

0.1 Variables

Both the Collector and Snatcher agents can log data during the model run. The variables outlined below (refer to Tables 1 and 2) are not limited, meaning more variables can be logged; however, for this paper, these variables would suffice.

This data is timestamped and can be found on our Github page in the Notes.

Variable	Description
AgentID	A unique ID number for each Collector
GoodPointAmount	The amount of good points each agent accumulates every second
BadPointAmount	The amount of bad points each agent accumulates every second
Velocity	The speed of each agent every second
xAxisPos	The position of the agent on the X-axis
zAxisPos	The position of the agent on the Z-axis
seenBySnatcher	The number of times the agent has encountered the Snatcher
touchedWall	The number of times the agent touched a wall
academyScore	The accumulated amount of reward gained by all agents (only for debugging)
date-time	The date and timestamps in which the data is traced

Table 1: Collector variables that are traced during the model run; these are then stored as .csv files for analysis.

Variable	Description
snatcherVelocity	The speed of the Snatcher every second
xAxisPos	Snatcher position on the X-axis
zAxisPos	Snatcher position on the Z-axis
distanceToTarget	The distance between the Snatcher and the Collector it can see
viewAngle	The angle in which the Snatcher is facing every second
touchedWall	The number of times the Snatcher touched a wall
date-time	The date and timestamps in which the data is traced

Table 2: Snatcher agent variables that are traced during the model run; these are then stored as .csv files for analysis.

0.2 Synthetic data analysis

The model-runs, also known as experiments, produce synthetic data for validation and testing purposes. We do not delve too deep into this; however, thought it would be useful for readers to see what type of data is produced and how it can be analysed. All of the code will be available on Github (refer to Notes).

0.2.1 Analysis of data from model-runs one and two

This sub-section describes the synthetic data from the first two model runs; these are results from the parameters in Table 7, model-run one and Table 7, model-run two.

Collectors -1096 and -1148 gather more good points at an early stage in the model run, compared to their peers (bottom left graph in Figure 1). When compared to Collectors in test two (top left in Figure 1), there is a similar increase of good points for all Collectors and the maximum amount of points collected by a Collector is 200 meaning the increase of ray length may have

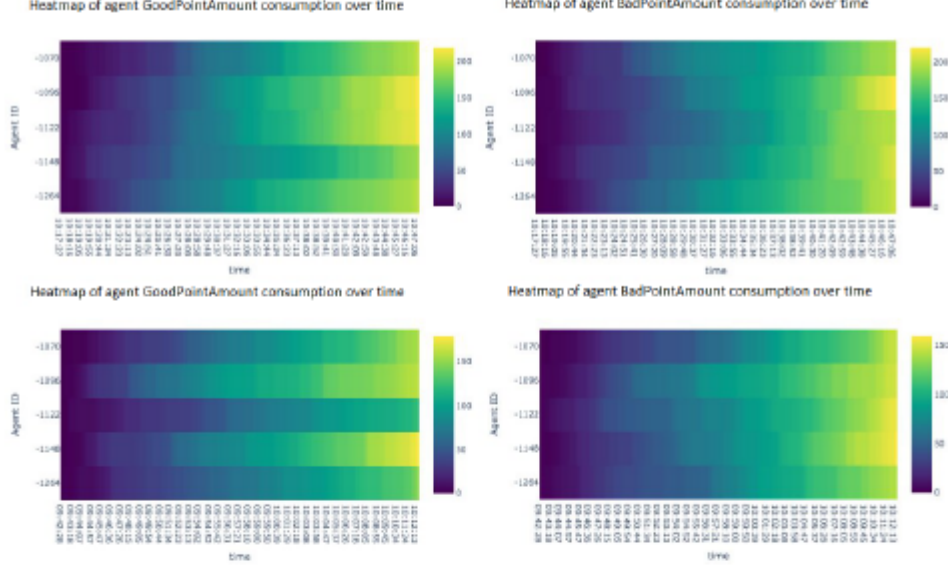


Figure 1: Results of point consumption by Collectors in two different model runs using the neural network trained in 580,000 steps. Top row: Snatcher agent’s viewing angle increased, and Collectors can see further (parameters from Table 7, model-run two). Bottom row: Collectors and Snatcher in the default configuration (parameters from Table 7, model-run one).

allowed Collectors to identify a more significant number of good points objects in the environment.

Collectors in both scenarios in Figure 1 collect bad point objects; this can be due to Collectors moving backwards and often colliding with bad point objects they do not see. It could also mean that Collectors make more mistakes when being chased by the Snatcher, the Collectors would instead run into bad point objects rather than encounter the Snatcher as this would lead to a more substantial penalty.

The Collector -1122 seems to be caught by the Snatcher less often in test one (bottom left in Figure 2) compared to the same Collector in test two (top left in Figure 2). Collector -1122 came into contact with obstacles/walls roughly the same amount of time in both tests; this means that given these two tests, a Collector is more likely to avoid a Snatcher by exploring a range of actions, not solely hiding behind barriers.

Collectors in test two (top row, left graph Figure 2) get caught more frequently than the Collectors in test one (bottom row, left graph, Figure 2) this means that an increase in view length and radius for the Snatcher matches that of the increase in ray length for the Collector. Both agent types can see further, and the Snatcher can see broader meaning Collectors are more likely to get caught regularly.

In Figure 3 , the size of each point represents the velocity. The larger the point, the higher the velocity. Each point represents the position of a Collector

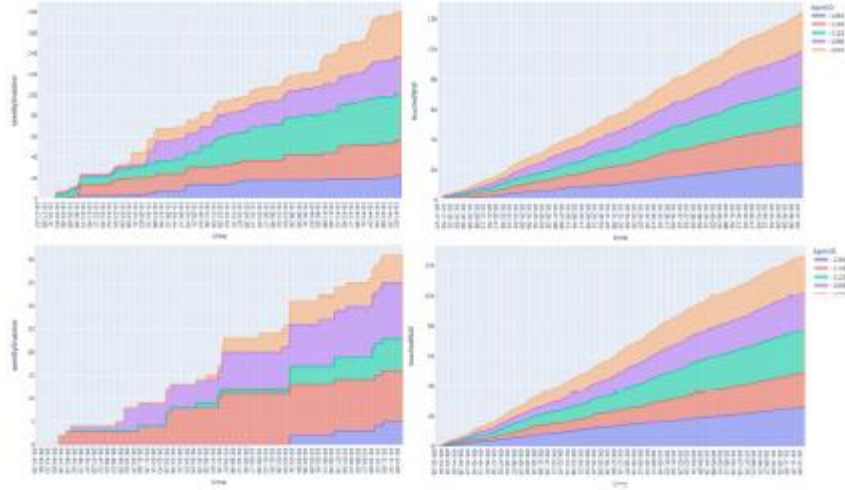


Figure 2: Stacked area plot comparing the number of times the Snatcher has seen Collectors, and the number of times Collectors touched a wall. Top row: parameters from Table 7, model-run 2 and bottom row: parameters from Table 7, model-run 1.

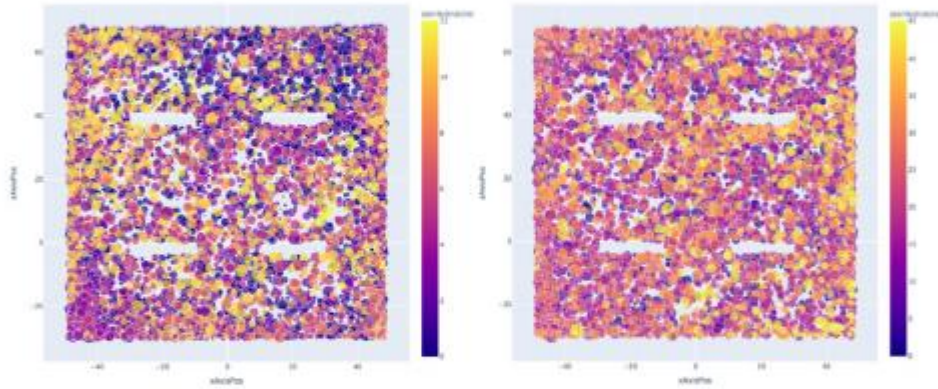


Figure 3: Scatter plots comparing three variables: Velocity, X and Z position and seenBySnatcher. Left: model run using parameters from Table 7, model-run 2, Right: model run using parameters from Table 7, model-run 1.

in time.

It becomes evident in Figure 3 that Collectors tend to be encountered by the Snatcher less in test two (left graph) than the number of times they get caught in test one (right graph). Furthermore, in test two, Collectors seem to get caught more often in the top left area of the environment, whereas, in test one, Collectors get caught more often in the midsection to the right.

We can see that Collectors get caught less around the four barriers in the middle of the environment in test two (left graph in Figure 3), compared to test one (right graph in Figure 3).

The Snatcher seems to raise its speed more often in test two (top row, left

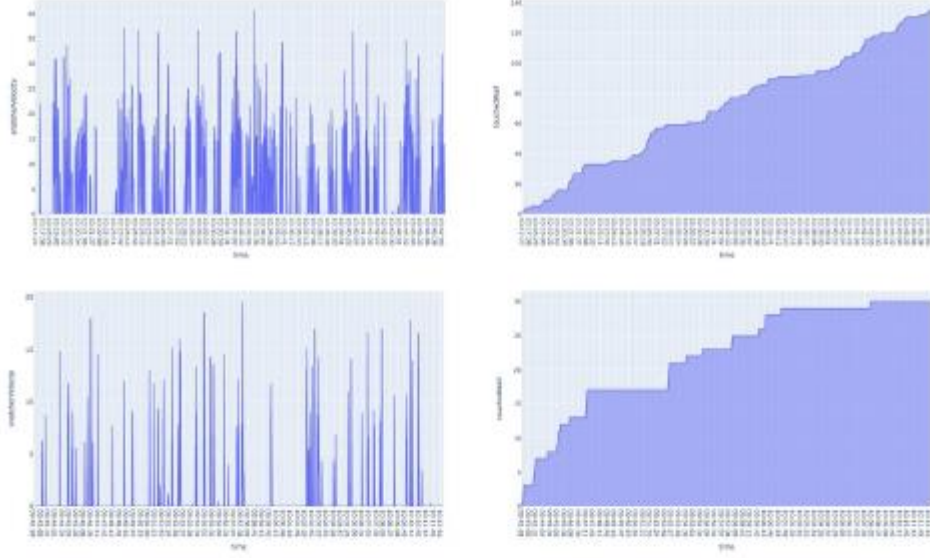


Figure 4: Area plot graphs comparing the Snatcher velocity and number of times a Snatcher touched a wall. Top row: Snatcher in test two and bottom row: Snatcher in test one.

in Figure 4) compared to test one; this could mean the Snatcher's ability to identify and chase Collectors is improved in test two due to the increase in view radius and viewing angle.

In Figure 4 , the Snatcher in test one (bottom row, right graph) touches walls/barriers less often than the Snatcher in test two (top row, right graph), this means that an increase in ray length for Collectors does not necessarily mean they hide more often.

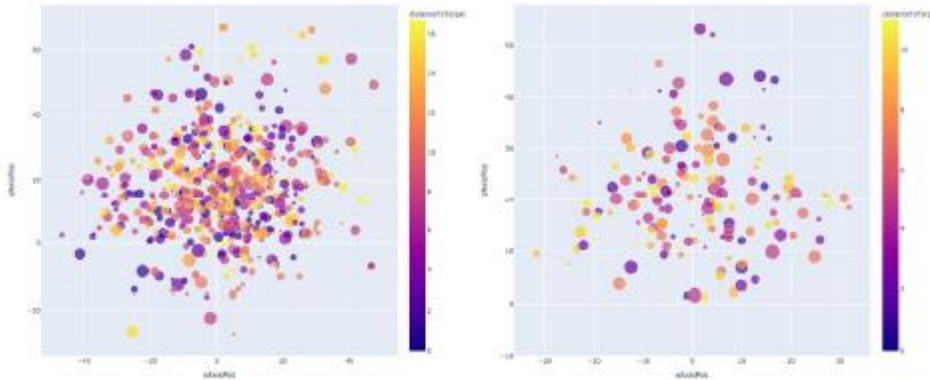


Figure 5: The velocity, position and distance to target variables for the Snatcher in test one (right graph) and test two (left graph).

It is immediately apparent that the Snatcher in test two explores more of the environment than the Snatcher in test one (refer to Figure 5).

It seems like the distance to a Collector is more abundant on the outskirts of the environment in test two compared to test one where the Snatcher tends to spot Collectors in the centre (refer to Figure 5).

The velocity of the Snatcher in test one increases as the distance to the Collector decreases; this is the case in test two also (refer to Figure 5).

When Figures 2 and 3 are compared, the collector -1070 has been caught more frequently, i.e. the total number of times this Collector is caught is 45 in test one, Figure 2 . However, in test two, there are more Collectors in total that have been seen/caught by the Snatcher whereas this value is much smaller in test one; this means there is a small number of Collectors getting caught multiple times in test one but more Collectors getting caught less frequently in test two.

0.2.2 Analysis of data from model-run three

In this sub-section, the data captured using the parameters from Table 7, model-run three is illustrated. The only difference between model-run one, two and three is the duration of the training process. The neural network used in this test was trained over one million time steps compared to runs 1 and 2, which were trained in 580,000 time-steps.

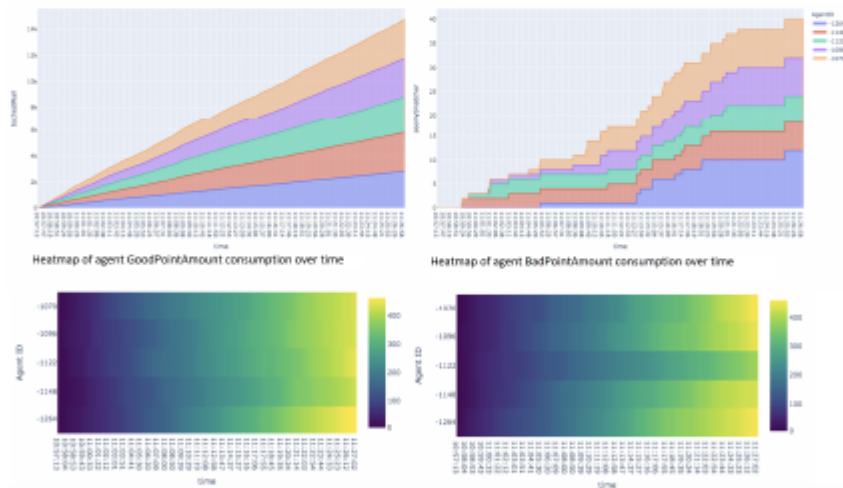


Figure 6: Collector data from model-run three visualised. Bottom row: good points and bad points consumption, top row, left: number of times Collectors touched a wall/obstacle over time, top row, right: number of times Collectors have been seen/caught by Snatcher over time.

Collectors in model-run three are gathering more point objects than the Collectors in model-runs one and two (refer to Figures 1 and 6). Therefore, training Collectors longer is more advantageous as they become better at collecting points over time.

Collector -1122 collects up to 500 good point objects and during this time only collects about 300 to 400 bad point objects. Therefore, Collector -1122

outperforms its peers (refer to Figure 6).

Collector -1264 collects over 500 good point objects; it encounters the Snatcher the least number of times and comes into contact with barriers/wall the least amount of times (refer to Figure 6). Therefore, Collector -1264 must be using alternative means of avoiding the Snatcher while optimising its resources.

Collectors that have encountered the Snatcher more frequently tend to be positioned around the outskirts of the environment and a few in the midsection (refer to Figure 7 , left graph).

When comparing Figure 3 (left graph) with Figure 7 (left graph), we can see that the highest number of times a Collector encounters a Snatcher is 12. As the scenario in Figure 3 was trained over 580,000 time-steps, and the scenario in Figure 7 was trained for one million time-steps. An improvement in the Agent abilities, like increased view length (test two), can lead to a similar effect on Collectors that have trained for a more extended period. Moreover, Collectors in test three seem to increase their velocity more frequently than those in test two; this could mean that a more extended training period provides Collectors with more knowledge about the environment meaning they are less hesitant to increase their speed when moving around.

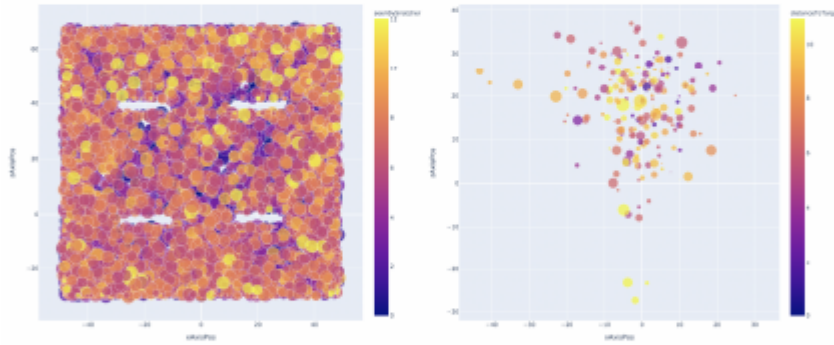


Figure 7: Visualisation of Collector and Snatcher data. Left: the position of Collectors (points), velocity (size of points, larger size indicating higher velocity) and number of times encountered by Snatcher (colour scale). Right: position of Snatcher, velocity and distance to Collector.

The Snatcher seems to have spotted Collectors more frequently in Figure 3 model-run two, compared to model-run three (Figure 7 , right graph).

The Snatcher agent in model-run one and three seem to have similar trends; both scenarios have a maximum distance to the target of 11 (Figures 3 , right graph and 7 , right graph).

The Snatcher encounters walls/obstacles roughly 22 times during the model run (Figure 8), compared to tests one and two (Figure 4), the Snatcher in test three avoids walls/obstacles more frequently.

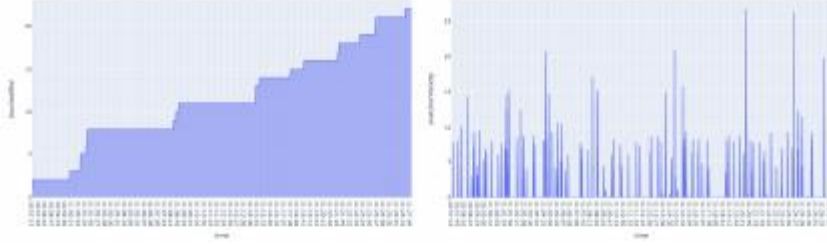


Figure 8: Left: the number of times Snatcher encountered walls/barriers over time. Right: Snatcher velocity over time.

0.2.3 Analysis of data from model-runs four and five

This section describes the results from model-runs four (where Snatcher is not present) and five (where Snatcher is present). The training process of the model did not include a Snatcher; therefore, testing the model with a Snatcher may provide some insight as to how Collectors adapt to situations they were never trained to encounter.

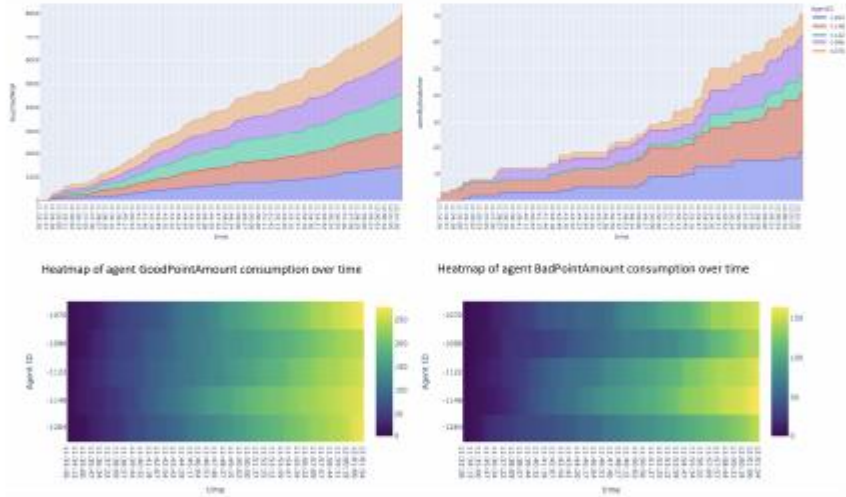


Figure 9: Collector data from model-run five visualised. Bottom row: good points and bad points consumption, top row, left: number of times Collectors encounter a wall/obstacle over time, top row, right: number of times Collectors have encountered the Snatcher over time (parameters from Table 7, model-run five).

The Collectors in test five seem to have encountered and consumed more good point objects than bad point objects when compared to previous tests (refer to Figure 9). Furthermore, the Collectors are encountering the Snatcher more frequently in this scenario; this was expected as they never trained to comprehend the existence of a hostile actor. When comparing model-runs 2 and 5, Collectors encounter the Snatcher more in model-run two than they do

in model-run 5 (refer to Figures 2 and 9).

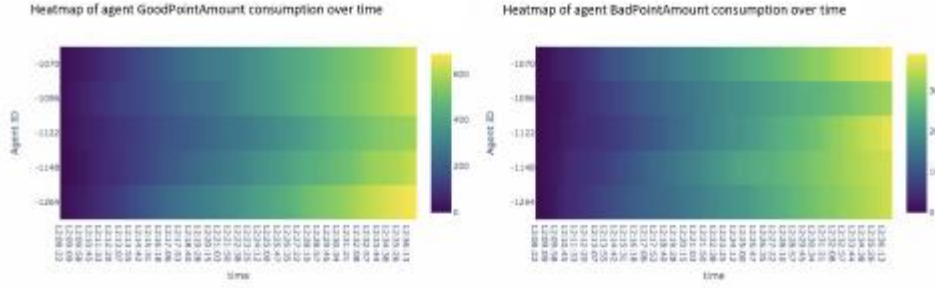


Figure 10: Results of point consumption by Collectors in model-run four using the neural network trained in one million steps. Left: Collector’s good points consumption, right: Collector’s bad points consumption (parameters from Table 7, model-run 4).

The maximum amount of points consumed by a Collector is 700 in model-run four (Snatcher not present), and the amount of bad points consumed is 400 (refer to Figure 10).

The Snatcher being present impacts the amount of point consumption when comparing Figures 9 and 10. Collectors do better when the Snatcher is not present.

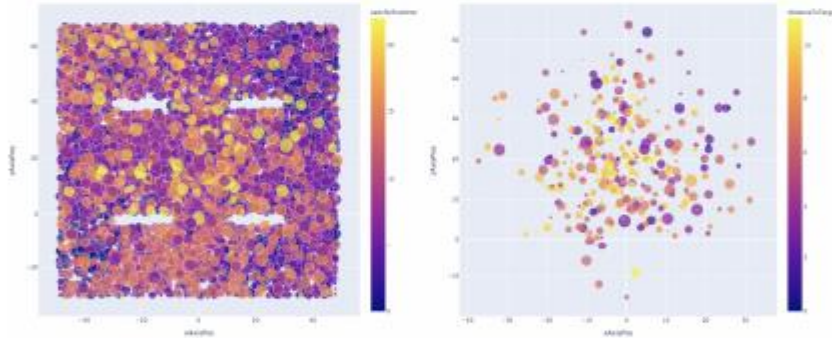


Figure 11: Visualisation of Collector and Snatcher data from model-run five; left: the position of Collectors (points), velocity (size of points, larger size indicating higher velocity) and the number of times encountered by Snatcher (colour scale). Right: the position of Snatcher, velocity and distance to Collector (parameters from Table 7, model-run 5).

The Snatcher seems to encounter Collectors more frequently in the centre region of the environment (refer to Figure 11). Collectors who travel on the outskirts of the environment tend to avoid the Snatcher.

Collectors move very fast throughout the environment (Figure 11) when compared to Collectors in model-run two (Figure 3, left graph) and explore more surface area.

The distance to the target decreases as the Snatcher moves out from the centre of the environment. The four barriers separating the middle section from the top and bottom sections of the environment may have caused this. Moreover, Collectors get spotted more often in the middle section (between the four barriers) than they do on the outskirts (refer to Figure 11, left graph).