Characterizations of Stochastic Agent Movements

in CovidSIMVL

by the authors

December 13, 2020

Introduction

The agent-based model CovidSIMVL [references] uses simulated agent movement in a finite geospatial 2D arena in which transmissions between infected and susceptible agents may take place if their representations overlap. The main factors that determine the dynamics of contagion are: the size of the agents (susceptibles and infectives both), the mingleFactor, which governs the stochastic distance between successive moves, and the density of agents within the defined arena of 800x60 pixels.

The temporal dynamics for viral growth and infectivity are adopted from the paper by Xi, He [ ] in Nature Medicine, April 2020. These in turn affect the size of infectives (viral load implies infectivity which implies larger size), while risk factors and mitigations (masks, etc) affect the size of susceptibles (the smaller the less likely to meet a transmitter agent). The movement dynamics reflect the social roles of agents (waiters move around more than diners), the effects of social distancing, and the relative movements of agents in different settings (dance floors versus classrooms).

Previous papers in the CovidSIMVL series [ , ] have explored the relationship of size and mingle factor to R0, to the intensity of an epidemic; the characterization of transmission paths and transmission trees to epidemics, and the time-dependent flow and ebb of simultaneous transmissions as saturation is reached.

In this paper, some characterizations of the stochastic movements of agents are explored, and their relationship to the likelihood of contact between infective and susceptibles.

METHODOLOGY

The CovidSIMVL program has been adapted to maintain on-screen the semi-transparent image of a single agent following a stochastic move, one per generation. It uses two .csv files, one defining a population, total of one, and the other defining an infected agent, which in this case is the same agent.

At the end of each 25 generations, a screen shot is captured. At the same time, the console.log of CovidSIMVL produces the following information at each generation:

generation

Hazard Radius (initial size of the agent)

X,Y coordinates of the agent in the 800x600 arena

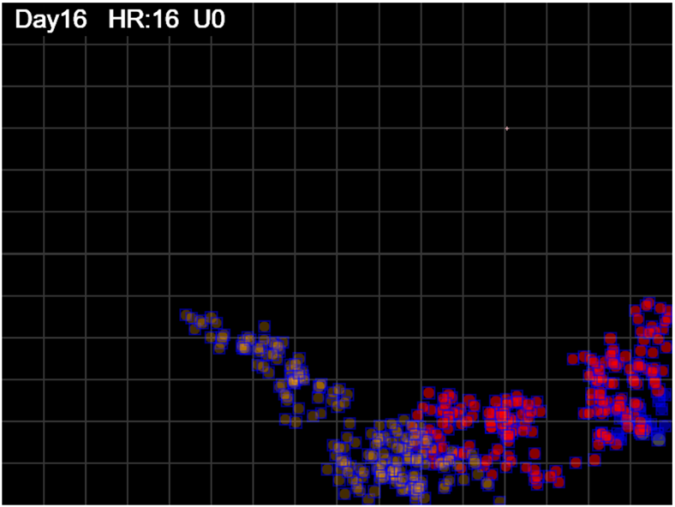
footprint the total number of pixels that agent imprints have made

sectors the number of 50x50 sectors that the agent has visited

The following is a screen capture at generation 400 of a trial with mingleFactor (“mF”) and a Hazard Radius (“HzR”) of 5, and a section of a representative console.log (not same trial). The screen shows D16H16 which is 16x24+16 or 400 generations (one per hour).



The Appendix contains many trials, with 20 screen captures per trial, and the data metric derived from the console.logs, as well as the derived charts showing the characteristics of the agent stochastic movements.



At this point, we will take this example, and work through the derivation of the metrics and charts that show the relationships we seek.

Some definitions:

1. The Footprint of the agent at time T=[gen] is the count of pixels which have been touched by the agent within a certain period. In the above, it would be from generation 0 to 400.

The computation of this is through the enclosing rectangle of a circle at (x,y) with radius r, which would be 2r x 2r, with the area of the circle being pi/4 X [rectangle area] = pi/4 x (4r\*\*2).

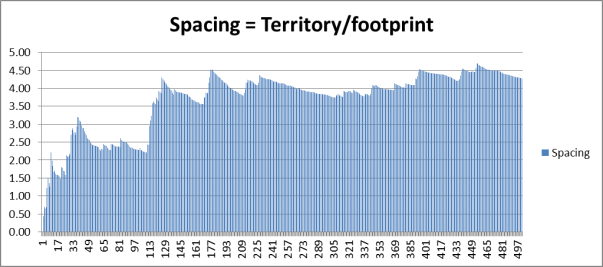
We mark the pixels in positions (x-r,y-r) to (x+r,y+r) as each movement is made, then find all such marked pixels in the arena and apply the adjustment pi/4 to the total. Pixels that have been visited several times keep the tally, though this intensity data is reserved for future studies.

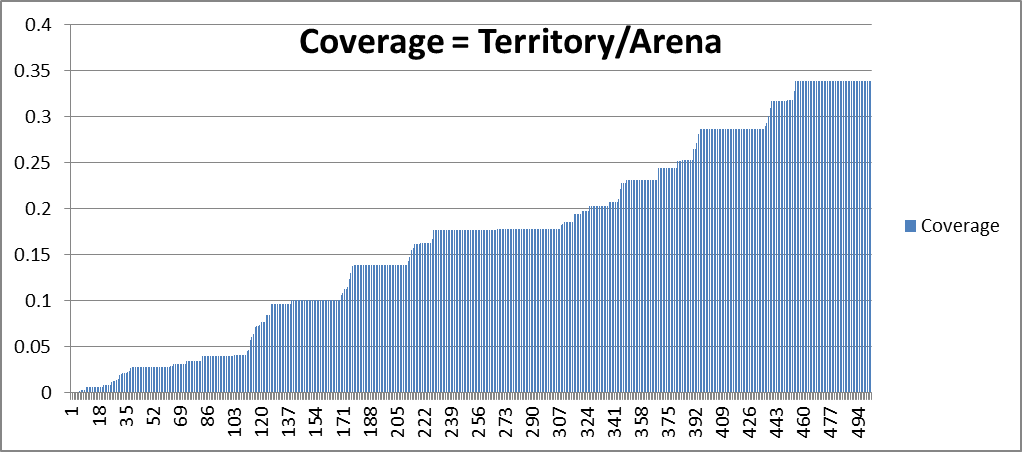
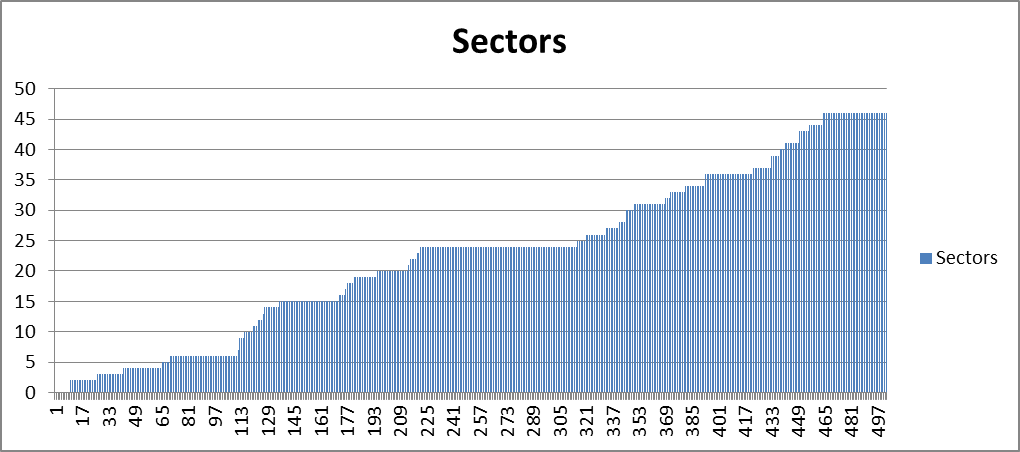
1. The Territory of the agent at time T=[gen] is the area within the bounding rectangle defined by the min(x,y) to max(x,y) for all positions (x,y) of the agent, within a certain period of time. This is the calculation (maxX – minx)\*(maxY-minY).
2. The Spacing of the agent movement to time T=[gen] is the ratio of the territory to the footprint. The larger the spacing, the more area of the bounding rectangle still unvisited, and the sparser the distribution of the footprint elements. Conversely, the smaller the spacing, the denser the footprint elements within the territory.
3. The Coverage of the agent movement to time T=[gen] is the ratio of the territory to the entire arena, which is 800x600 or 480,000 pixels. Since this ratio is always territory/K where K is the constant 480,000, it is obvious that the shapes of the territory values vs time and the coverage vs time are identical.
4. The Sector Count of the agent movement to time T=[gen] is the number of 50x50 sectors that have been visited by the agent as it makes successive moves, in a certain period of time. The 800x600 arena has 16/12 sectors = 192 of them.

The following is a portion of the table of calculations which includes data from the console.log as well as derived calculations, in sequence order of generations.



From this table (see Appendix), we create the following charts for Spacing, Coverage and Sectors.

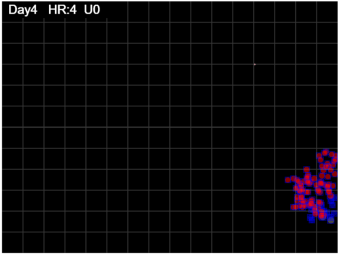
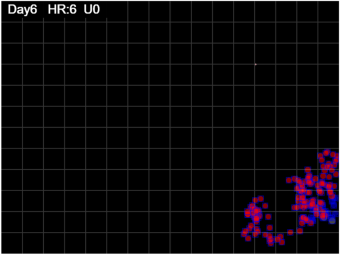


When there is a sharp rise in Spacing, the Territory has expanded quickly, which implies that a new position (x,y) beyond the bounds of the current Territory. A drop in spacing means an increasing footprint *within* the Territory, which implies that the agent is moving about, but not outside the existing Territory.

Similarly, the Coverage increases (here it is no more than 0.4 or 40% of the arena) when the agent moves beyond existing boundaries. In the time span from T=1 to T=500, the Territory would keep increasing, and a sharp increase again means the agent has made a significant move outside the existing boundary.

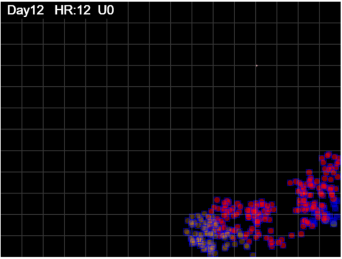
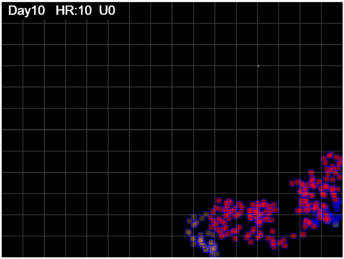
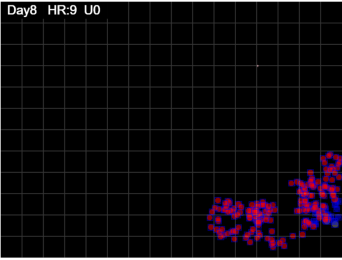
Sector count increases should follow increases in Territory, where sectors are blunt measures of 50x50 blocks while Territory are measured in pixels. Flat lines in Coverage and Sectors mean no change for those generations – the agent sticks to the Territory.

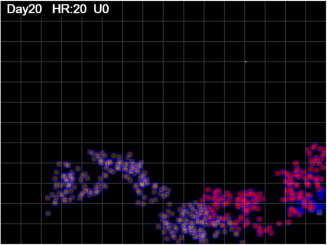
We can show here all the screen captures for this example, in 50-generation steps.

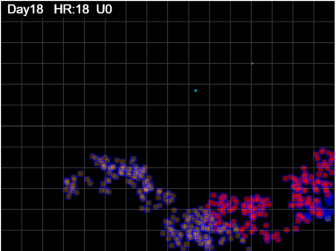
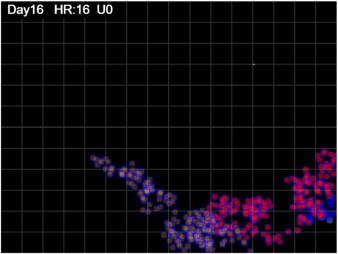
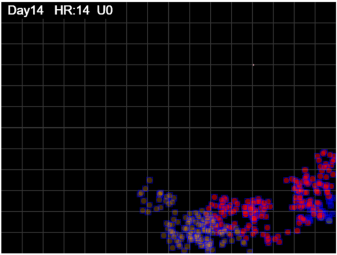


a breakout 100-150

gen 50 gen 100 gen 150



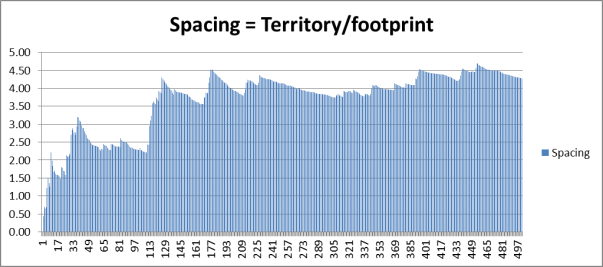
 gen 300 gen 250 gen 200



gen 350 gen 400 gen 450 gen 500

We can see that the agent movements are clumped at the right side, then begin to spread toward the bottom and left, fairly densely, but at gen=400 appears to create a tendril which gets another cluster.

These breaks should be visible in the Spacing chart, as the Territory expands while the Footprint is slow to follow.

All these peaks can be seen as

corresponding extensions of the Footprint and Territory of the agent in the time frames.

This instantiation of the mF=1 and HzR=5 gives a pathmap at gen=500 of 35% of the arena covered by the Territory, within which the Spacing is between 2.0 and 4.5, within which there is a jump in expansion of Territory between gen=100 to gen=240.

We can at this point see whether these Spacing and Coverage metrics are approximately the same for mF=1 and HzR=5, varying as we change mF and HzR. Before we do that, we can complete the examination of this trial, to study the most recent changes in Territory and Spacing, at any given time.

25-Generation Recency

The metrics above were calculated using the min-max(xy) for the entire duration from Gen=1 to Gen=current ending at 500. Thus, the Territory grows, the Sectors grow, with the passage of time. The question can be asked: what happened recently, say in the last 25 generations? Did the agent move in an expansive or a constricted manner; was the 25-generation Territory going back from *now* larger or smaller than the corresponding Territory of 25 generations ago? Similarly for Spacing, and for Sectors travelled.

The reason for this is to examine whether the stochastic nature of the movements produce a smoothly varying composite Territory, or whether discontinuities exist. It is not clear that from any given point of time T=[gen] we can predict the future path, but knowing the most recent parameters of the paths may give us an insight into the total, and beyond.

The first thing that has to be done is to calculate the min-max(xy) for the most recent 25-generation set of movements, rather than for the entire series from gen=1 to gen=now.

We also need the Footprint for that period alone, which we do by finding the difference between Footprint values from T=[i] to T=[i-25], so that we are in the same time frame. These are shown in the partial table below.



The last line is for GEN=75, and the values of min-max(xy) are for the values of X and Y just for the time interval from GEN=50 to GEN=75. Thus the “territory-25” is computed from the bounding rectangle for just those points.

The column “del footpt” is calculated by taking the difference between the footprint at GEN=75 to the footprint at GEN=51 which is 6898.93 – 5315.57 = 1583.36.

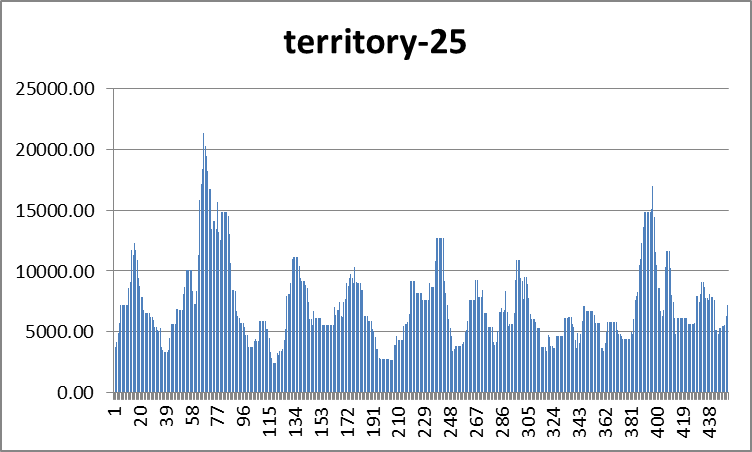
Similarly, the column “del sectors” is the change in total sector count between the value at GEN=75, which is 6, and the value at GEN=51, which is 4, so the difference (sectors gained in the 25-gen time frame) is 2.

The value for “territory-25” at time GEN=75 is 6504.55, which is the territory generated by the last 25 moves, from GEN=51 to GEN=75. Therefore, the change (delta) for territory when comparing territory generated to GEN=75 to that at GEN=51, is 6504.55 – 4177.55, which is 2327.00. This calculation allows us to see that the most recent moves may be quite tight, while the ones before may be loose (larger territory), so that the change could be negative, if the time-based territories are smaller, or positive, if they are increasing.

Finally, “del Spacing” is calculated the same way. Since Spacing is the degree of sparseness of the bounding rectangle of the footprint within a certain time frame (GEN=51 to GEN=75), then at GEN=75, looking back at the last 25 moves, there is a Spacing for those moves at GEN=75. Thus, we can compare this Spacing Index to that for the 25 moves preceding GEN=51, and this difference is the delta for Spacing.

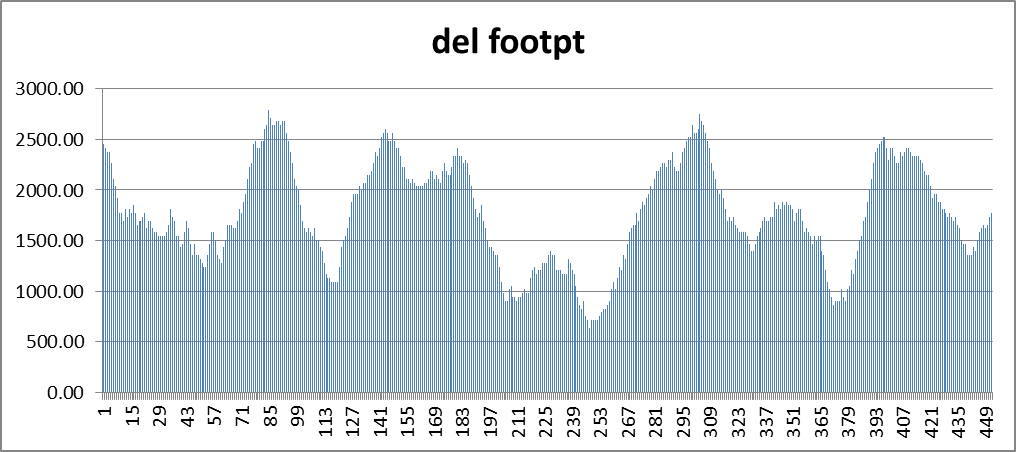
In this case, we calculate delta Spacing by taking the (delta Territory/delta footprint), where the delta Territory is for the 25-move Territory values at GEN=75 to GEN=51, and likewise for delta footprint. The value at GEN=75 is therefore 2327.00/1583.36 = 1.47.

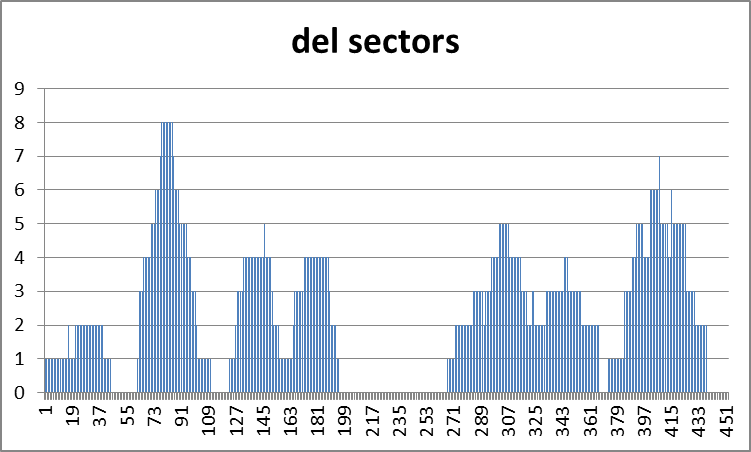
These metrics can be understood by seeing their charts, as follows.



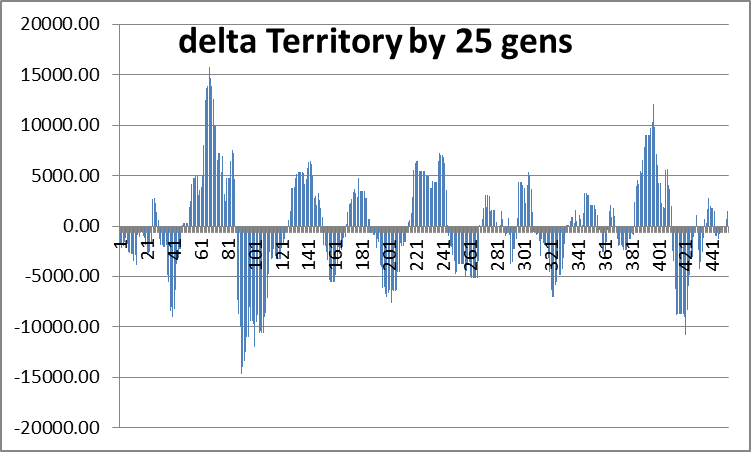
This is the chart of the territory defined by the previous 25 moves, at any T=[gen] position. Instead of the total territory at T, it is how the recent 25 moves behaved in terms of the territory boundaries.

Whether tight, or expansive, they are always positive. In this chart, we can see many positions at which the most recent territories expanded and was thus larger than its neighbors. When declining, the moves were more local.

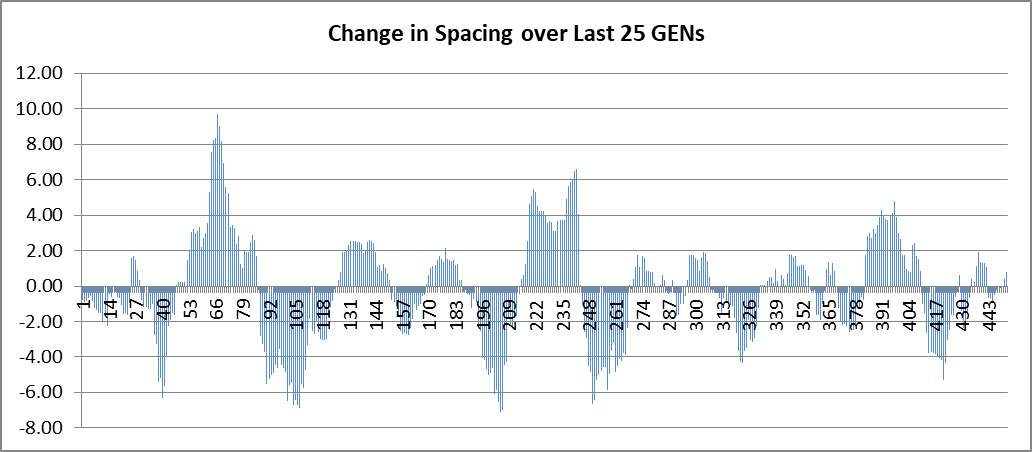
This chart shows the totality of pixels visited by the last 25 moves at any time T=[GEN]. When they are increasing the interpretation is that there is less overlap, so the number of discrete pixels is larger. When they decrease, it is because many moves cover the same pixels, so the count is less. This will happen when moves are tight, so low “del footpt” are in clustered footprints.

This chart shows the change in sectors that were visited by the last 25 moves. When this number is large, or rising, it indicates that the at T=[now], the sector count for the sectors visited in the last 25 moves is bigger than the same count for T=[now-25.

A large displacement move will lead to an increase in Territory and an increase in delSectors. This metric decreases as moves are more clustered tightly.

When we compare the change in territory generated by the last 25 moves for T=[now-25] to T=[now], we can see that the changes can be positive (more territory) or negative (change is to a smaller clustered territory).

In this trial, we see a number of positive and negative clusters. These are alternating, and they are large. The change of 15,000 pixels can be 300x50 for example, or 60x250. The pattern is one of growth, followed by clustering, followed by growth, in alternation. This is unexpected, and we will see whether this is repeated in other trials. The stochastic move generation algorithm for CovidSIMVL does not have memory of a previous move to lead to progressive increments and decrements within changes in displacement.

As discussed above, Spacing is Territory/Footprint, so the 25-frame Spacing change from T=[now=25] to T=[now] can be both positive and negative. The metric will be positive when the sparseness increases in high-displacement moves, and negative when sparseness decreases as moves get tighter. Thus, delta-Spacing tends to follow delta-Territory in this trial.

RESULTS

We have run trials for mF=1 and HzR=5, for mF=3.2 and HzR=5, and for mF=0.4 and HzR=10. For the first parameter set, we have run five separate trials. The results of these trials will be tabulated below. Before we contemplate comparing these charts, we will highlight the most important attributes, which are: the progression using steps of 100 generations, of:

1. The growth of overall Territory from T=1 to T=500
2. The Spacing within the overall Territory from T=1 to T=500
3. The growth of the overall Footprint from T=1 to T=100