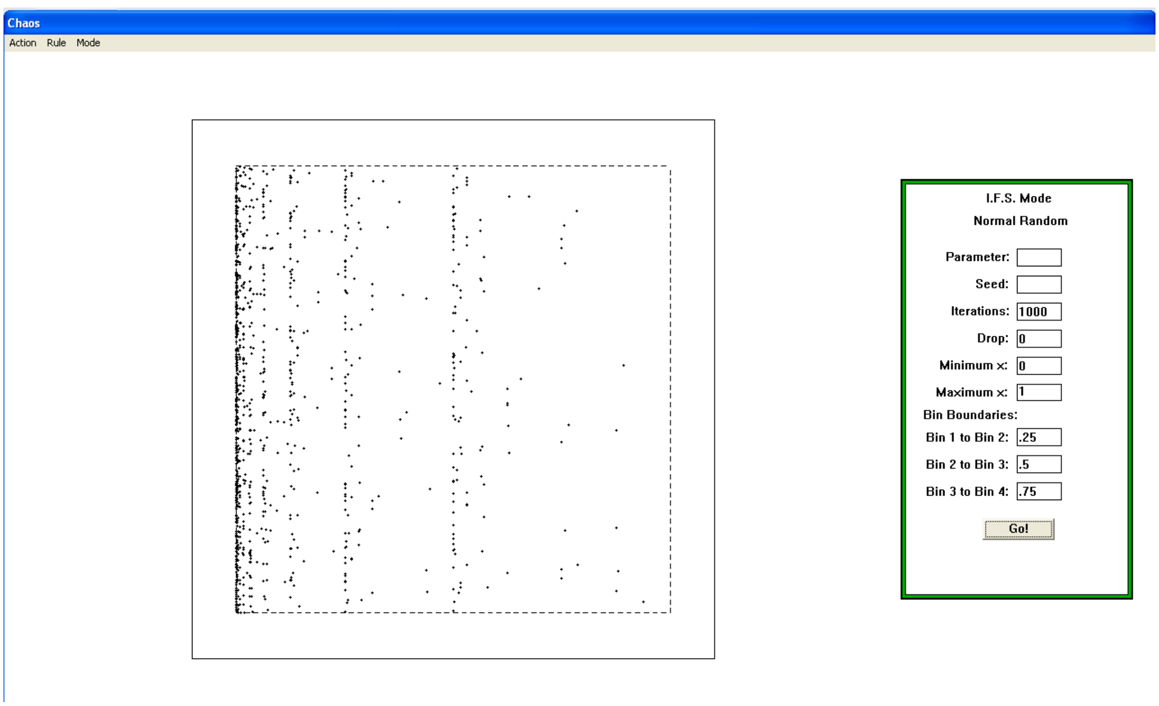

Figure 7

The final mode in **Chaos** is the Kelly mode. Here is how the graph is generated using default values of the parameters. Use one of the rules to generate a sequence of values, all between 0 and 1. For every entry in the sequence, assign it a bin (or color) according to the following rule:

- If the entry value is equal to or greater than 0 but less than 0.125, the entry is assigned to Bin 1, or the “Red” Bin.
- If the entry value is equal to or greater than 0.125 but less than 0.25, the entry is assigned to Bin 2, or the “Green” Bin.
- If the entry value is equal to or greater than 0.25 but less than 0.375, the entry is assigned to Bin 3, or the “Brown” Bin.
- If the entry value is equal to or greater than .375 but less than 0.5, the entry is assigned to Bin 4, or the “Magenta” Bin.
- If the entry value is equal to or greater than 0.5 but less than 0.625, the entry is assigned to Bin 5, or the “Blue” Bin.
- If the entry value is equal to or greater than 0.625 but less than 0.75, the entry is assigned to Bin 6, or the “White” Bin.
- If the entry value is equal to or greater than 0.75 but less than 0.875, the entry is assigned to Bin 7, or the “Yellow” Bin.
- If the entry value is equal to or greater than 0.875, the entry is assigned to Bin 8, or the “Cyan” Bin.

A 15 x 15 grid of squares is displayed, and the grid in the top left is colored according to which bin the first entry in the sequence is assigned. The square to the immediate right is colored according to which bin the second entry in the sequence is assigned. The squares in the rest of the first row are colored according to which bins entries 3 through 15 in the sequence are assigned. The squares in the second row are colored according to which bins entries 16 through 30 are assigned, etc.

The parameter (if applicable), seed, drop, minimum *x*, and maximum *x* are all defined as in earlier modes and can all be modified by use of their edit boxes. In addition the grid size and the boundaries between the bins can also be modified. Figure 8 shows the Kelly diagram that is generated using the Tent map rule with parameter 1.75.

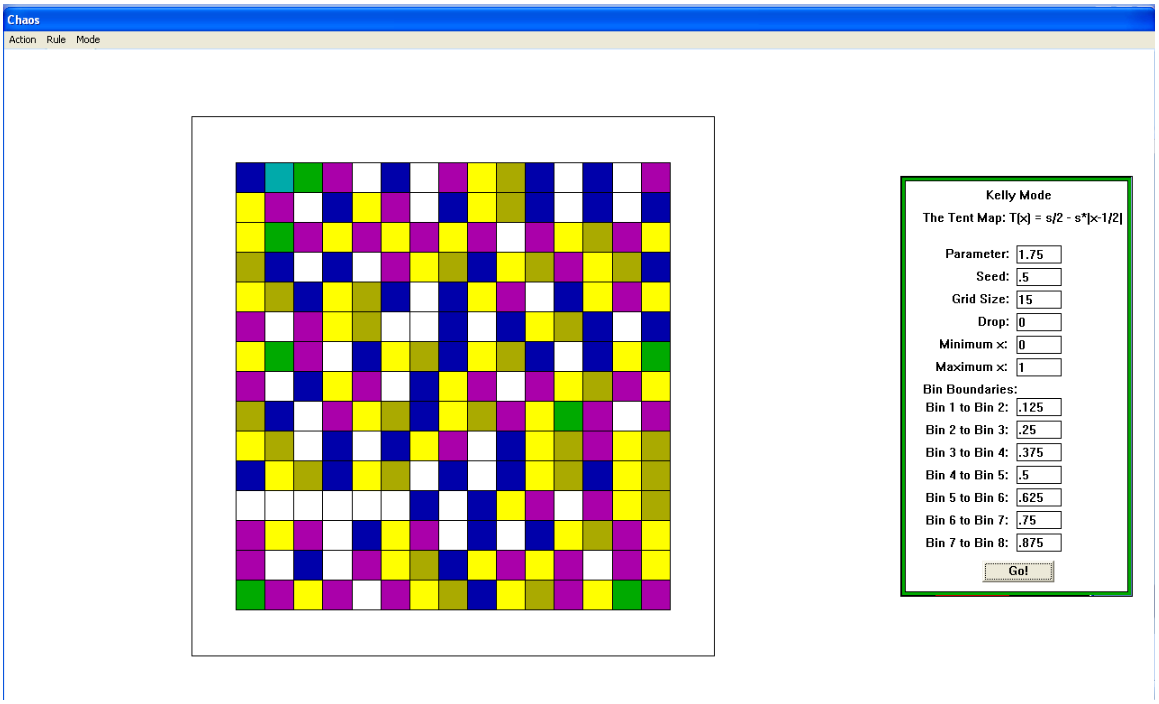
Chaos
Action Rule Mode
Kelly Mode
The Tent Map: T[x] = s/2 - s*|x-1/2|
Parameter: 1.75
Seed: .5
Grid Size: 15
Drop: 0
Minimum x: 0
Maximum x: 1
Bin Boundaries:
Bin 1 to Bin 2: .125
Bin 2 to Bin 3: .25
Bin 3 to Bin 4: .375
Bin 4 to Bin 5: .5
Bin 5 to Bin 6: .625
Bin 6 to Bin 7: .75
Bin 7 to Bin 8: .875
Go!

Figure 8