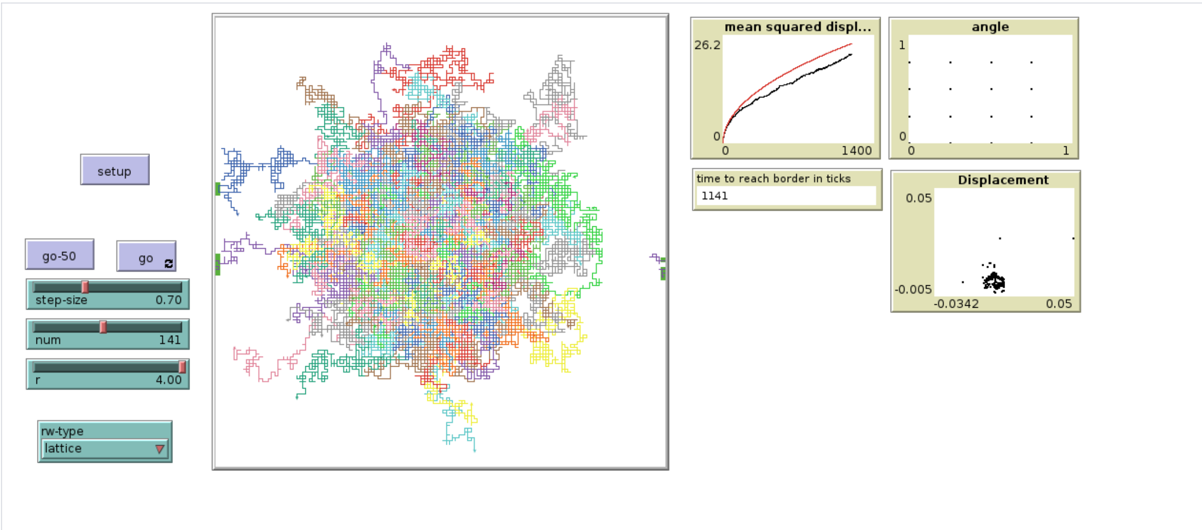
https://drive.google.com/drive/folders/102-emHzKUgnBG25JCuXIzEgCLy66Lpoz?usp=sharing result files with csv data used for the further visualization

Lattice Random Walk

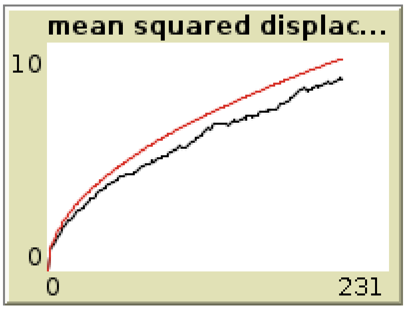


(pic1)

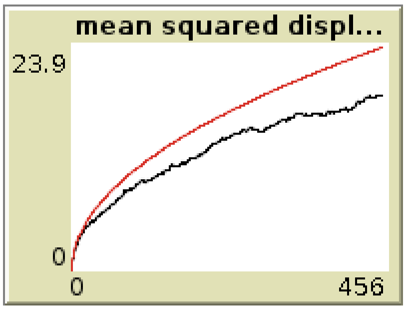
pic1.4.1 Mean Squared Displacement

This plot is quite useful, because the red line represents the theoretical mean squared displacement of all turtles over a period of time, and the black one represents the real one, we really get. As we can see from the graph - the real one is quite close to the theoretical one.

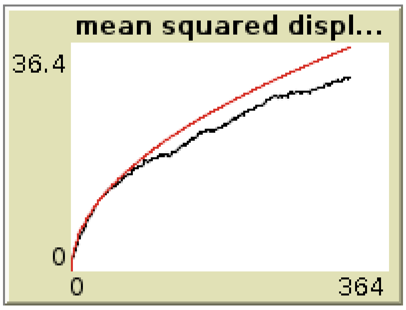
It also makes sense to try using different parameters in order to compare how the difference between these two curves will change



step-size 0.66, num_turtles 122

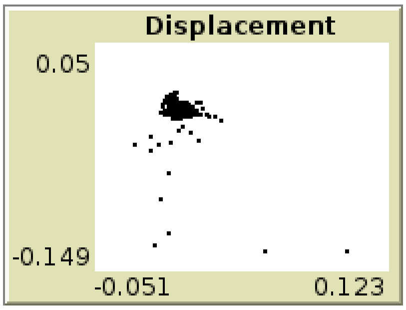


step-size 1.11, num_turtles 122



step-size 2.00, num_turtles 122

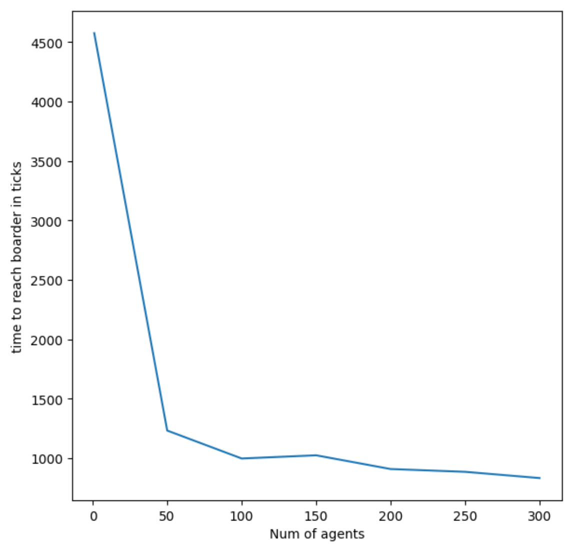
As we can see from the above pictures - it is all quite similar - the curves are close enough though they never coincide.
The reason why they are not the same is because in this model the angle for each turtle on every tick is set in a random choice between one of four angles (0, 90, 180, 270), therefore we can not guarantee the exact MSD over some particular periods of time. Still we can expect it to be close to the theoretical MSD, because of a randomness factor.

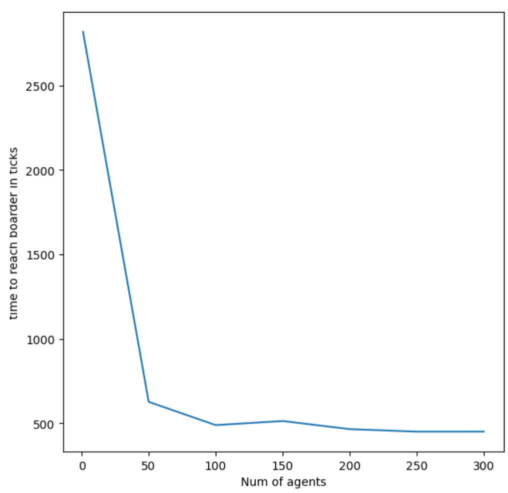


I depicted displacement in dots - which is actually very useful, because proving the point above - it all goes up to zero, because the final displacement should give us zero as a result of hundreds of ticks and random angle choices between one of (0, 90, 180, 270 degree). Even if we have only one turtle - the probability that the next angle won't change the direction of turtles movement is P(x-1)* 1/4 where P(x-1) is the probability that all angles before this one haven't changed a thing. Therefore as the turtle moves it creates new movement vectors, sum of which finally will be close enough to zero. This is why dots are being so much gazered at such a small x and y respectively.

pic1.4.3 Time to reach the border in ticks
This is the model which always definitely sooner or later reaches the border, even though its movement angles are being chosen randomly between one of (0, 90, 180, 270 degree). And

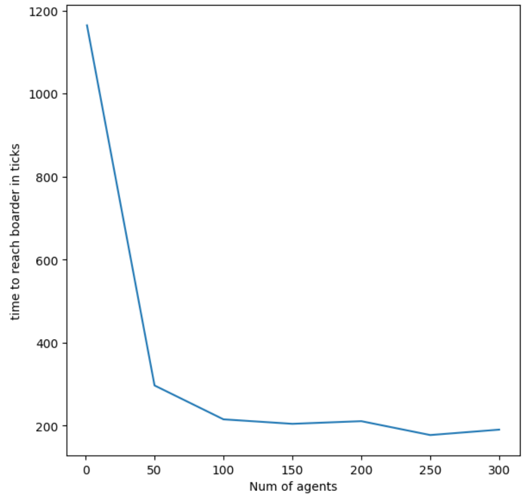
it actually makes a lot of sense why with the bigger number of turtles it is much easier and faster to reach that border. While investigating the number of agents parameter - I got these results:

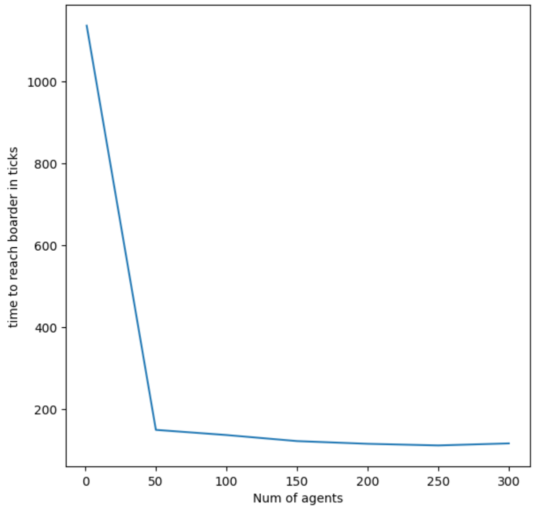




pic1.4.4 Angle

In case the step-size 0.7, 1.0





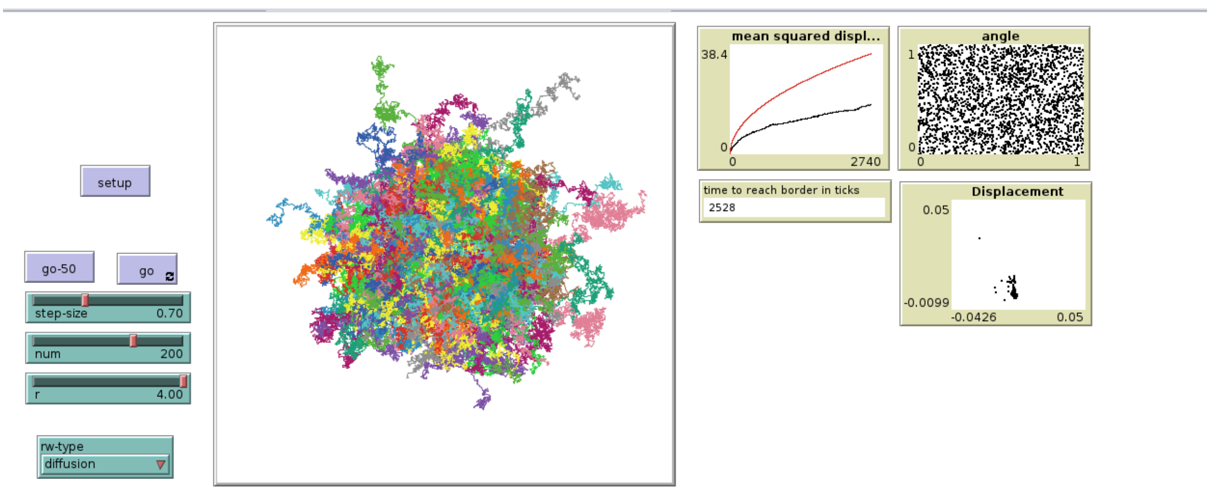
and 1.5 and 2.0 respectively

As we see on these graphs both step-size and number of active agents play a vital role when it comes to the time to reach the border. The more agents we have -> the higher the probability it is that one of them will hit the border and given the fact that all of the agents have the same probability to hit it, the increase of their number increases the probability. The analogical conclusion may be made for the step-size increasing. The bigger it -> the more probability that one of the agents will move far enough to hit the border.

pic1.4.4 Angle

We also may find this graph quite useful as it depicts the change of angles. As we can see - for this random walk, which angle only changes between one of 4 angles - (0, 90, 180, 270) therefore the final changes would be in one of the depicted points.

Diffusion Random Walk

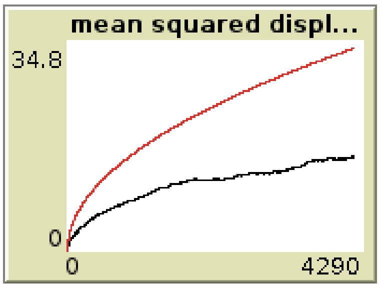


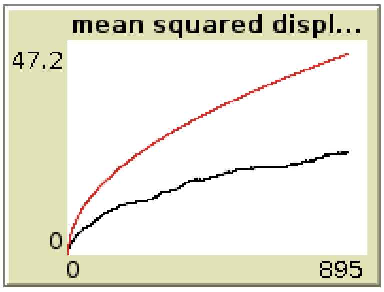
pic1.4.1 Mean Squared Displacement

This plot is quite useful, because the red line represents the theoretical mean squared displacement of all turtles over a period of time, and the black one represents the real one, we really get, all the same as in the previous model. As we can see from the graph - the real one is not as close to the theoretical one as in the previous model, but still close.

It also makes sense to try using different parameters in order to compare how the difference between these two curves will change

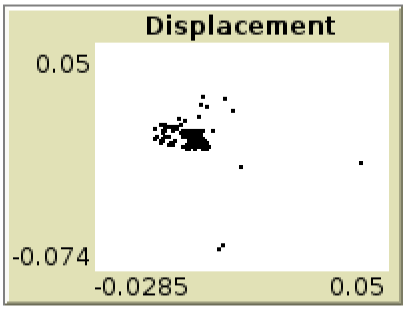
pic1.4.1.1
step-size= 0.52, num turtles = 1.52, step-size= 1.52, num turtles = 1.52,





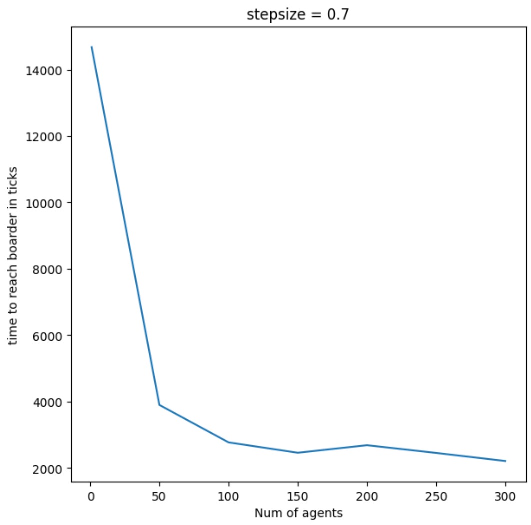
As we can see from the above pictures - it is all quite similar - the curves are not close, but the shape is quite similar.
The reason why they are not the same is because in this model the angle for each turtle on every tick is set in an absolutely random way, therefore we can not guarantee the exact MSD over some particular periods of time. Still we can expect it to be close to the theoretical MSD, because of a randomness factor.

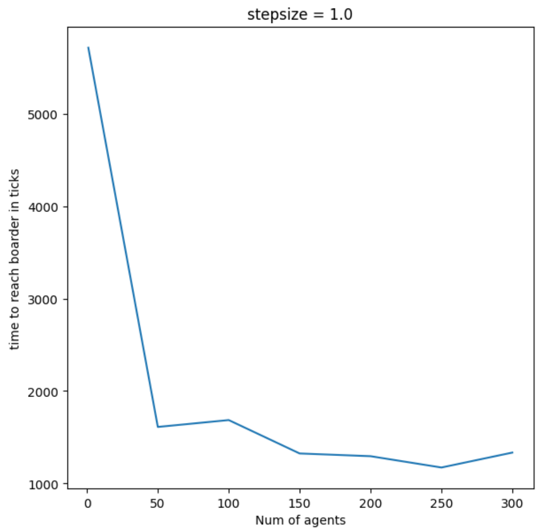
pic1.4.2 Displacement



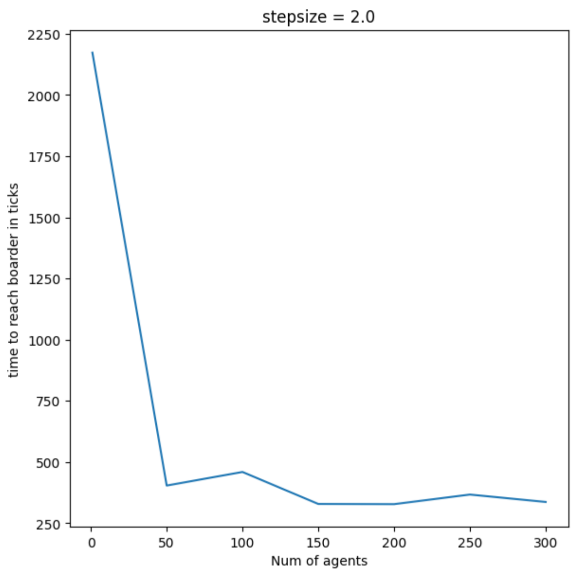
Here the displacement is expected to give the same result the previous model gave us. only the step-size is defined for different runs, although the angles are being totally randomly chosen, which gives us the same conclusion as in the previous model - why does this graph look the way it does? (i.e. gazers dots in some place around 0)

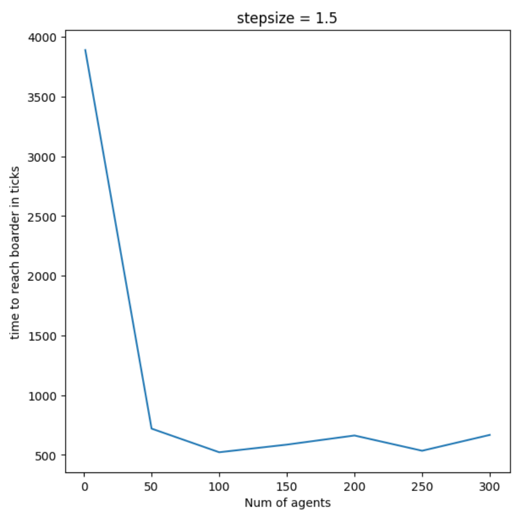
pic1.4.3 Time to reach the border in ticks
This is the model which always definitely sooner or later reaches the border, even though its movement angles are being chosen randomly. And it actually makes a lot of sense why with the bigger number of turtles it is much easier and faster to reach that border. The conclusions here - why it takes more agents and bigger





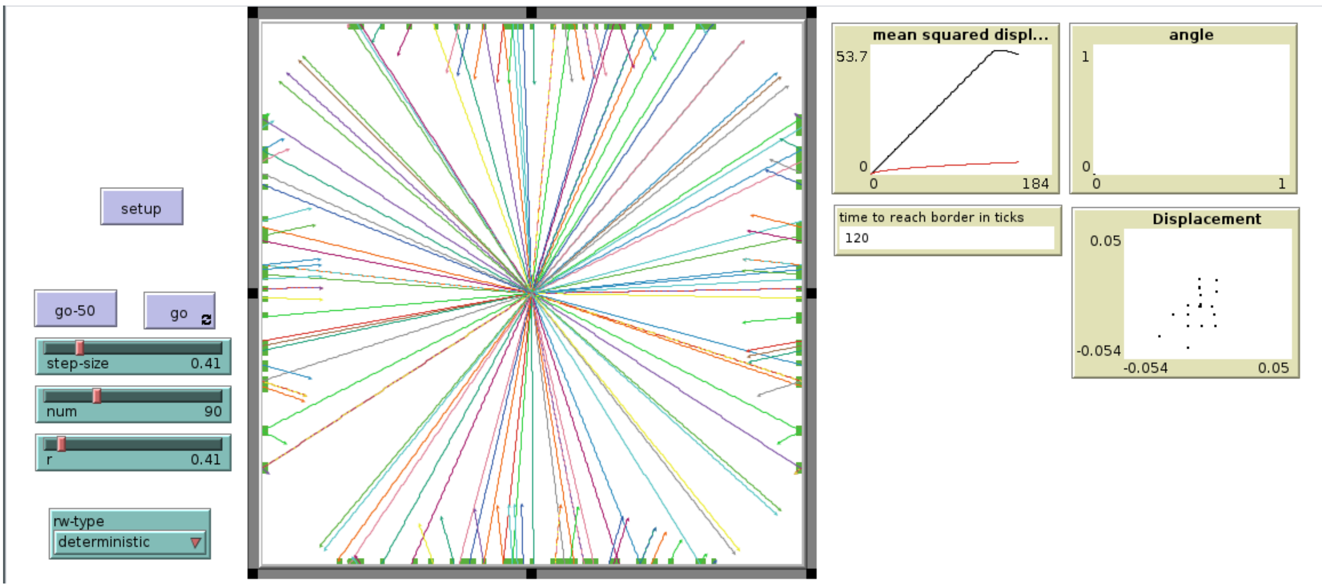
step-sizes to reach the border faster are the same as for the previous model. While investigating i got these results:





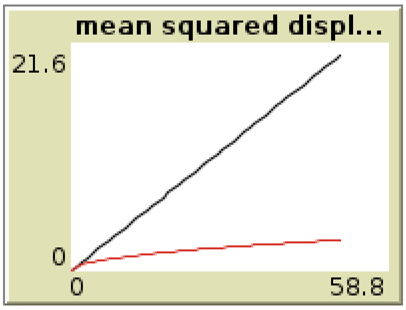
The pikes sometimes mean the randomness of the model angles choices. this is why even larger amounts of agents with bigger step-sizes may still take more time to reach the border than the smaller amount.

__Deterministic Random Walk__

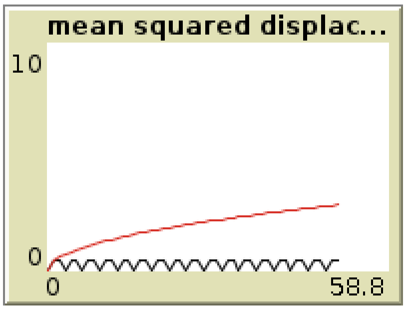


pic1.4.1 Mean Squared Displacement

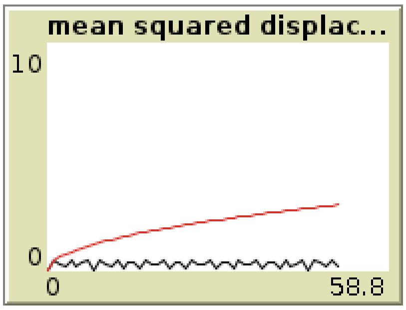
Same plot as in two other models, although here the MSD behaves quite differently.



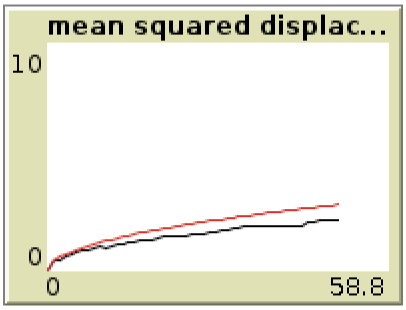
params lambda = 0.61, num = 90, step-size = 0.41



params lambda = 1.52, num = 90, step-size = 0.41



params lambda =2.65, num = 90, step-size = 0.41



params lambda = 3.59, num = 90, step-size = 0.41

If we take into account that this

*Angle = lambda * current-angle * (1 - current-angle)*

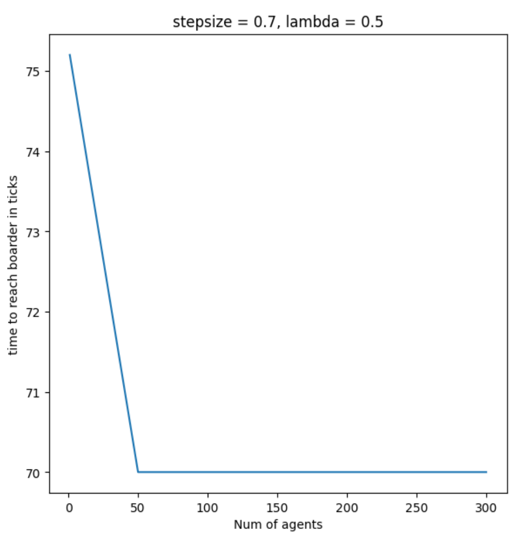
is the equation for this particular random walk angle choice, then the explanation is quite obvious. In the first situation, the lambda is below 1, therefore it will make the whole new angle smaller than it was before, this is why the difference is so big between the theoretical and actual curves.

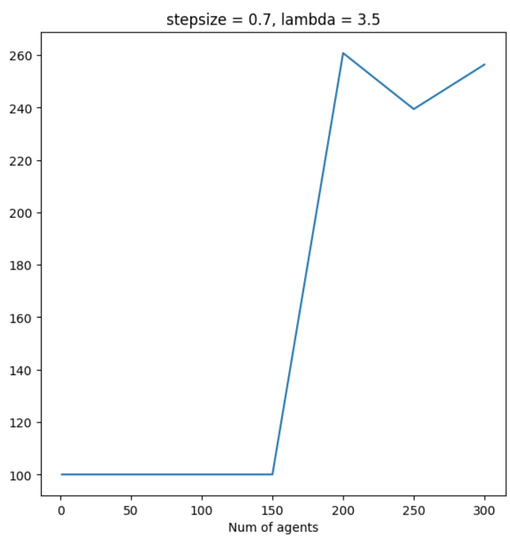
Second third situations are quite close. This can even be seen from the graphs themselves, the change between new angle and current angle will be changing between one of two numbers, this is why the curve is so wavy. Same for the third graph, but the curve is a bit biased due to the lambda value.

pic1.4.2 Displacement

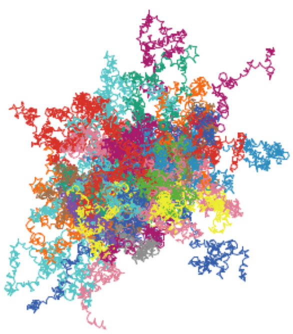
The displacement works here the same as with the previous two models, so I will not stop on this one. It behaves quite the same independently of lambda, number of turtles, world size .. etc.
It gazes the dots in one relatively small point (i.e. both x and y are very small numbers), which leads to the same conclusion that the total sum of all vectors is zero or very close to it.

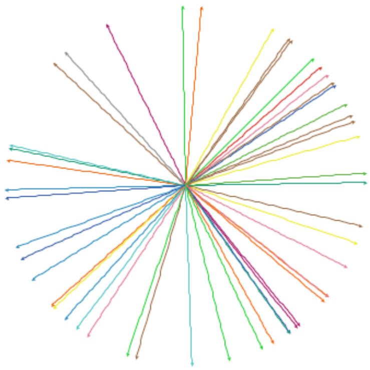
pic1.4.3 Time to reach the border in ticks
This model is quite tricky in reaching the border. If first and last case (i.e. the lambda is either below 1 or above 2.5 (+-) - it allows the model to reach the model, because it soesnt get stuck circling inside the matrix. There are tightly bonded with the graphs depicting MSD. We can take a parallel here - that i first and last case - these are the time dependency graphs:





When the generated models for these situation look like this:



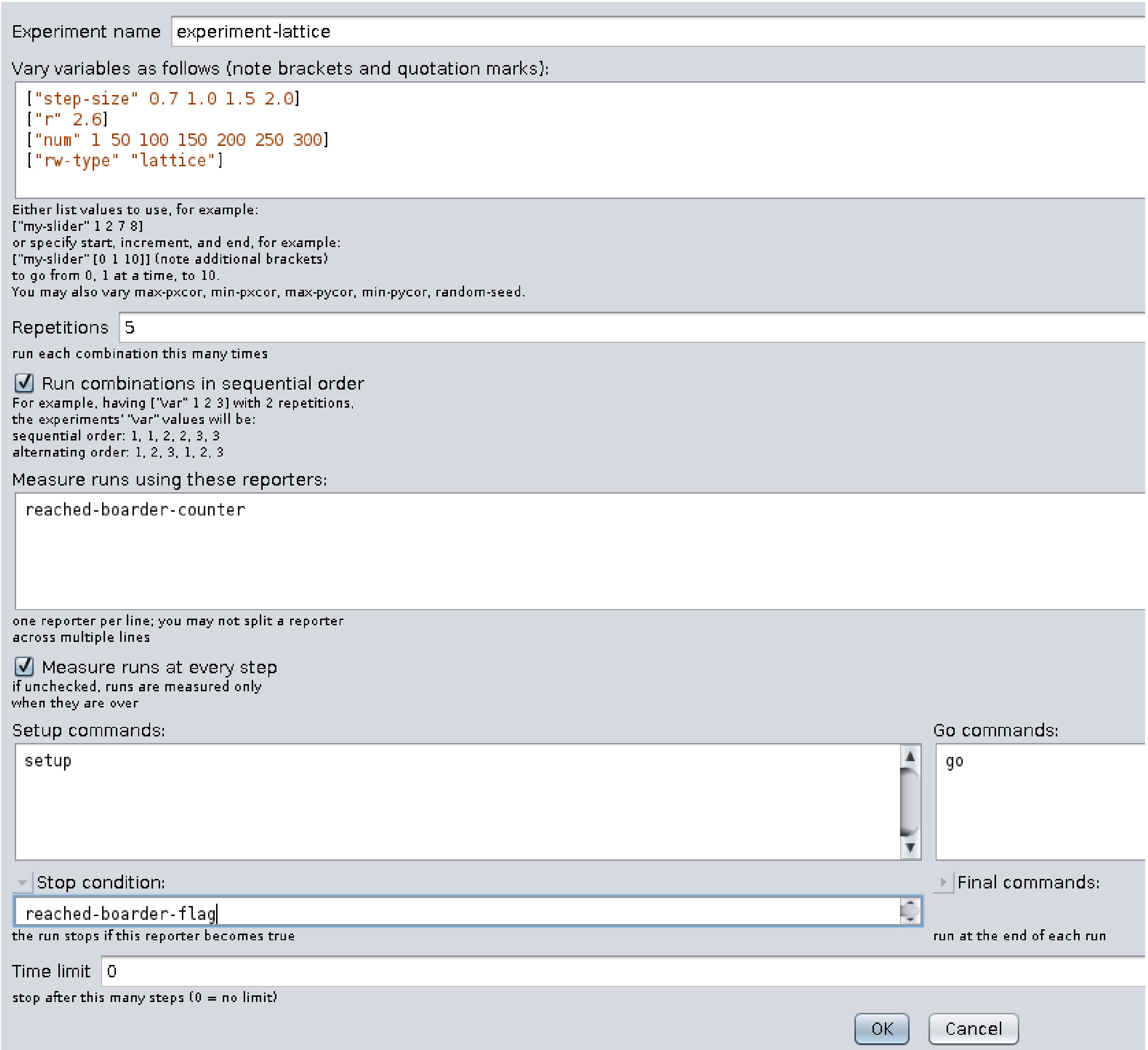


These are cases 4 & 1 - (in first case lambda is equal to 3.5, in second one it is equal to 0.5)
While if we take respectively 3& 2 cases, we would get something like this:
Which are basically the same, the graph goes in circles, the same as we've just seen by looking at the MSD graph - the perfect sinusoid of the case #2 and sinusoid-like graph of the case #3. Still gives us these:

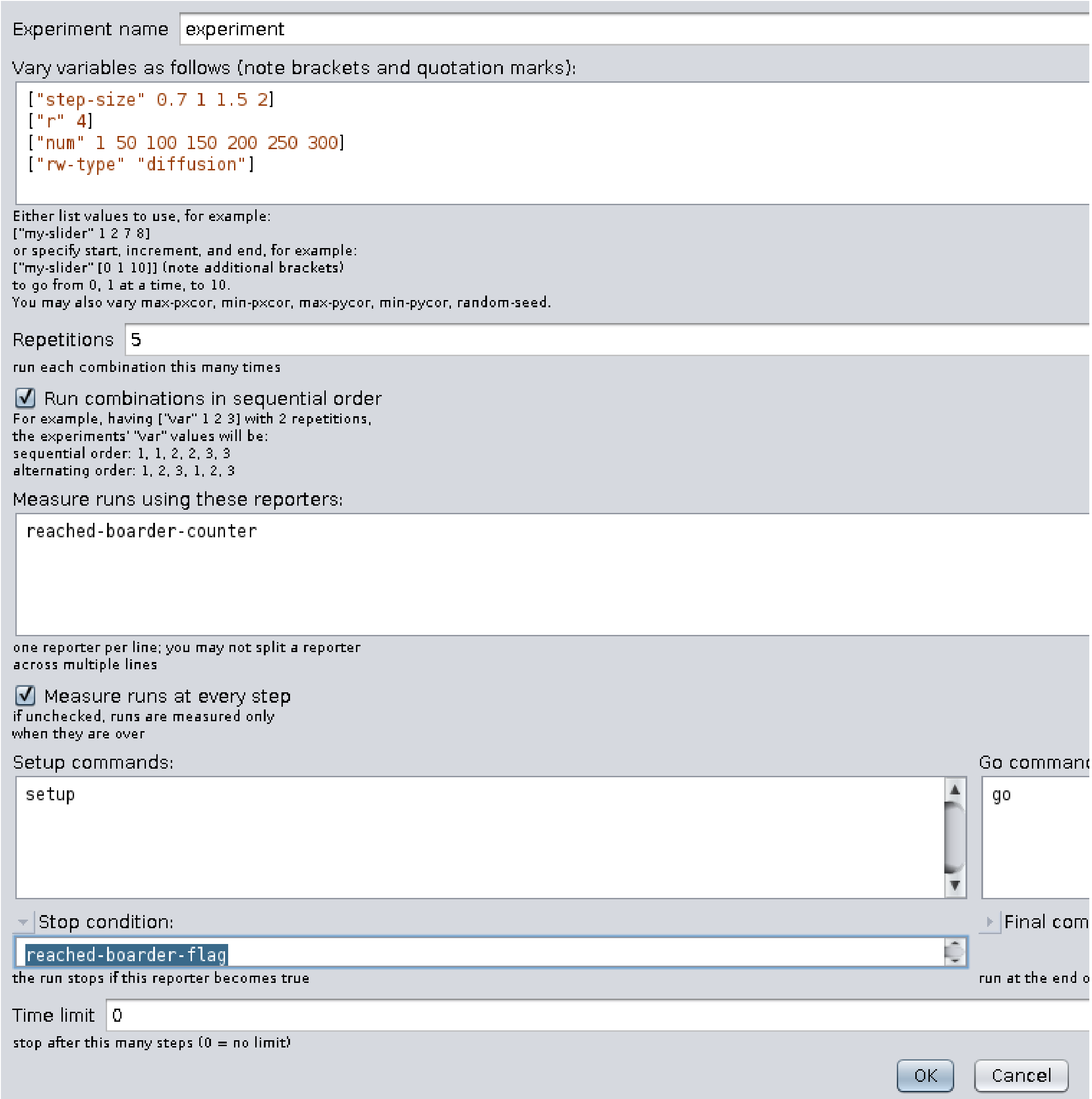
Final conclusion:

The investigated params for the

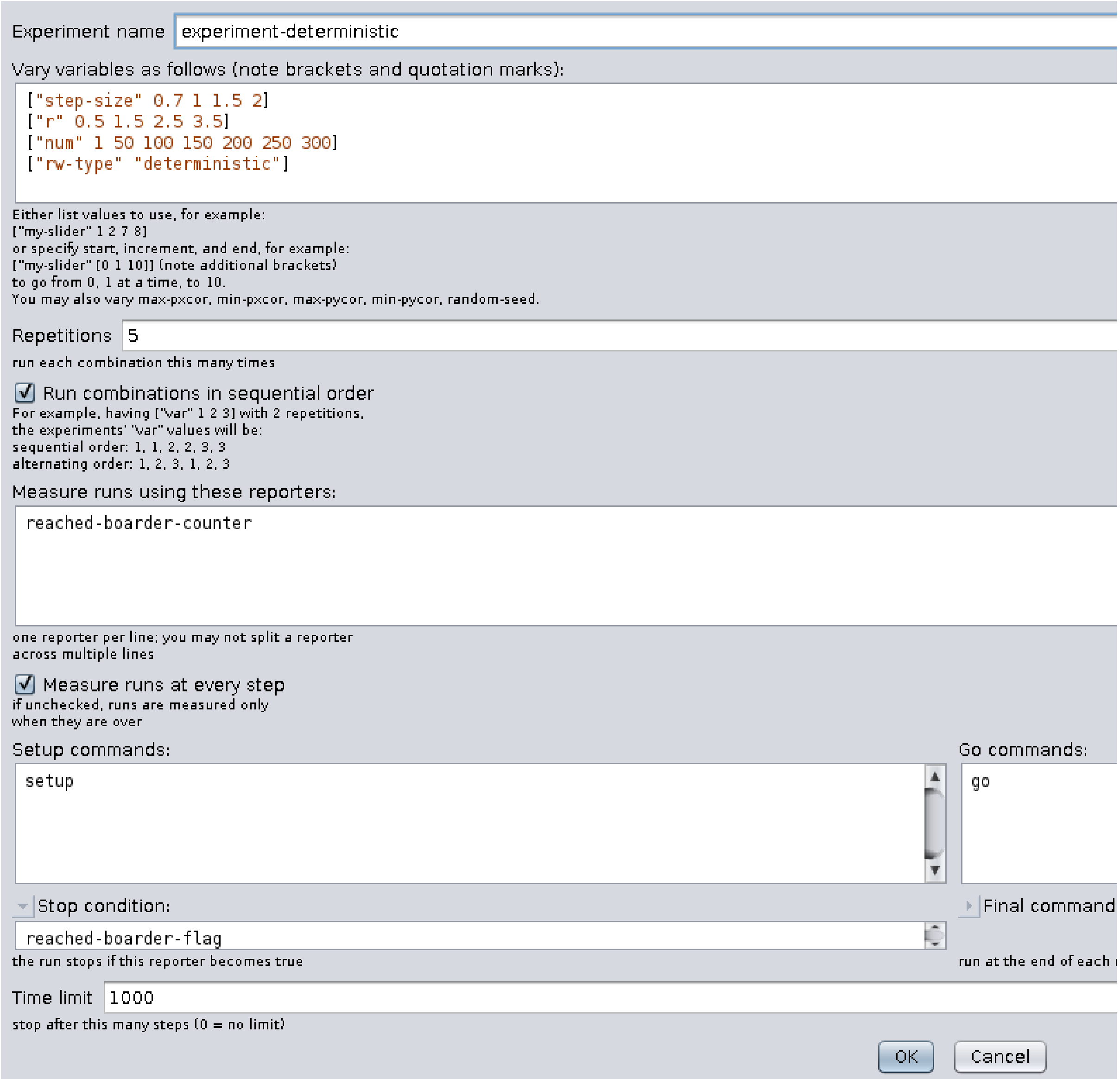
Lattice model -> num of agents, size.
The results I got are both described here, using all of the visualization from the lab itself or from the experiment. The experiment I set for this model is this:



For diffusion model-> max-length, size, number of agents.
Here the max-length of a step-size doesn't really play any role, because it can either the max distance between the spawn point and the corner - and then the time to reach the border is 1 tick - because 100% all of the turtles definitely get to the border independently of the angle. Other cases I've tested and described in my work, the setup experiment for this model looked like this:

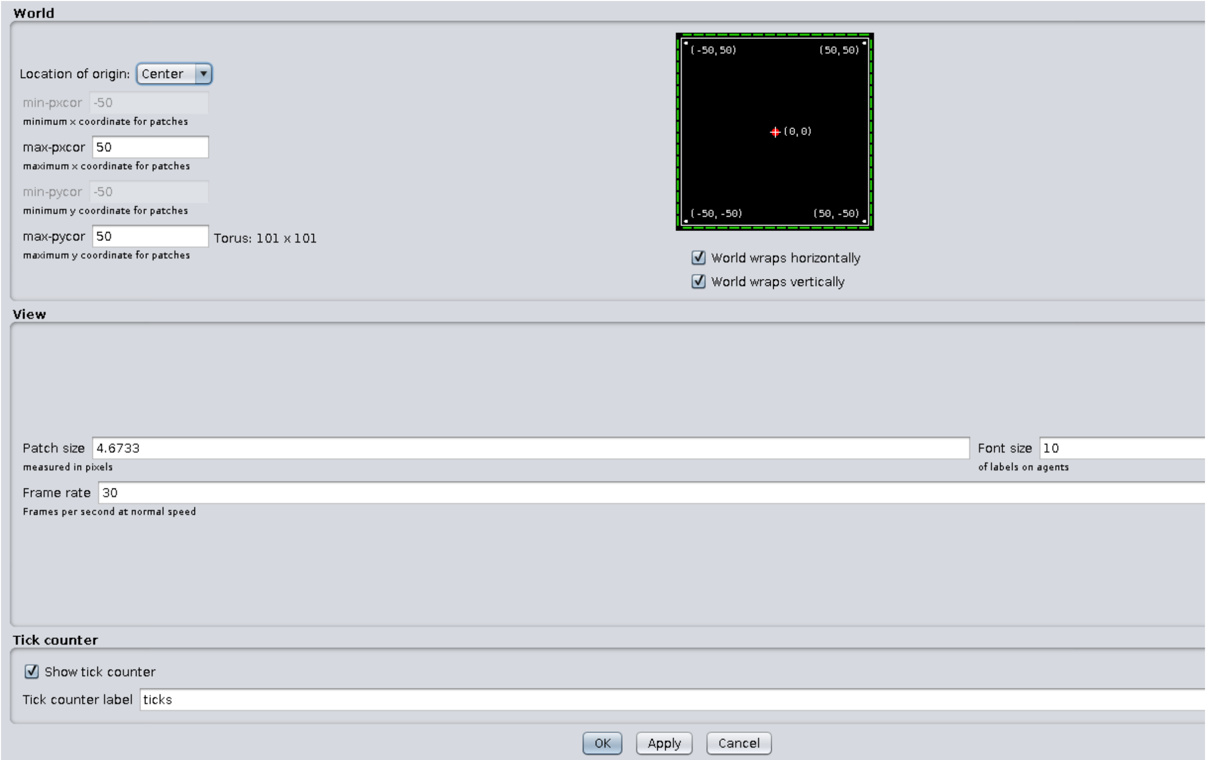


For the deterministic model -> alpha (i took 4 different), number of agents and step-size.



NOTES:

The params of the world for all three models while investigating them here are:


World
Location of origin: Center
min-pxcor -50
minimum x coordinate for patches
max-pxcor 50
maximum x coordinate for patches
min-pycor -50
minimum y coordinate for patches
max-pycor 50
maximum y coordinate for patches
Torus: 101 x 101
(-50, 50)
(50, 50)
(0, 0)
(-50, -50)
(50, -50)
World wraps horizontally
World wraps vertically
View
Patch size 4.6733
measured in pixels
Font size 10
of labels on agents
Frame rate 30
Frames per second at normal speed
Tick counter
Show tick counter
Tick counter label ticks
OK
Apply
Cancel