3/10/2018

Student Number:1731302

**The Schelling model**

Bradley Morgan

Please note that all code originates from my own programmatic build of the model and is not the same as the code provided by the tutor

|  |  |
| --- | --- |
| **Code Sample** | **Explanation: How do agents decide that they are in the minority?** |
| function [gridworld, relocation\_list, mean\_segregation,percent\_satisfied ] = satisfaction(gridworld, tolerance)    [rows, cols] = size(gridworld);  a = 0;  for r = 2:rows-1  for c = 2:cols-1  row\_pos = r;  col\_pos = c;  scope = gridworld(row\_pos, col\_pos);  if isequal(scope,1) || isequal(scope, -1)  a = a+1;  mask = gridworld(row\_pos-1:(row\_pos-1)+2, col\_pos-1:(col\_pos-1)+2);  mask(2, 2) = NaN;  mask = reshape(mask, [1, 9]);  mask(isnan(mask)) = [];  mask( :, ~any(mask,1) ) = [];  num\_neighbours = length(mask);  same = 0;  for i = 1:num\_neighbours  if isequal(mask(i), scope)  same = same + 1;  end  end    if isequal(num\_neighbours, 0)  percent\_same = tolerance;  elseif num\_neighbours > 0  percent\_same = same/num\_neighbours;  end    relocation\_list(a,1) = percent\_same;    if percent\_same >= tolerance  relocation\_list(a,2) = 100;  relocation\_list(a,3:4) = [row\_pos, col\_pos];  elseif percent\_same < tolerance  relocation\_list(a,2) = -100;  relocation\_list(a,3:4) = [row\_pos, col\_pos]; | * **For loops (r &c):** Iterate through the simulation space (grid world) that holds the locations and group ID of each agent. Scope is a spotlight that considers each element of the grid, if scope is equal to Group 1 (+1) or group 2(-1) then the current matrix element represents an agent. * **Mask Variable:** Extracts a 3x3 window of a targeted agents neighbors based upon the Moore neighborhood criterion which is composed of a central cell and the eight cells which surround it. Mask removes the central agent (the agent in question) from the assessment, as they themselves cannot contribute as their own neighbor. NaN values pad the grid space to allow the mask to spill over the edges of the matrix without causing MATLAB matrix dimension errors, these are also removed from the mask. Finally, 0 values are removed from the mask as empty space cannot contribute as a neighbor in an agent’s surroundings. * **For loop(i):** counts how many elements in the mask have equal group ID values to the agents. The results of this are then divided by the total number of surrounding neighbors to produce a percentage of agent satisfaction, which is directly related to the level of individual segregation in that specific region. * **Percent\_same & Tolerance Variables:** Agent satisfaction / percentage of similar neighbors is then compared to the tolerance threshold. Tolerance indicates the minimum percentage of similar neighbors required for the agent to remain in its location and not relocate. If the agents level of satisfaction is smaller than or equal to this threshold then agent will remain, whereas if it is greater the agent will indicate that it wants to move due to the actual percentage of similar neighbors being smaller than its required percentage of similar neighbors. If an agent is surrounded by empty space then the agent will remain as the rule states agents only move when they are in a minority. * **Relocation List:** Stores the agents group Id, location and whether they want to relocate(-100, unsatisfied) or remain(+100). |
| **Code Sample** | **Explanation:** How do agents move? |
| function new\_world = MoveAgents(relocation\_list, world)  [num\_agents, ~] = size(relocation\_list);    for i = 1:num\_agents  [available\_loc\_rows, available\_loc\_cols] = find(world == 0);  available\_loc = [available\_loc\_rows, available\_loc\_cols];  len = length(available\_loc);  agent = relocation\_list(i, 2);    if isequal(agent, -100)  %agent wants to move  %find agents group and current location  agents\_group = world(relocation\_list(i, 3), relocation\_list(i, 4));  new\_pos = available\_loc(randi([1, len], 1), :);    %update world state with agents new position  world(relocation\_list(i, 3), relocation\_list(i, 4)) = 0;  world(new\_pos(1), new\_pos(2)) = agents\_group;    end    end  new\_world = world;  end | * The Function receives the current agent world state and the list of agents locations and whether they want to stay or relocate(relocation list). * For loop(i)  1. Finds all 0 or empty grid space locations in the agent world for each agent. If the agent wants to relocate then they randomly select their new location from the available empty spaces. 2. Agent movement is represented by the current matrix element switching from +1/-1 to 0 and the new location element in the matrix switching from 0 to the agents respective group number (+1, -1). This allows the possibility for agents to move into locations previously occupied by another agents, rather than just the initially available empty space. 3. All agent moves or remains are carried together within 1 timestep. The world state is updated and then passed back to the main file for plotting. |

|  |  |
| --- | --- |
| **Code Sample** | * **Explanation:** When does the simulation stop? |
| timesteps =300;  satisfied\_lim = [100, 100, 100, 100, 100, 100, 100, 100, 25, 15, 5];  if isequal(percent\_satisfied, 100) || isequal(time, timesteps)  if isequal(endtrial, 1), break, end | This following statements are used within the (statement 1) Social\_seg\_model, (statement 2) PlotDrawSave functions. Both statements allow for the simulation to run until all agents are satisfied with their surrounding ratio of same or different neighbors, however the variable timesteps indicates the maximum number of simulation iterations that can be run before forced simulation termination, regardless of agent satisfaction. Likewise, satisfied variable indicates the global percentage of agents satisfied for each simulation trial. |

|  |  |
| --- | --- |
| **Code Sample** | **Explanation:** How to measure segregation? |
| mean\_segregation = (mean(relocation\_list(:, 1)))\*100; | Given that all agents follow the prime directive to avoid being in a minority, then their level of satisfaction demands that they move away or ‘segregate’ themselves, by varying degrees of tolerance, from non-similar agents that lead to such instances of being in a minority. Thus agent 'satisfaction' is directly proportional to the 'number of similar neighbors compared to total neighbors’, and so agent satisfaction represents the percentage of segregation in the local region surrounding a specific agent. Therefore, calculating the mean satisfaction across all agents reveals a metric of the current global level of segregation across all regions of the simulation space that occupies an agent. |

**%PARAMETERS DESCRIPTIONS**

% **Grid\_size:** Size of simulation space

% **group 1 = +1 group 2 = -1**

% **tolerance:** A percentage threshold of required similarity between agent and neighbours

% **percentage(:, 1):** percentage of agents that populate the world space vs empty space

% **percentage(:, 2):** percentage of group1 agents with respect to the total population of agents in the world space

% **timesteps:** Maximum number of iterations per simulation run

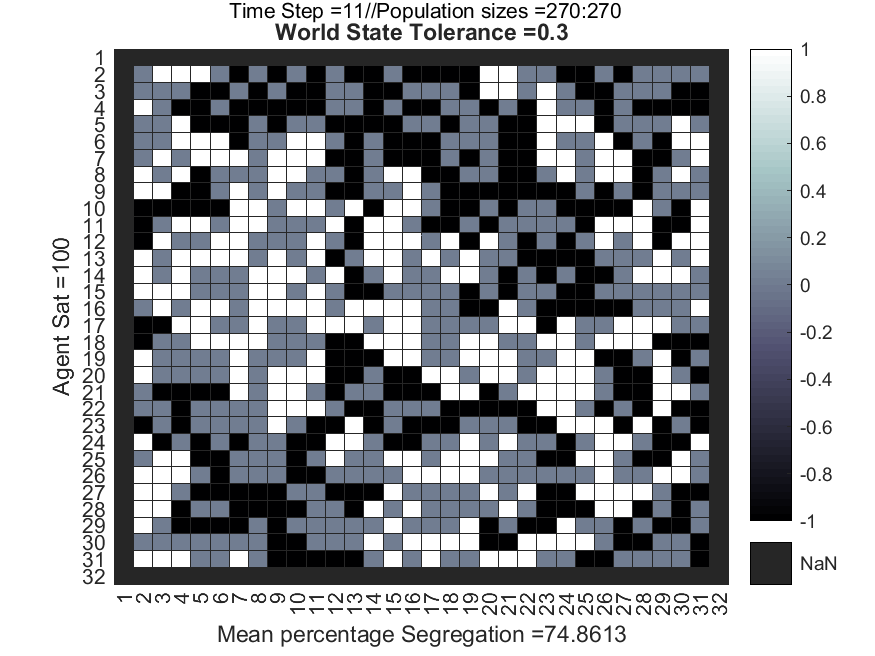
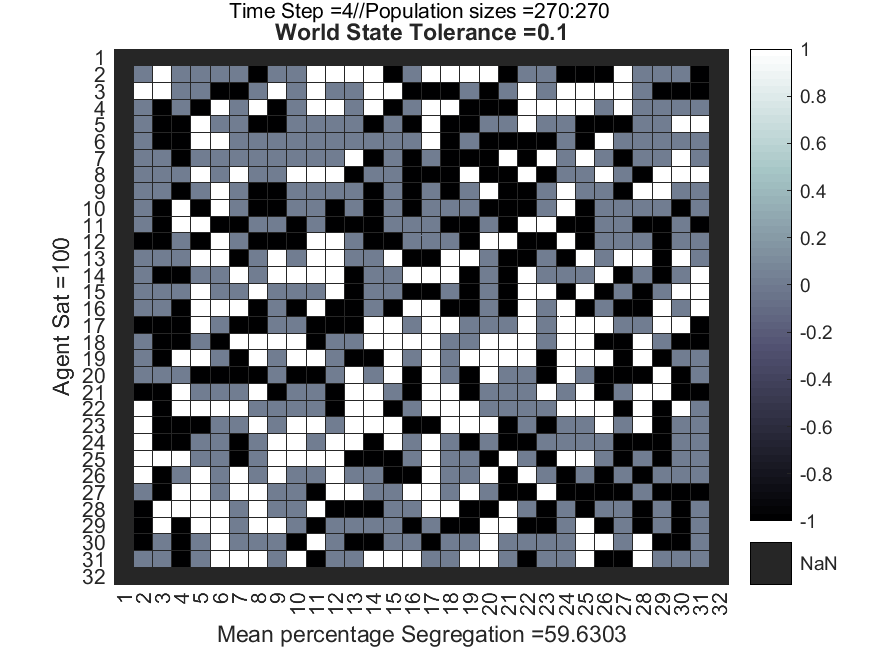
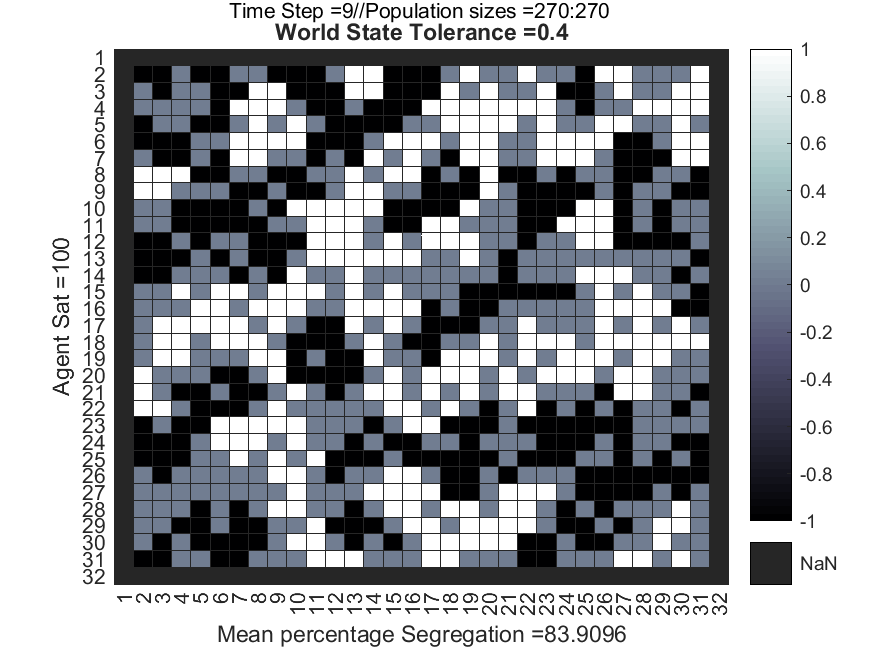
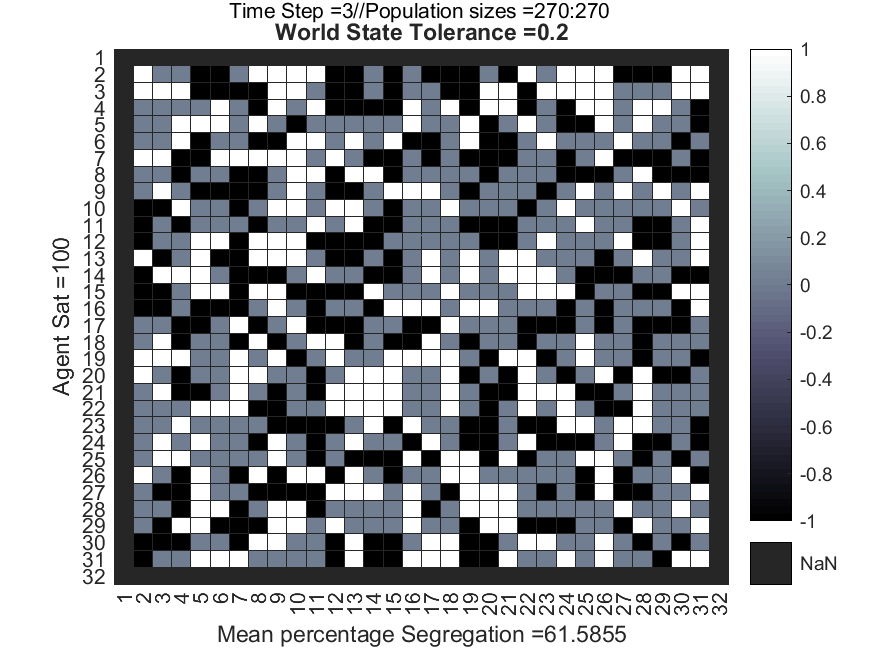
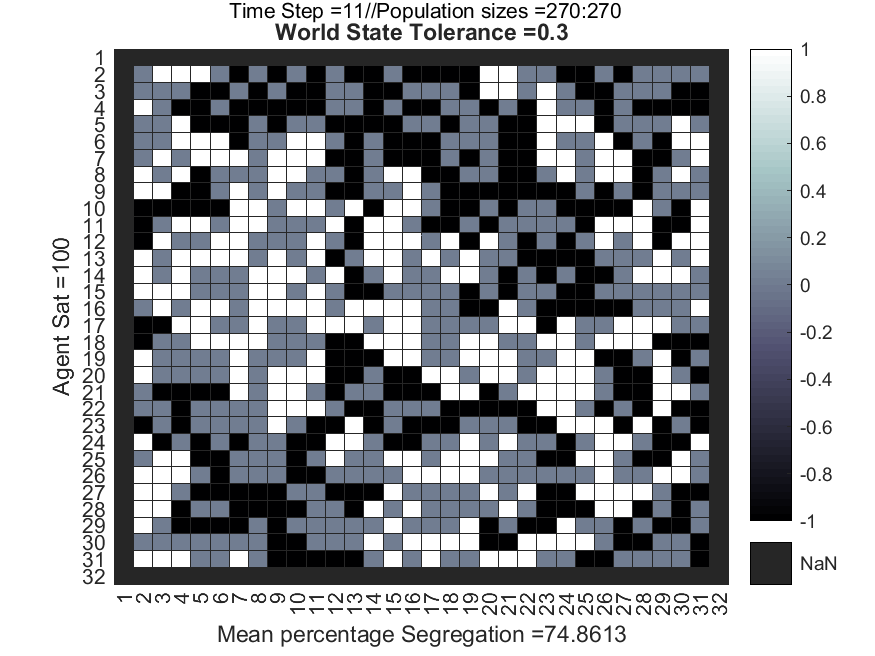
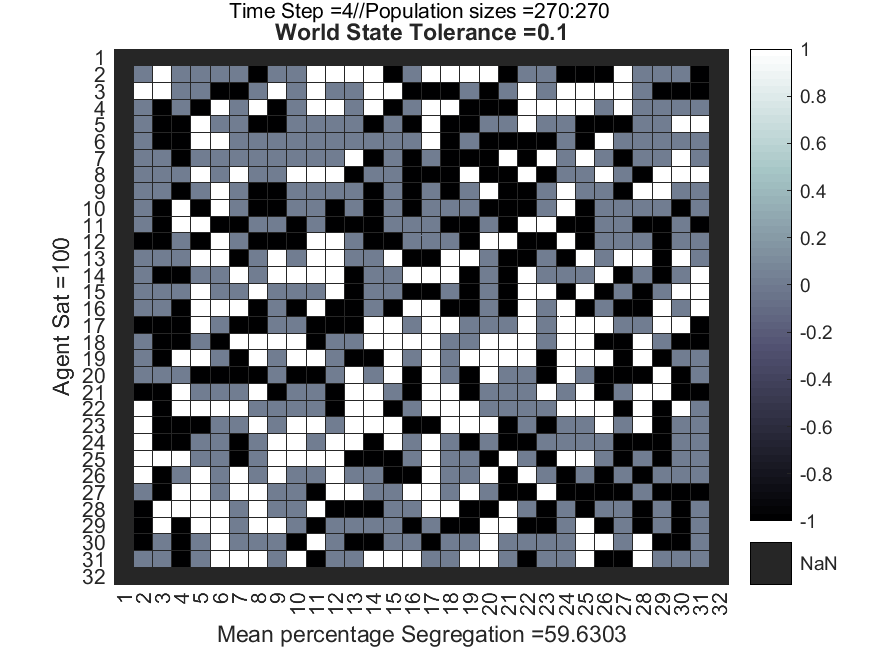
% **satisfied\_lim**: Stop the simulation when agent satisfaction reaches

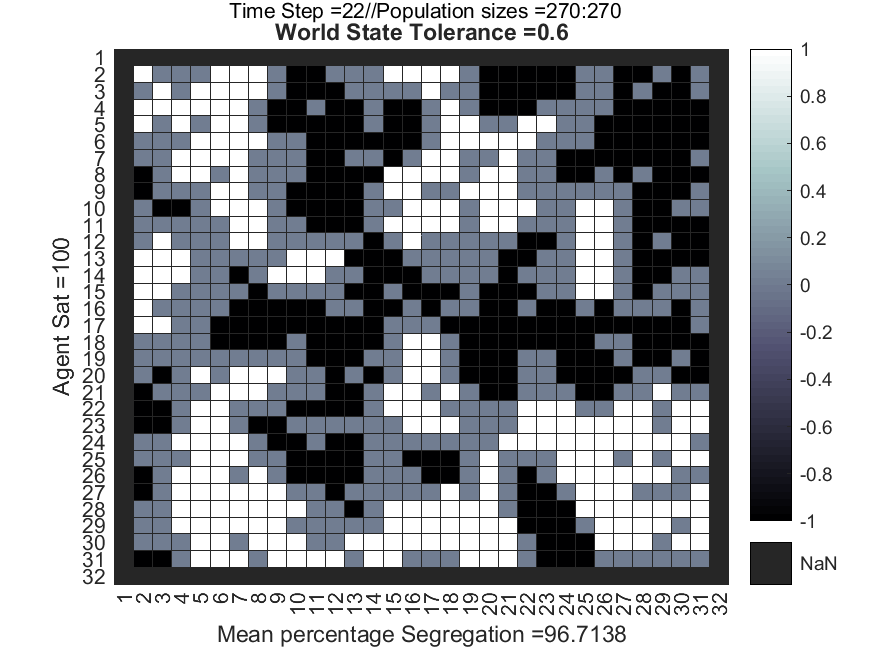
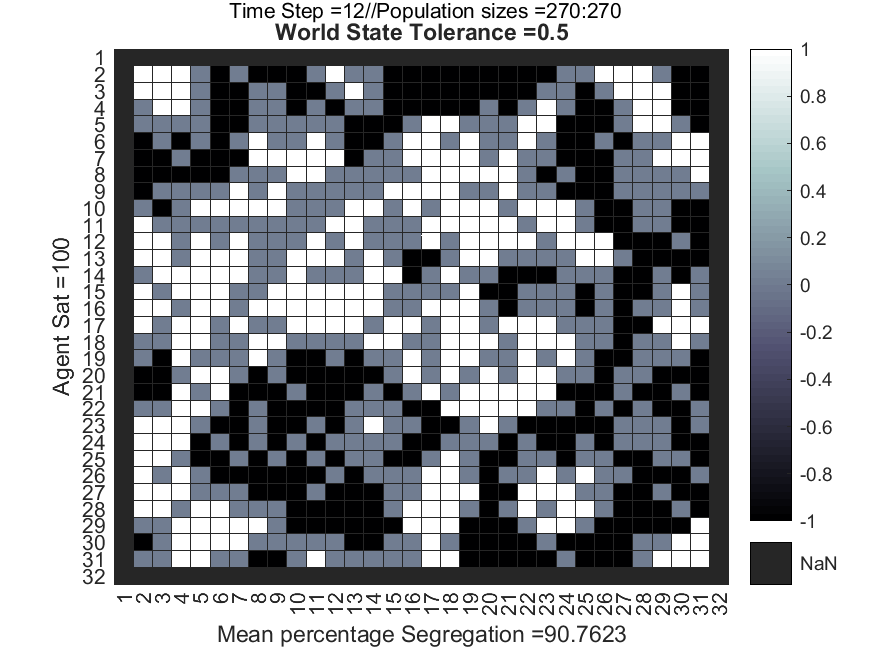
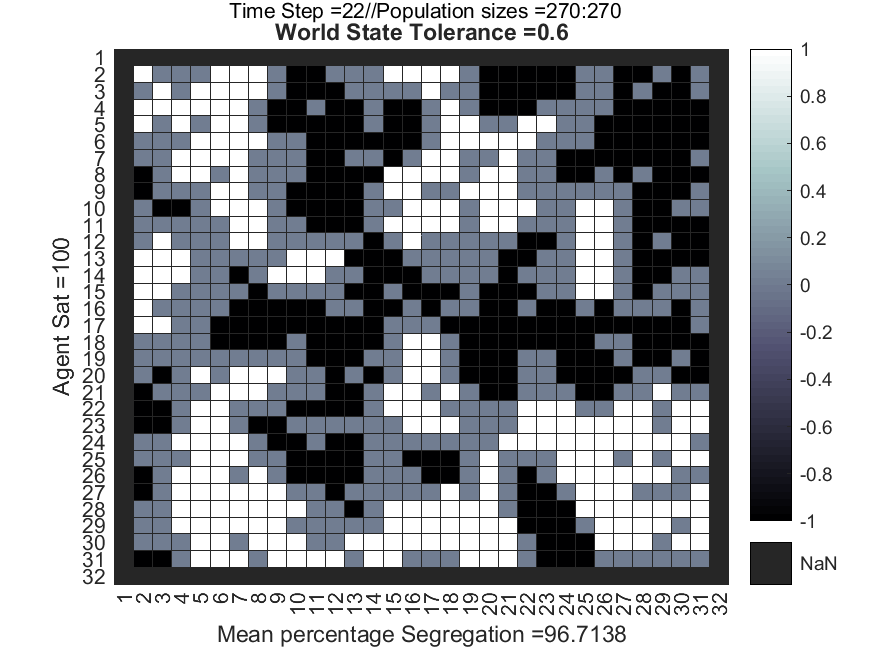
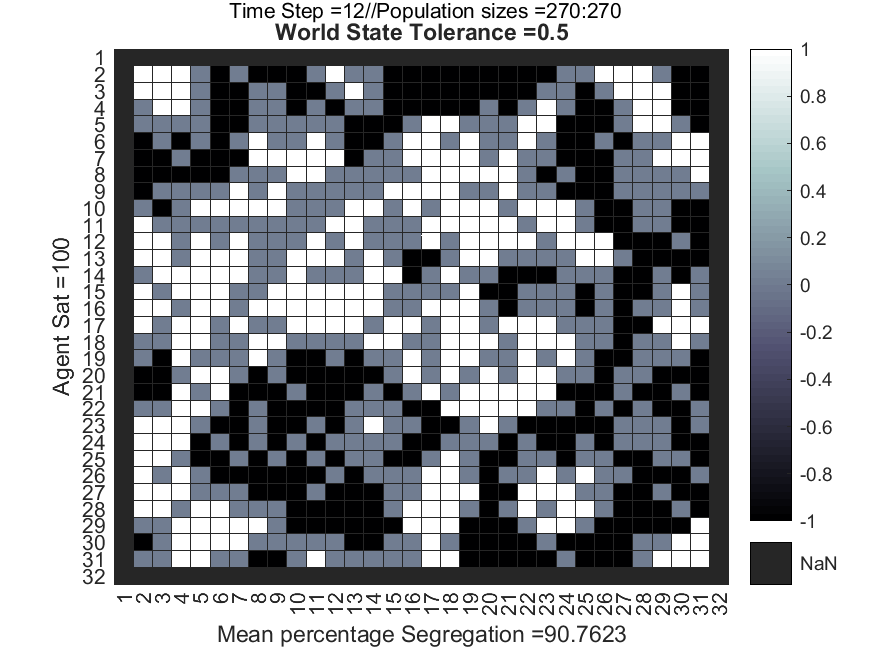
% spesfied limit.

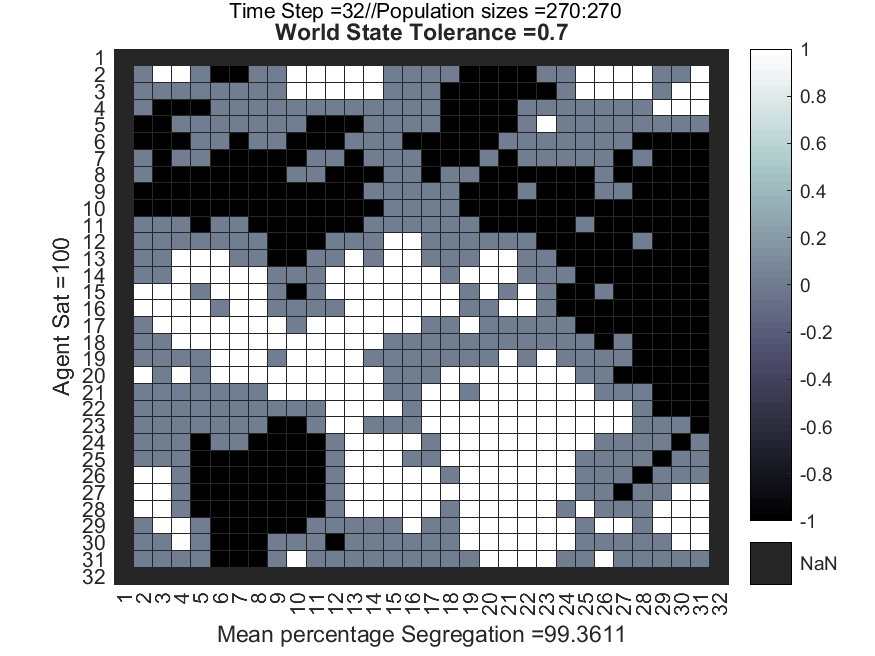
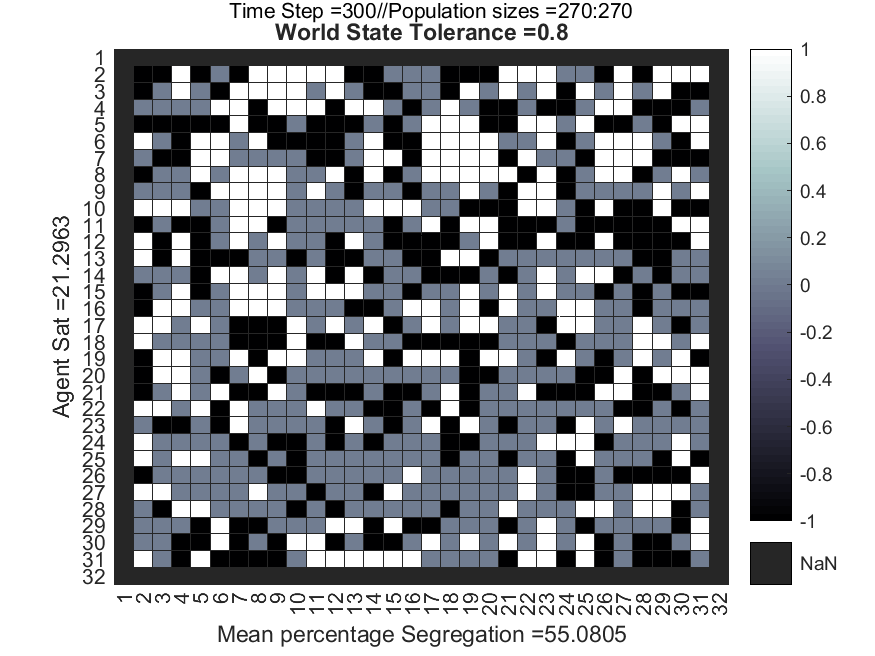
% **NobodyAround\_Rule:** If agent has no neighbours then by default their level

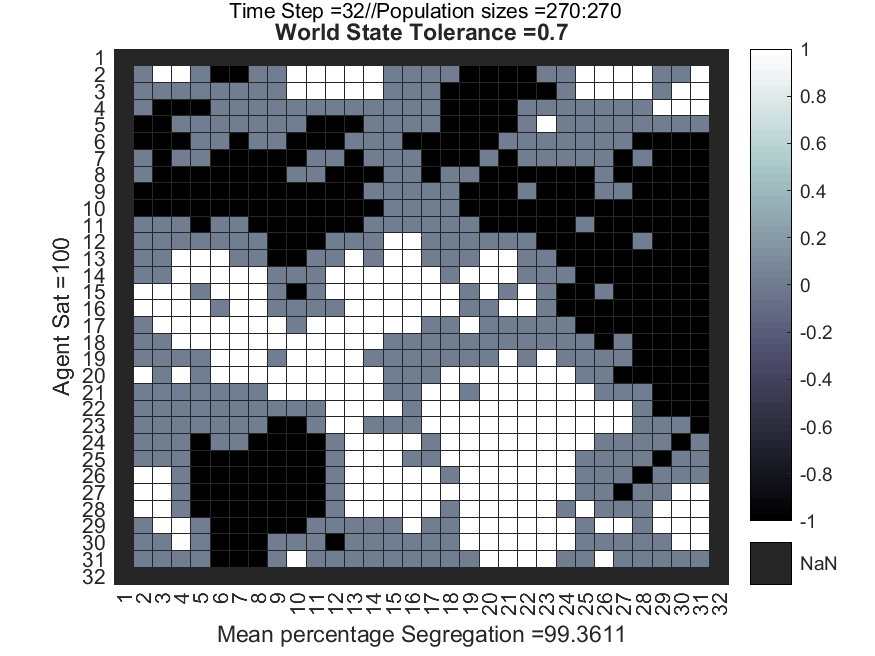
% of segregation must be equal to 0;

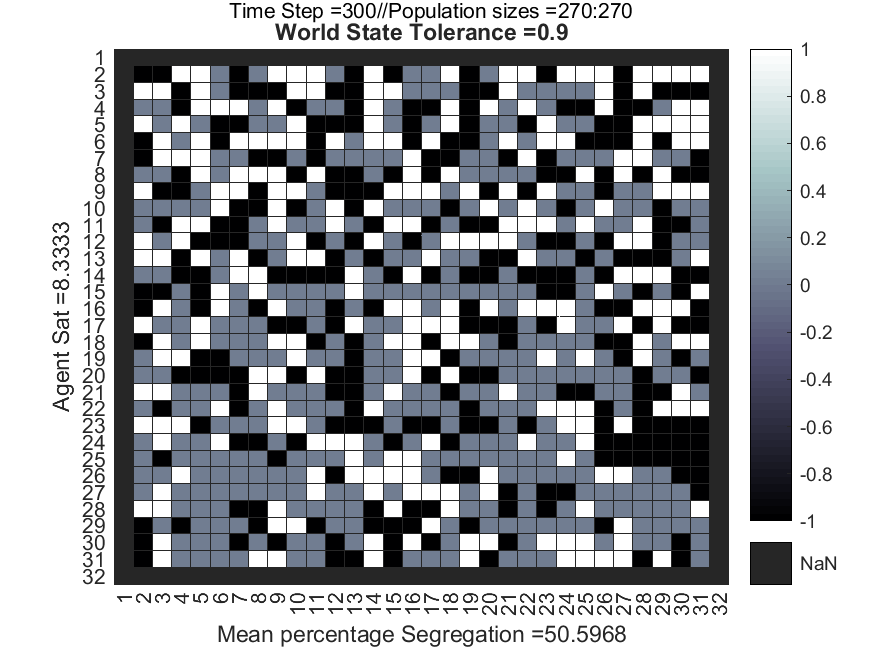
% **Avoid\_total\_emptyspace:** Agents will or will not relocate if they are surrounded by empty space. Agents avoid minorities and seek being in a majority or agents only just seek to avoid being in a minority.

**Parameter Experimentation and respective Results**







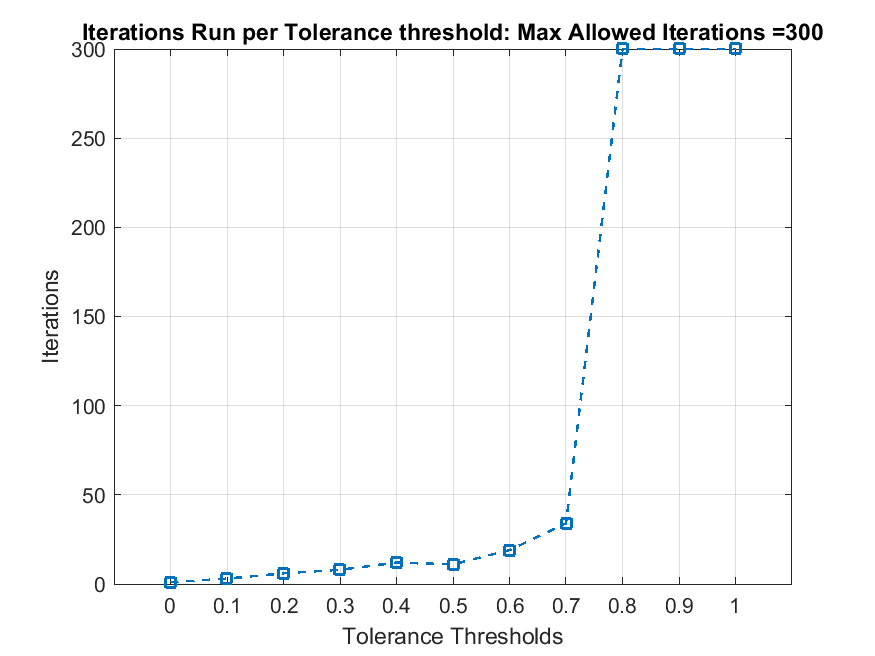
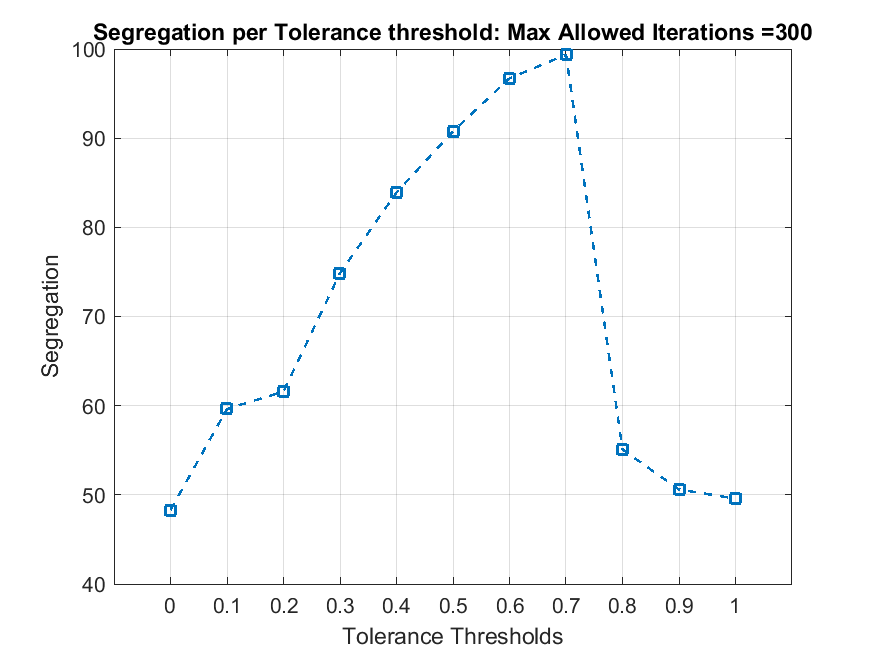


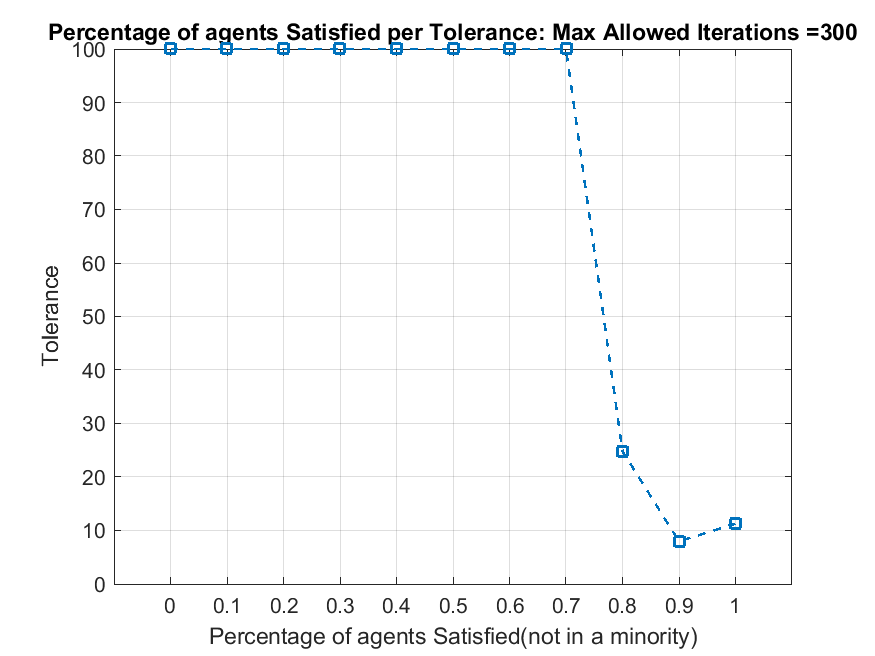
**%MODEL PARAMTERS:**

* grid\_size = [30 30 30 30 30 30 30 30 30 30 30];
* groups = [1, -1];
* tolerance = [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1];
* percentage = [60,50]; 60% of the grid will be populated with agents and 50% of that agent pool will be group 1.
* timesteps =300;

**Results:**

* Overall results show that agents can be individually very tolerant of agents from a different group, yet despite this the macroscopic effects lead to a high percentage of global segregation between groups, as shown by simulation tolerance = 30%. The effects are amplified when agent tolerance is slightly biased to their own group (see simulation tolerance = 60%), causing almost 100% regional segregation between groups.
* When Agent tolerance is very high at above 0.75, then one observes a rapid decrease in regional segregation and agent satisfaction. Increases in segregation requires increased clustering between the different groups of agents. Group specific cluster structures must be maintained and increase in size over time, in order to maintain or increase segregation. If agents have an extremely high tolerance threshold of say 0.8, then this will only require the presence of 2 non-similar agents causing agents within the cluster to begin relocating. The probability that two non-similar agents will simultaneously appear at the borders of a group cluster is higher than the probability of four non-similar agent’s(akin to tolerance = 0.6) simultaneously appearing. Therefore, clusters containing agents with higher tolerances will be less stable and more likely to be disbanded, randomly redistributing the agents across the grid space, which will decrease overall segregation.

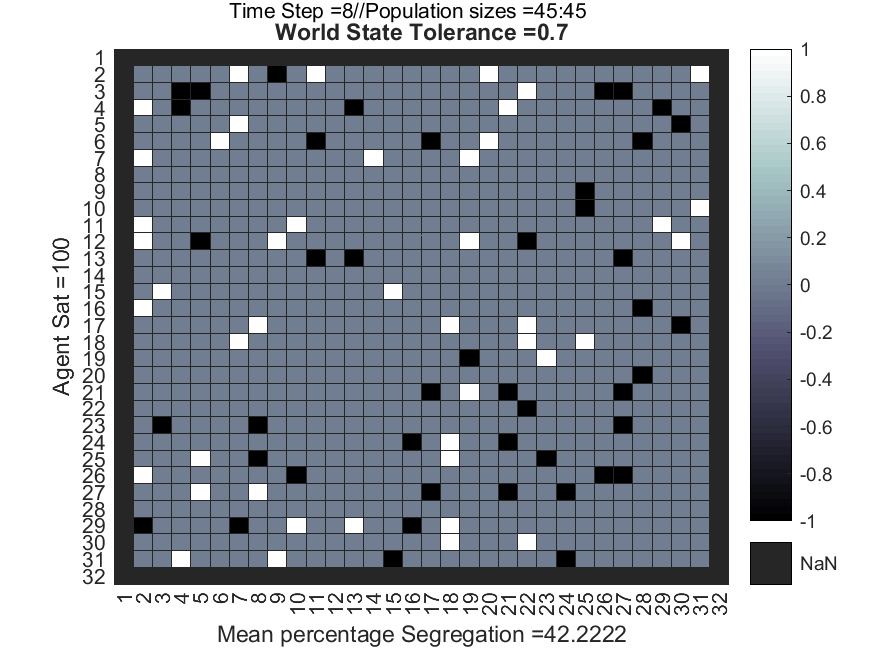
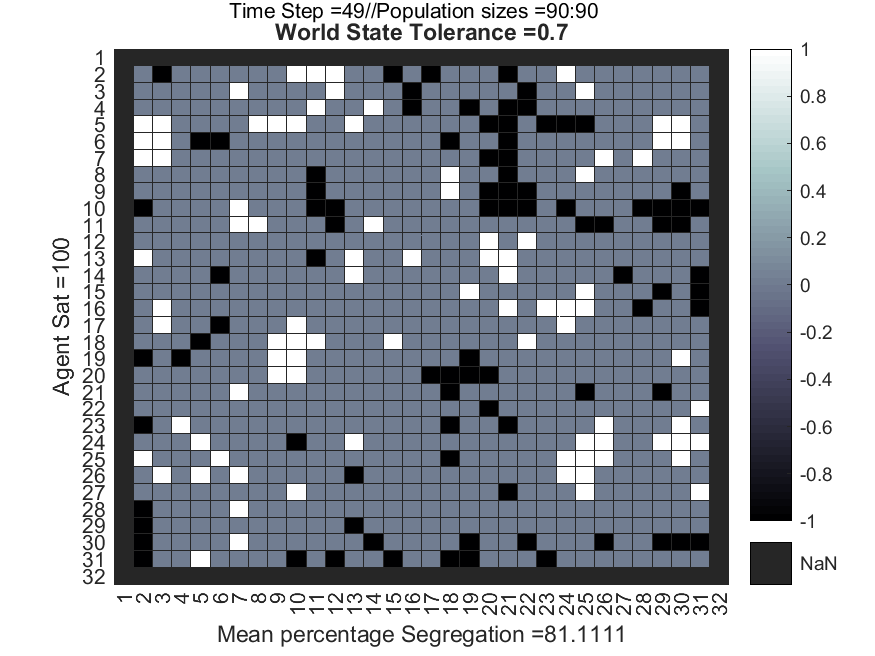
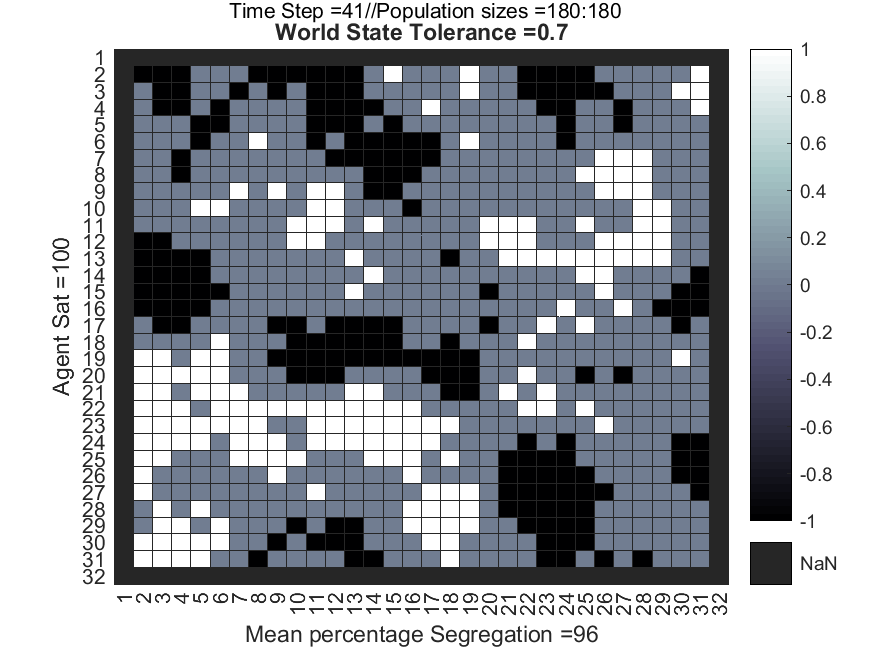


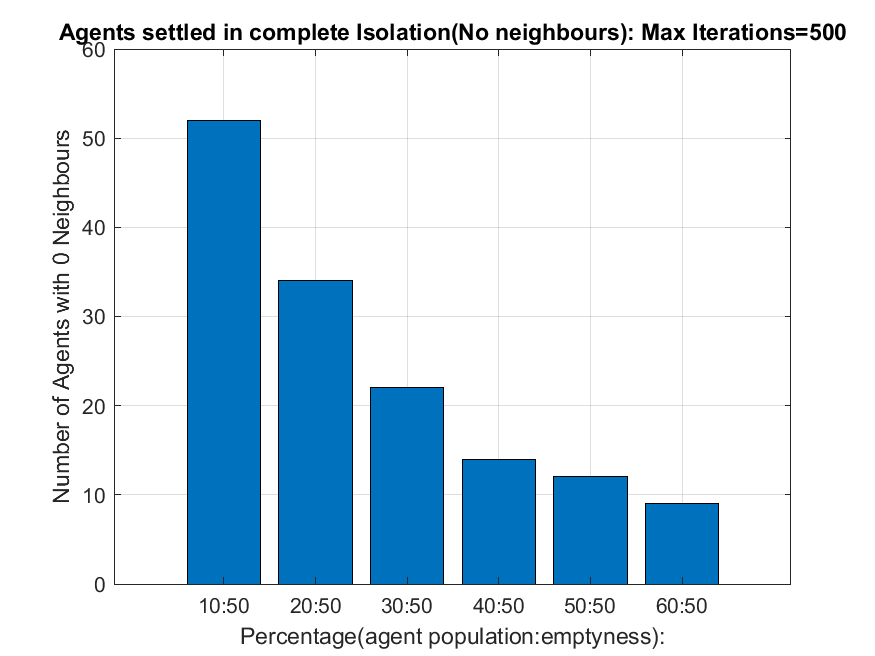
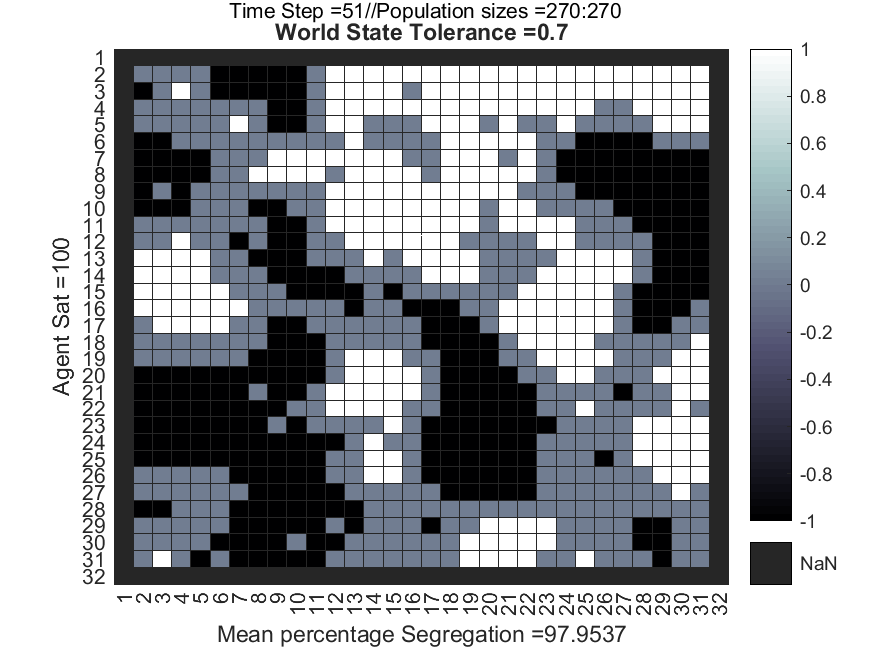


The top right graph indicates how the frequent destabilisation of group clustering between agents causes the required number of iterations for all agents to reach 100% satisfaction to exponentially increase towards the maximum iteration limit. The lower left graph also shows how simulations with tolerance = 0.8 or higher have extremely low levels of agents satisfied within 300 iterations. For example, the simulation with a tolerance of 0.8 reaches around 25 iterations over 300 trials, one may crudely predict that it will take 300\*(100-25/25) = 900 iterations to reach 100% agent satisfaction.

Percentage of agents Satisfied (not in minority)

Tolerance

Interesting observation



**Observation: Influence of Empty space and Agent Tolerance**

The ratio of agent population vs empty space and how agents respond to empty space effects overall segregation. Technically speaking if an agent is surrounded by empty space then they are not in a minority nor in a majority. Agents will not relocate from their current position but likewise will not contribute to the overall level of segregation as they will have 0 neighbours against 0 neighbours that are the same as them. This can only be meaningfully represented as a value of segregation in a specific local region, with respect to an agent, being equal to 0.

Results show that if agents can remain in a location surrounded by empty space then more agents will tend towards relocating to a position in which they are neither in a minority nor a majority region. The more empty space available the more agents will display the above behaviour, thus leading to reduced segregation despite agents having very low tolerances for non-similar neighbours. However, as the amount of available empty space decreases the more agent’s tolerance levels will influence the positioning of agents. The result of this being increased clustering behaviour between groups, leading to much higher levels of segregation and fewer numbers of agents choosing to reside in complete group isolation (no neighbours). The heatmaps and bar charts on the left and above evidences this. For clarification the x axis on the bar chart shows that for example 10% of the grid space is occupied by agents and 50% of those agents are from group 1, hence 10:50. Thus as the percentage of occupied space by agents increases the smaller the amount of available empty space.

**%MODEL PARAMTERS**

grid\_size = [30, 30, 30, 30, 30, 30];

groups = [1, -1];

tolerance = [0.7, 0.7, 0.7, 0.7, 0.7, 0.7];

percentage = [10,50;20,50;30,50;40,50;50,50;60,50];

timesteps =500;

satisfied\_lim = [100, 100, 100, 100, 100, 100];

NobodyAround\_Rule = [0, 0, 0 0 0 0];

Avoid\_total\_emptyspace = {'no', 'no', 'no', 'no', 'no', 'no'};

Interesting Observation:

Despite the many insights the Schelling model may offer one may argue that the model in its current form is too simplistic, and only represent agents making decisions on sole basis of minority avoidance. However, it is clear that in reality active agents/individuals will consider many other factors besides minority avoidance, when selecting whether to remain or relocate. I believe a vital factor that plays a role in an agent’s life is survival. Survival may refer anything from physical or economic/financial wellbeing. Given so it seems reasonable to suggest that an agent will compromise its own perception of minority avoidance in exchange to preserve itself. Thus I constructed an adaptation of the Schelling model to incorporate survival needs, the specifications are as follows:

* Agents are initialised with personal survival needs meters, ranging from 1-100, 100 being an optimum state of wellbeing. Survival needs decrease per timestep
* Agents are initialised with personal resource pools, resources are spent on the upkeep of an agent’s survival requirements, per timestep.
* Agents can do work and gain resources if they are located on a cell with an active pool of resources to mine from.
* Agents measure whether they are in a local minority and can still choose to relocate but only if they can afford it. Relocation cost lots of resources(variable).
* Agents prime directive is survival and secondary is their avoidance of minorities, there available decisions are:

1. Agent is satisfied with the neighbourhood but has low resources. Agent can choose to relocate or do work and gain resources if they are on a resource pool
2. Agent is satisfied with the neighbourhood and has enough resources. Agent will remain in current location.
3. Agent has enough resources but is not satisfied with the neighbourhood. The Agent can choose to relocate if they can afford it, remain if they cant afford it, remain and exploit potential profits, do work if located on a resource pool that contains twice the amount of the agents current pool of resources, Despite not wanting to be in a minority.
4. Agent has little resources and is not satisfied with the neighbourhood. Agent can Relocate if they can afford, remain if they cannot, or do work if they are located upon a resource pool.

* The Cost to relocate, the distribution of resources and max and mins of available resources, can be varied per simulation. Agents can have tolerance thresholds independent of others or be uniform across all.
* Agents have their own resource threshold that allows them to assess they criterion for warning them on having low amounts of available resources.

%MODEL PARAMTERS

grid\_size = [27];

groups = [1, -1];

%tolerance = [0.8];

**tol\_max = 70;** % Max Tolerance percentage

**tol\_min = 70;** % Min Tolerance percentage

**percentage = [65,50];** % Percentage of agents that will populate the space and their respective group population %’s

**resources = [20,10,5];** % Varying tiers of available resources. 1st element is the bottom 3rd of possible values from the maximum resource pool value e.g 1000 = split into 1-333, 333-666, 666-999 ranges.

**agent\_resmax = 320;** % Maximum value of resources an agent can begin with.

**survival\_max = 100;** % Maximum value of survival requirements an agent can begin with. 0 = is bad!!!!! 100 = Awesome!

**res\_pool\_max = 800;** % Maximum value that a resource pool can begin with.

**res\_threshold = 100;** % Financial sensibility, at what point does one declare I’m running out of resources.

**maxpay = 30; %** Max possible pay an agent may receive when doing work at a resource pool

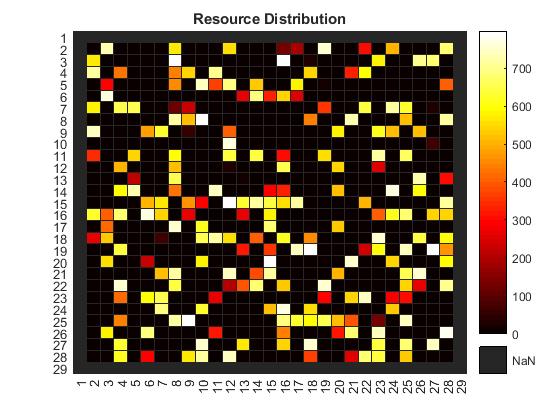
**relocation\_cost = 55;** % the cost to change where an agent lives

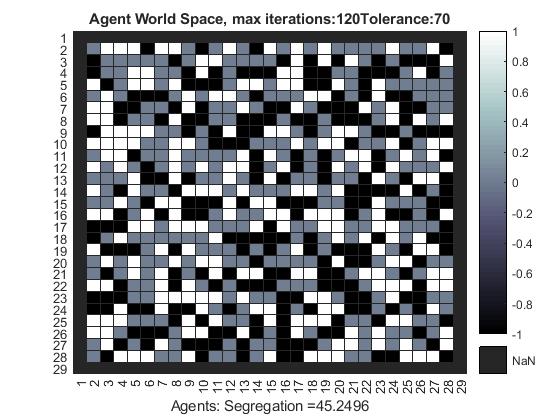
**timesteps =120;** % maximum number timesteps the simulation will run for.

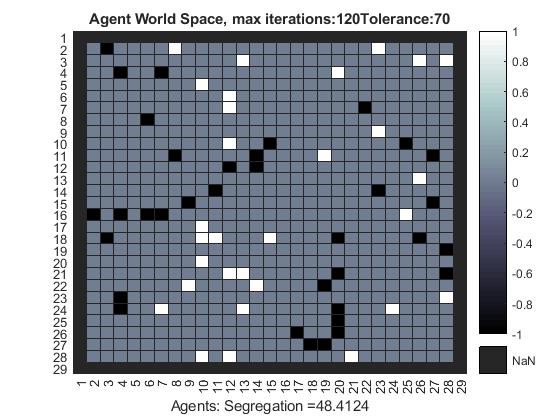
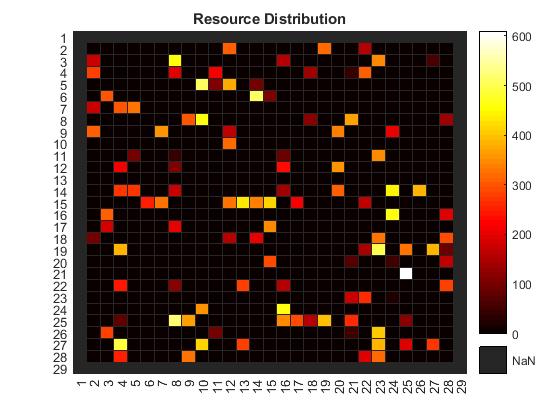
**NobodyAround\_Rule = 0;** % how agents show respond to empty space.

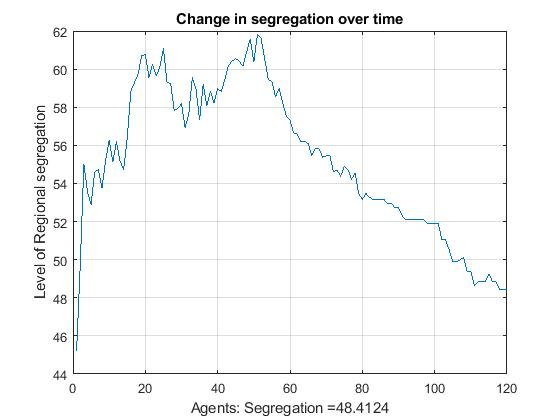
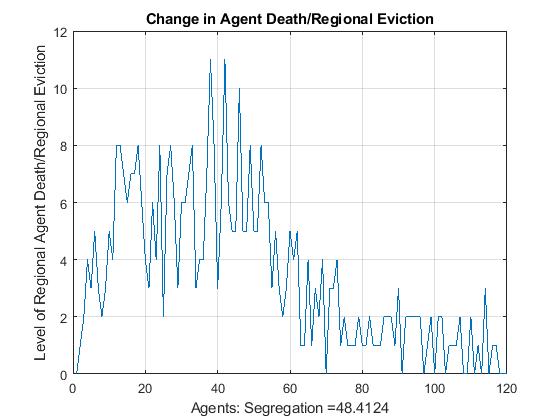
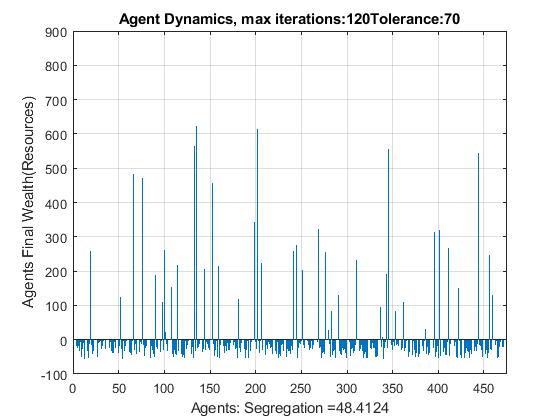
**Avoid\_total\_emptyspace = 'no'**; % should agents seek being in a majority as well as avoid being in minority

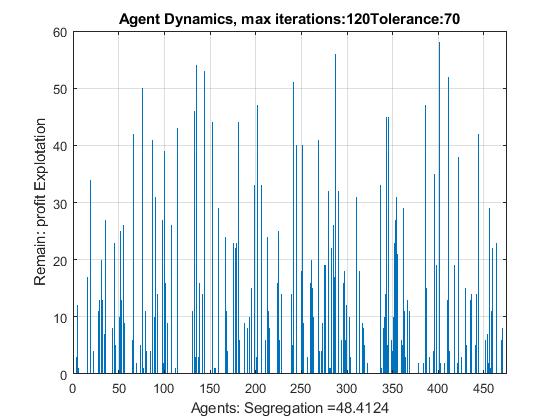
**frame\_rate = 0.05; %** refresh rate of the plots.

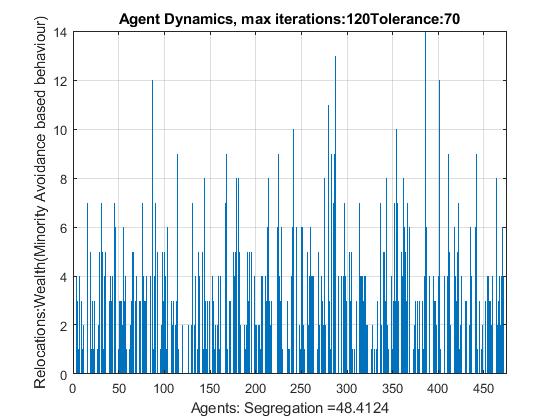
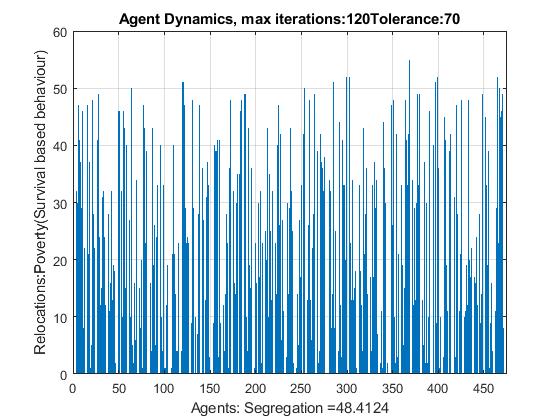
Results:

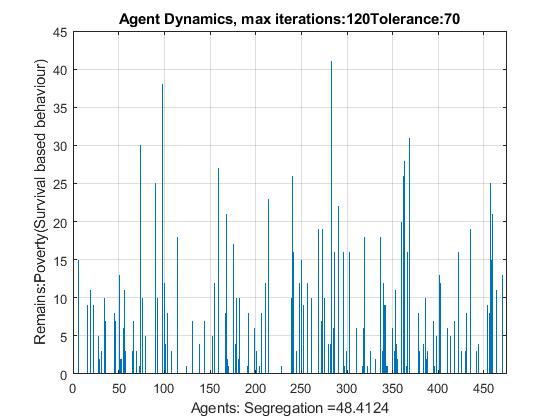












Overall results demonstrate agent behaviour during which there is a limited distribution of shallow pooled resources, high relocation rates, and a high initial population with a 70% intolerance for non-similar neighbours. As show by the plot of segregation over time, one can see that in the initial 60 trials segregation increases due to agents having access to deep resource pools that pay relatively high amounts to the agents for work. More people are wealthy in this period and thus can afford the luxury to relocate themselves by preference of minority avoidance based behaviours. One can see from the initial agent space heatmap that the groups are much more clustered and segregated. However high wages and access to resource is also occupied by high relocation prices, likewise agents are initialised with there own threshold of financial sensibility. Agents that have low resource threshold are more likely to spend quickly and realise they have run out resources a lot slower, causing them to over spend without cautiously saving. As the economic climate takes a turn for the worst, one observes that after trail 60 the total amount of available resources has decreased too rapidly, prices for moving and wages for working are too high and become unbalanced with the available resources. Interestingly one sees that the change in level of segregation over time rapidly decreases as the amount of available resources decrease. This is firstly due to the financially insensible agents being quickly evicted for bankruptcy from their homes and the region, this is evidenced by the plot of change in agent death/eviction overtime, one sees a rapid spike between trails 40-60 at precisely the same time frame as the initial economic down turn. Nonetheless this spike in evictions decreases much more rapidly after trail 60 than the overall level of segregation. Why does the level of segregation decrease more smoothly? This is due to the interplay between luxury over necessity. As time progresses more of the remaining middle and higher wealthy agents must sacrifice the luxury of deciding to relocate to avoid being in a minority, but rather must relocate on the premise of finding new resource pools, to sustain their survival. Its most likely that wealthy agents that have lasted this long will be very financially sensible and have high resource thresholds causing them to begin making more necessity-based decisions earlier, allowing them to save for longer. Nonetheless even the wealthy despite having a strong intolerance for non-similar neighbours must compromise and accept a much lower intolerance level to remain or move to regions still rich in resources, thus driving down the level of segregation. Eventually as the economic situation reaches it extremes so many people are evicted that only the wealthiest agents remain, as show by the final wealth plot. There is now a lot of empty space allowing the wealthy to live alone without minority nor majority, which as explained earlier allows the agents to be satisfied relative to their intolerance for non-similar neighbours. Most of the remaining agents do not contribute to segregation, thus further driving the level of segregation down. Additionally, the added space allows the remaining agents more freedom of available resources, which allows for a relatively stable pattern of segregation and agent eviction. Overall this demonstrates how the Schelling model can be expanded to describe much richer and complex interactions between varying hierarchies of prime directives, and the alternate role that group avoidance may play in such structures. Finally, it also shows that economically driven behaviours may help to aid in reducing overall group segregation despite the group populations being highly intolerant of one another.

