Robot Swarm Decision-Making:
Simulation of Kilots "Bees"

Cecilia Aponte

Sapienza Universita di Roma

Master of Artificial Intelligence and Robotics

Elective in Robotics: Control of Multi-Robot Systems

March 21, 2020

1 Introduction

Swarm robotics studies systems composed of a multitude of interacting robotic units. They can be composed of the same or different types of simple robots working together through communication towards a shared goal. Their collective behavior is decentralized due to the limitations of perception and action of each separate robot component. The information communicated between each robot is both limited by the amount of information that can be sent and by the distance it can reach. So the agents have access only to limited local information and share this in order to reach their collective goal.

Swarm robotics often takes inspiration from natural systems. For instance, collective decision making in robot swarms can be designed by looking at the nest site selection behaviour of honeybees. When a hive becomes overcrowded, the colony usually splits in two and must search for a new hive to move. The selection of the new hive involves a collective decision-making process. Hundreds of scouting bees leave the nest to explore the area for new prospects and return with information about the quality and location of the new potential hive, which is communicated to other scout bees a waggle dance. However, when two sites of equal quality are found the bees have to make a collective decision. Biologist Thomas Seeley from Cornell University investigated this phenomena and noticed during experiments that bees would not only communicate about the hive they found but also would disrupt other bees from completing their dance by head-butting them and emitting high-pitched beeps, as seen in Figure 1. These inhibitory signals help break the deadlock and allow to choose a nesting site faster. Once the group reaches a quorum, which is the minimum number of bees that have to be committed to a hive, the decision is made to move to that hive.

Figure 1: A bee performing head-butting to competitor bee so it will stop their waggle performance, which spreads their chosen hive information to other bees.

Swarm robotic systems use bees as an example to design decision-making in an artificial swarm. The paper "Effects of Spatiality on Value-Sensitive Decisions Made by Robot Swarms" [2] addresses this problem using a swarm of kilobots, which are low-cost swarm robots shown in Figure 2, that have to choose between two options. This project then explores the behavior of kilobots through simulations to provide a qualitative comparison with the experiments performed in the above-mentioned paper, and to explore the ability to arrive at a collective decision by varying different parameters regarding the problem setup and the mobility pattern of the robots.

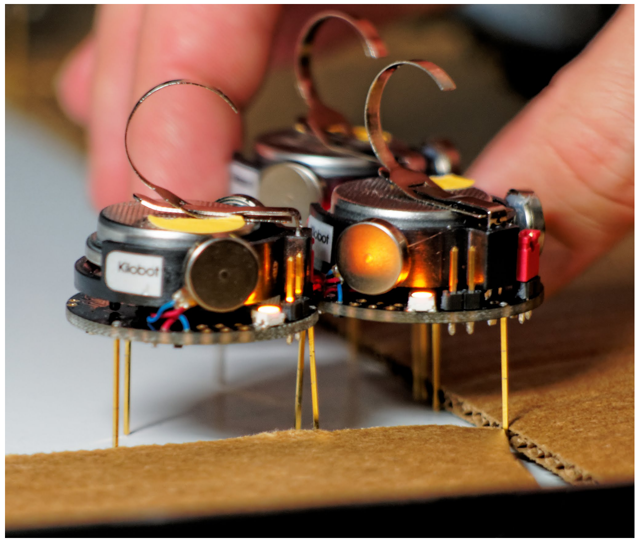


Figure 2: Kilobot are low-cost robots used in a multitude group as swarms.

2 Problem

The goal of the study is understanding the effects of spatiality into a decision-making process inspired by the honeybees nest-site selection behaviour. This problem consists in choosing between two target sites each characterised by a quality value where a higher value describes a better target. These values provide important information when a decision has to be made by a spatially dispersed swarm, especially when the targets might need different times to be discovered. The information about low-quality options should spread slowly, granting time to the robots to possibly find better-quality options.

An easy and fast decision can be made when the options have different quality values, where the target with the highest value would win. However, a more complex and interesting scenario is when both targets have the same quality value, which can lead the swarm to a deadlock situation. In this case, real bees spread information through the waggle dance and disturb the spread of information about the competing target using the methods of head-butting and emitting high-pitched beeps, as mentioned above. Likewise, kilobots and the simulated agents likewise use a decision-making process to both simulate the spread of data and the disturbance of such through four decision-making processes: discovery, recruitment, abandonment and cross-inhibition.

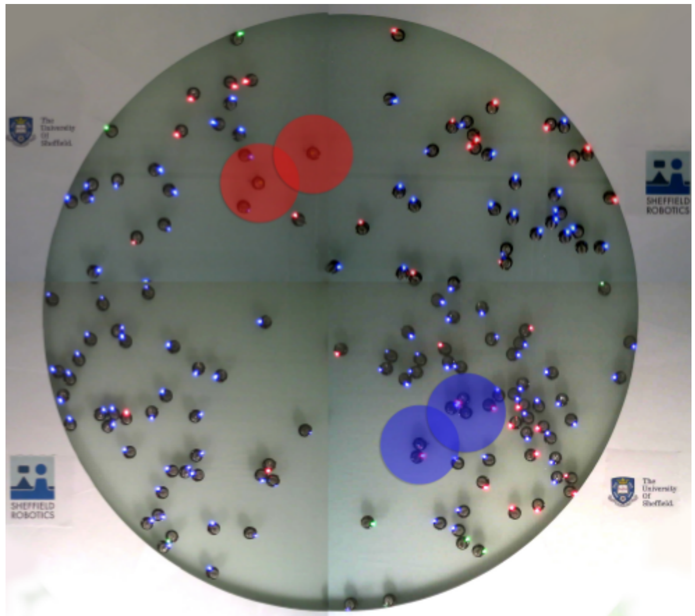


Figure 3: Kilobot choosing between two target hives that have the same quality value.

The kilobot experiments, as shown in Figure 3, consists of a circular environment of $A_E = 1.77m^2$ and two targets with two beacons each that communicate their quality value to any agent that is inside the area ($A_O = 0.058m^2$). The targets are located $0.76m$ distance between each other. Their quality values for both targets are the same, in order to investigate the progression of decision-making.

There are 150 kilobots; they can have three possible status: uncommitted, committed to the red target or to the blue target. They display their status through the RGB LED that each kilobot has as black, red, blue colors, respectively. Only when the decision quorum (70%) is reached, consisting of a specific percentage of kilobot agents with a commitment to one of the targets, the decision is made for with the target corresponding to the quorum.

Each kilobot agent moves on a flat surface at a speed of $13\,mm/s$ and rotation of $40\,°/s$ on average. It uses random walk by altering a straight movement and on-place rotations done randomly. This method lets uncommitted robots to explore the environment to discover options and to mix and exchange information with their neighbors. They can communicate 9-bytes messages between neighboring kilobots that are within a $0.1m$ distance range.

The decision made by the collective of these agents depends on having a sufficiently high quality value (above a given threshold θ_v). If the quorum of agents is not reached, the agents keep exploring the area for other options until it breaks the decision deadlock or remain stuck in indecision. The agents do not have prior information - amount of options available, their locations, and quality - about the decision problem they have to solve. The swarm explores the environment to identify the options and selects the option with highest value.

With these criteria, a macroscopic model is created at [2] to describe the value-sensitive decision-making inspired by the nest-site selection behaviour of honeybees. This model will be described in the next section.

3 Model

The model used to solve the problem follows a design pattern that links individual agent behavior rules to the macroscopic model parameters. The agent behavior rules have to include:

- explore the environment to search for available options
- recognize available options once found
- estimate options' quality
- exchange with other agents the options' ID and quality

The agents can commit or uncommit to a target option. There are four transitions that an agent could go through in order to make a decision:

- $Discovery(\gamma)$: an uncommitted agent explores the environment to search for available options and upon *discovery*, gets committed with probability P_γ
- $Recruitment\,(\rho)$: an agent committed to one option *recruits* uncommitted robots with probability P_ρ
- $Abandonment(\alpha)$: an agent committed to one option spontaneously *abandons* its commitment and reverts to being uncommitted with probability P_α
- $Cross - Inhibition\ (\beta)$: an agent committed to an option sends a stop signal to an agent committed to a different option. The latter becomes *inhibited* and returns uncommitted with probability P_β

At each time step in the simulation, robots broadcasts information to neighbors about their committed state, the option and quality if committed. The decision process is performed every 10 time steps in the simulation (corresponding to one second) to update the commitment states of the agents. Since the agents broadcast information every time step, they also saves the information received from its neighbors at each time step. Therefore when a decision can be made, the agent always has a random, saved neighbor information needed for recruitment and cross-inhibition if there are no current neighbors in that moment. Similarly, when an agent encounters a target option it saves the target information (keeping always saved the most recent target option's information) and processes when a decision can be made even when the agent is not any more on a target's vicinity.

To build the model, a macroscopic and microscopic parameterisation are used which are mediated from the original paper [2] :

3.1 Macroscopic Parameterisation

A macroscopic model is built to describe the decision process. It uses ordinary differential equations (ODE) that describe the changes in the proportion of agents committed to each of the two available options. The model is described as:

$$dx_1 = (\gamma_1 x_U - \alpha_1 x_1 + \rho_A x_1 x_U - \beta_2 x_1 x_2)dt \tag{1}$$

$$dx_2 = (\gamma_2 x_U - \alpha_2 x_2 + \rho_B x_2 x_U - \beta_1 x_1 x_2)dt \tag{2}$$

$$x_U + x_1 + x_2 = 1 \tag{3}$$

where as previously stated, γ_i, α_i, ρ_i, β_i corresponds to the transition rates of discovery, abandonment, recruitment, and cross-inhibition respectively; the subscript corresponds to the target option number $1, 2$. The parameterisation used is given by:

$$\gamma_i = \nu_i \cdot P_D, \quad \alpha_i = \nu_i^{-1}, \quad \rho_i = \nu_i, \quad \beta_i = \beta, \quad dt = d\tau/s, \quad i \in 1, 2 \tag{4}$$

where ν_i is the quality value of each target option which is the same value for both options (either 1.5 or 5 for the experiments); γ_i takes into account the episodic nature of a discovery, which happens only when agents physically move over the target (hence approximately $P_D = A_O/A_E$)

The macroscopic stability depends of the quality ν and the cross-inhibition rate β. As seen in Figure 4, the behavior of the macroscopic model is analyzed to determine the parameters when a deadlock breaking occurs so that a decision can be made by reaching the necessary quorum. With this analysis the chosen value in the paper [2] was $\beta = 0.3$ and $\nu \geq 1.5$; however for this project, we chose a better parameterisation inspired from another paper [2] which ensure better convergence even when there are more than two options. In this new parameterisation., the value used for β is the same ν target value.

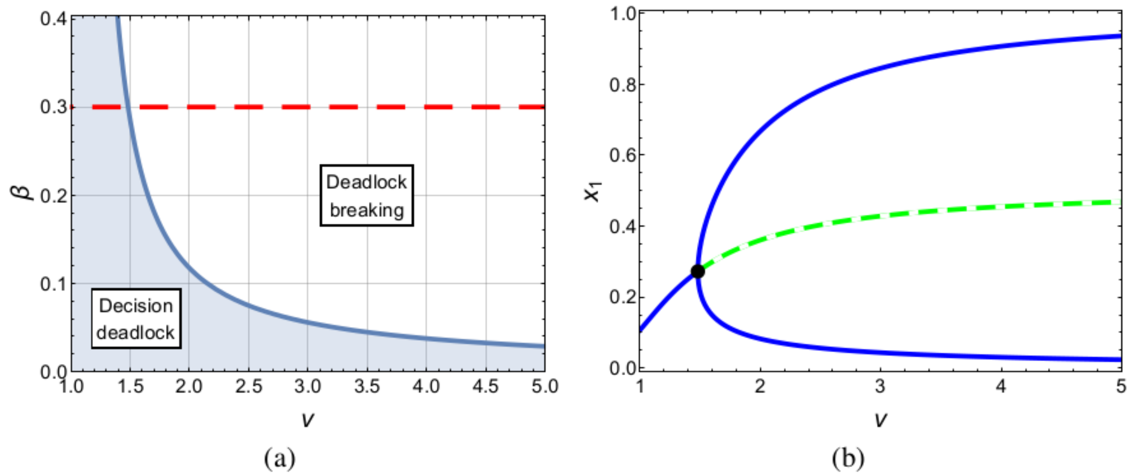


Figure 4: Analysis of behavior of macroscopic model. (a) Stability diagram comparing ν to β which will create a decision deadlock or deadlock breaking. $\beta = 0.3$ is chosen. (b) Shows the stability of the equilibria, where the blue lines correspond to the stable and the green to the unstable equilibrium. The decision deadlock occurs for quality values greater than about 1.5.

3.2 Microscopic Parameterisation

The macroscopic model is then converted to individual robot probabilities, creating the microscopic model. To convert it to the agent implementation, the temporal domain is added since the kilobots operate in discrete time. The agents each have to be converted as follows:

$$P_{\gamma_i} = \gamma_i \cdot T, \quad P_{\alpha_i} = \alpha_i \cdot T, \quad P_{\rho_i} = \rho_i \cdot T, \quad P_{\beta_i} = \beta_i \cdot T, \quad i \in 1, 2 \tag{5}$$

where T corresponds to the update timestep length, which is determined as $T = \tau_c \tau_u s$. The clock period is given by $\tau_c = 0.1$, the decision step given by $\tau_u = 10$, and timescale given by $s = 0.1$. This creates the update timestep length to be equal to $T = 0.1$.

4 Simulation Implementation

A simulation was adapted to provide a closer resemblance between the kilobot experiments and those generated by the simulation. All parameters specified in the paper were used exactly, except those specified above for any small deviation. As seen in Figure 5, the GUI of the experiments looks alike that of the kilobot experiment shown earlier. The circular area of the entire environment shows all the 150 agents, which at the beginning are all black since they are uncommitted and located at random; and the two target areas, which are the colored circle either green or purple. They are spaced initially for the experiments at the same distance specified in the paper as 0.76m; later this distance is changed to analyze the result of target options that are in closer or farther proximity. Location of the target options are random, but always around the center of the environment and around equal in distance and opposite in direction.

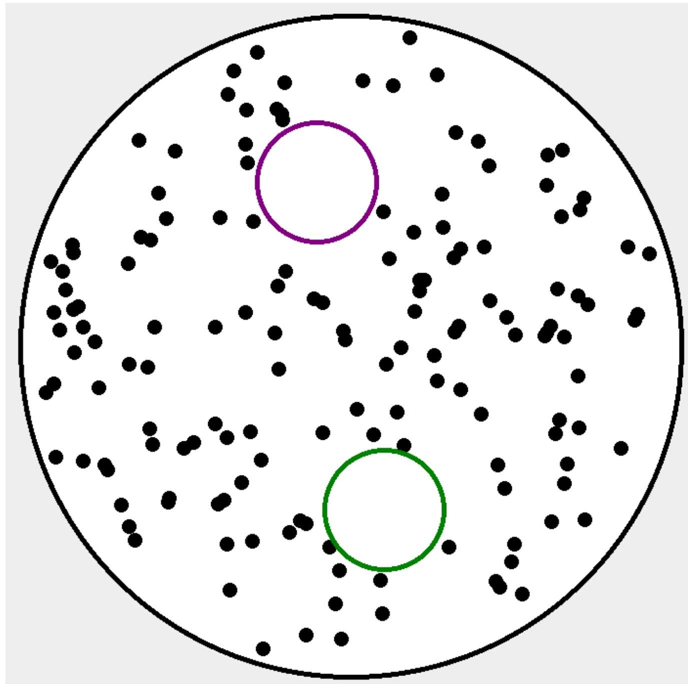


Figure 5: Diagram shows the simulation GUI at start, where each agent is uncommitted and therefore color black. The two target areas are defined by the circles of purple and green.

As the run starts, the agents that are located inside the area of a target option can discover it. If it is not time to make a decision, since this is done only every 10 seconds, it saves the information of the target in order to process it during decision time. However if it is time to make a decision and the probability of it is greater than a random value, the agent gets committed to that target option and keeps its color and discovery value. This discovery phase is seen in the first two top images in Figure 6, in the initial phase. Already at this initial phase for this particular experiments, the image on the left corresponding to $\nu = 1.5$ shows less committed agents for both targets compared to the image on the right corresponding to $\nu = 5$

When there are some agents that have been committed to a target option, there can be other decisions that can be made such as recruitment, abandonment, and cross-inhibition. Recruitment is addressed to an agent that is not yet committed. During each second, the agent saves a random neighbor agent that could be committed or not; if the recruitment probability is higher than a random value, the agent takes the commitment of that neighbor (target option 1 or 2).

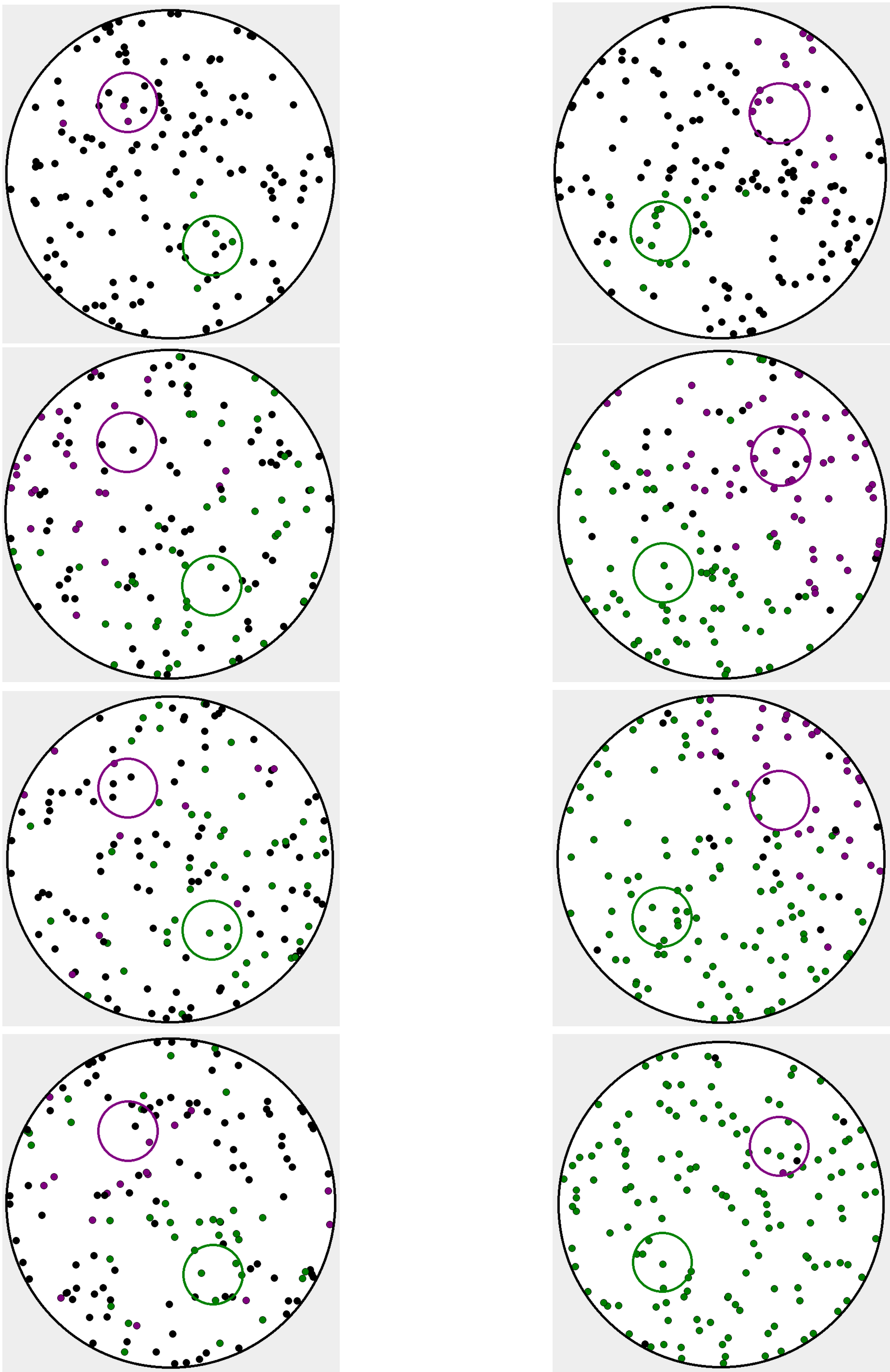


Figure 6: Diagrams on the left corresponds to results at $\nu = 1.5$ and on the right at $\nu = 5$. Each diagram shows agents committed to any/no target option (blue, red, black). Vertically from top to bottom shows simulation at the: start, middle, possible quorum time, and end (1, 15, 22, 30 min.).

Similarly, for cross-inhibition the agents will be interacting with one of its neighbors. This simulates the head-butting mechanism of the bees where one agent committed to one option makes the other agent committed to the opposite option stop communicating its information. If the sum probability of cross-inhibition is higher than a random value, the agent that is inhibit ted by another random neighbor agent of the opposite option will make the agent become uncommitted. Likewise, an agent will abandon its commitment if its sum probability is higher than that random value.

As the experiment is running, as shown in the second row of images in Figure 6, the agents normally change from either option 1 or 2 or uncommitted. As they continue to mix in the environment, the agents will tend towards either option until the quorum is reached and a collective decision can be made by the option that has this majority. This can be seen in the third row of images in Figure 6, where a majority of 70% of agents have picked the green target option in the right image when $\nu = 5$; yet the left image for $\nu = 1.5$ has not yet reached quorum and continues to mix to try to reach a decision.

For $\nu = 1.5$, the experiment would continue to run until the deadlock is broken (if ever). The left image in the fourth row of images in Figure 6 shows how even after 30 minutes the run has not reached quorum or made a decision. On the other had, the image on the right for $\nu = 5$ shows that almost all agents have reached the same decision; the decision was reached previously at quorum, however for analyzing the results shown in the next section, the experiments were left to run even after the quorum is reached. It would only be finalized when 30 minutes have passed.

5 Results

All experiment runs were left to run for a constant time of 30 minutes even after the quorum is reached. This is done to analyze correctly and equally to every run. For each experiment, 100 runs were done to get an average behavior of the effects of the change in parameters. Every 2 minutes data was saved, including the number of robots committed to option 1 or 2 and uncommitted.

5.1 General Results

Firstly, comparing the original paper's results of the kilobot and simulation experiments and the results of this simulation experiments. Figure 7 shows the comparison between these two experiments when using the quality of 1.5. The overall trend of both are alike where both target options have a constant amount of agents committed throughout time. It starts at low quantity until it stabilizes for both around 35-45 agents for blue (i.e., the option with the higher number of committed agents) and 20-30 agents for red (i.e., the option with the lower number of committed agents) on average for both. Similarly, the uncommitted agents also start at a higher quantity and stabilizes at 75-80 agents on average. This means that if the quality of the target options are low, then the agents have a weaker decision. This makes none of the options standout distinct to the other and no option picked, or one picked by a small margin difference.

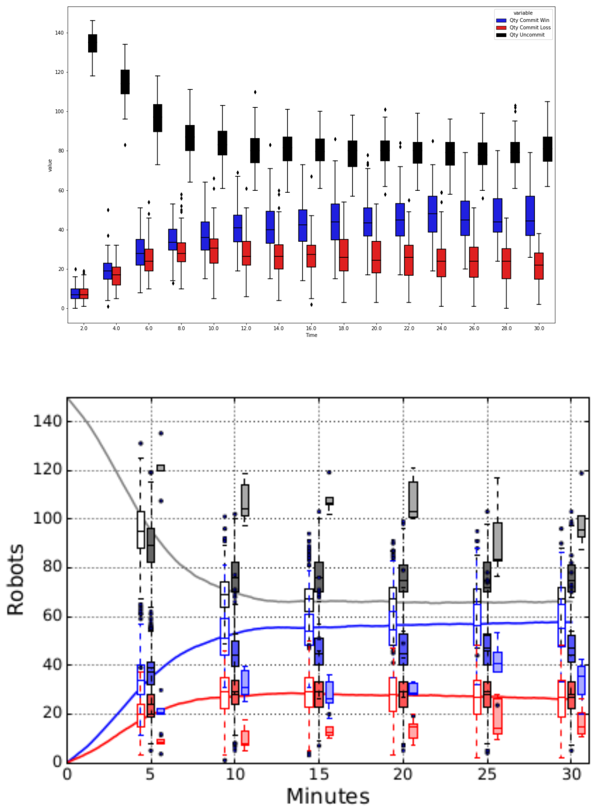


Figure 7: Diagrams corresponds to results at $\nu = 1.5$. The plot on the top corresponds to current simulation results and the bottom from the original paper. Each plot shows amount of agents committed to the target option that won (blue) and the other (red) with time.

Figure 8 shows the comparison between these two experiments when using the quality of 5. The overall trend of both are alike where both target options have an increase in the blue option and decrease of the red option throughout time. For blue, it starts on average at 80 agents and increases until on average 100 agents. For red, starts on average 50-60 agents and decreases until on average 30-40. Similarly, the uncommitted agents remains constant on average 10-20 agents. This means that if the quality of the target options are high, then the agents have a stronger decision. This makes one of the options standout distinct to the other and become the option picked.

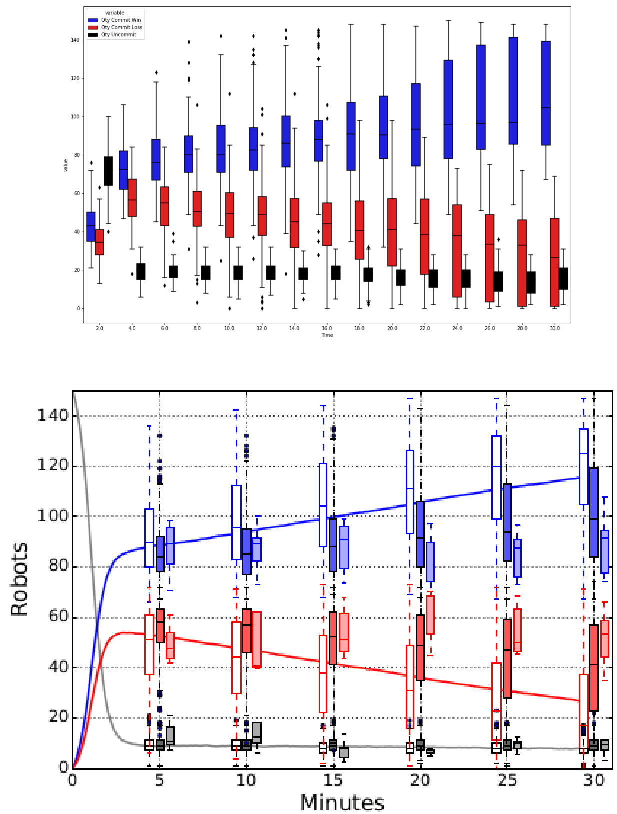


Figure 8: Diagrams corresponds to results at $\nu = 5$. The plot on the top corresponds to current simulation results and the bottom from the original paper. Each plot shows amount of agents committed to the target option that won (blue) and the other (red) with time.

5.2 Results from Experiments with Changing Distance between the Target Options

Secondly, to compare the same plots with increasing the distance between the two target options. Figures 9, 10, and 11 shows the set of increasing distance respectively: 0.28, 0.36, 0.46, 0.56, 0.66, 0.76, 0.86, 0.96, 1.06, 1.16, and 1.22. It shows on the left all plots for $\nu = 1.5$ and on the right for $\nu = 5$. The overall trends maintains consistent to what is expected to see given the original paper's results.

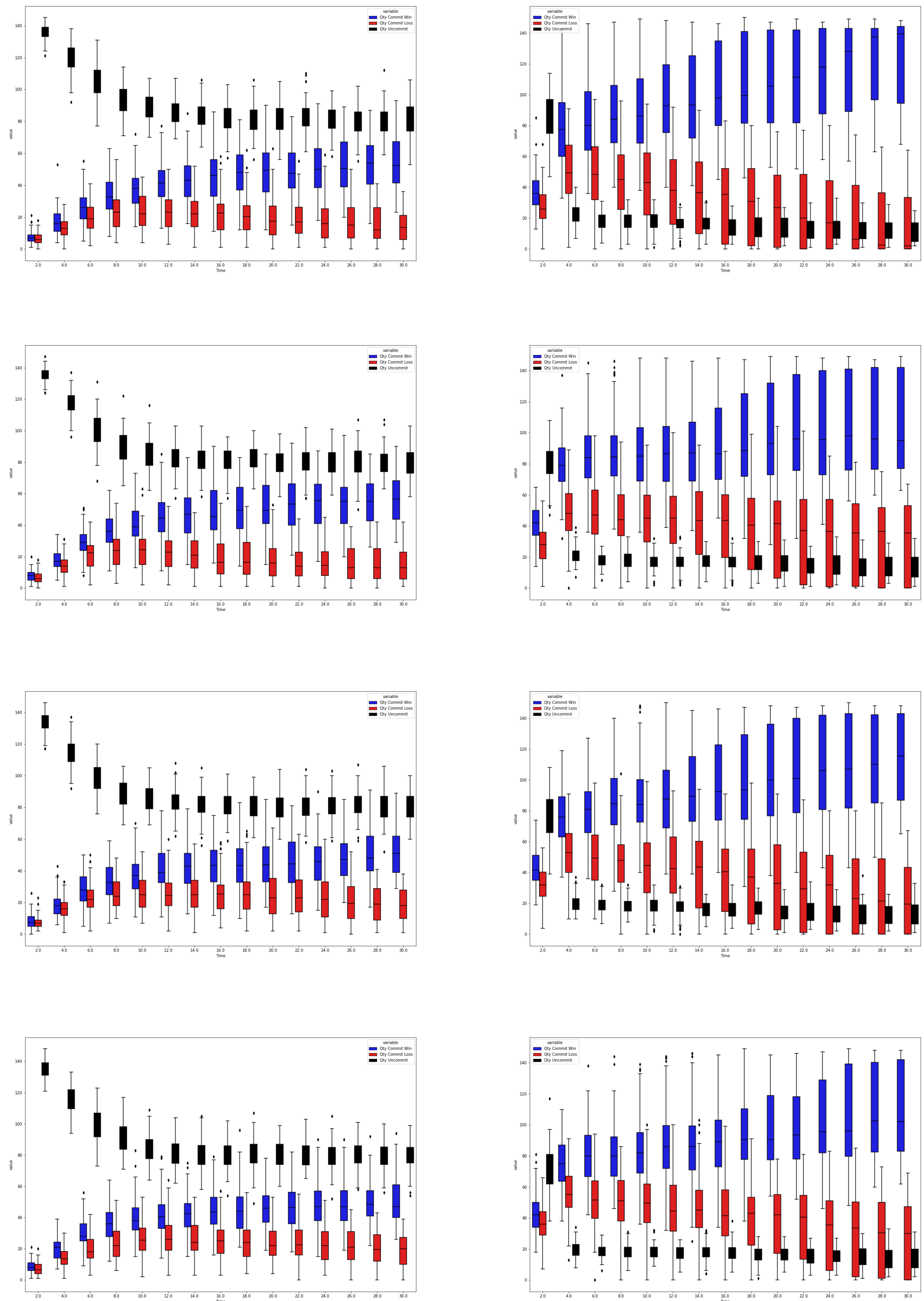


Figure 9: Diagrams on the left corresponds to results at $\nu = 1.5$ and on the right at $\nu = 5$. Each diagram shows amount of agents committed to the target option that won (blue) and the other (red). Vertically down corresponds to changing the distance from top to bottom: 0.28, 0.36, 0.46, and 0.56.

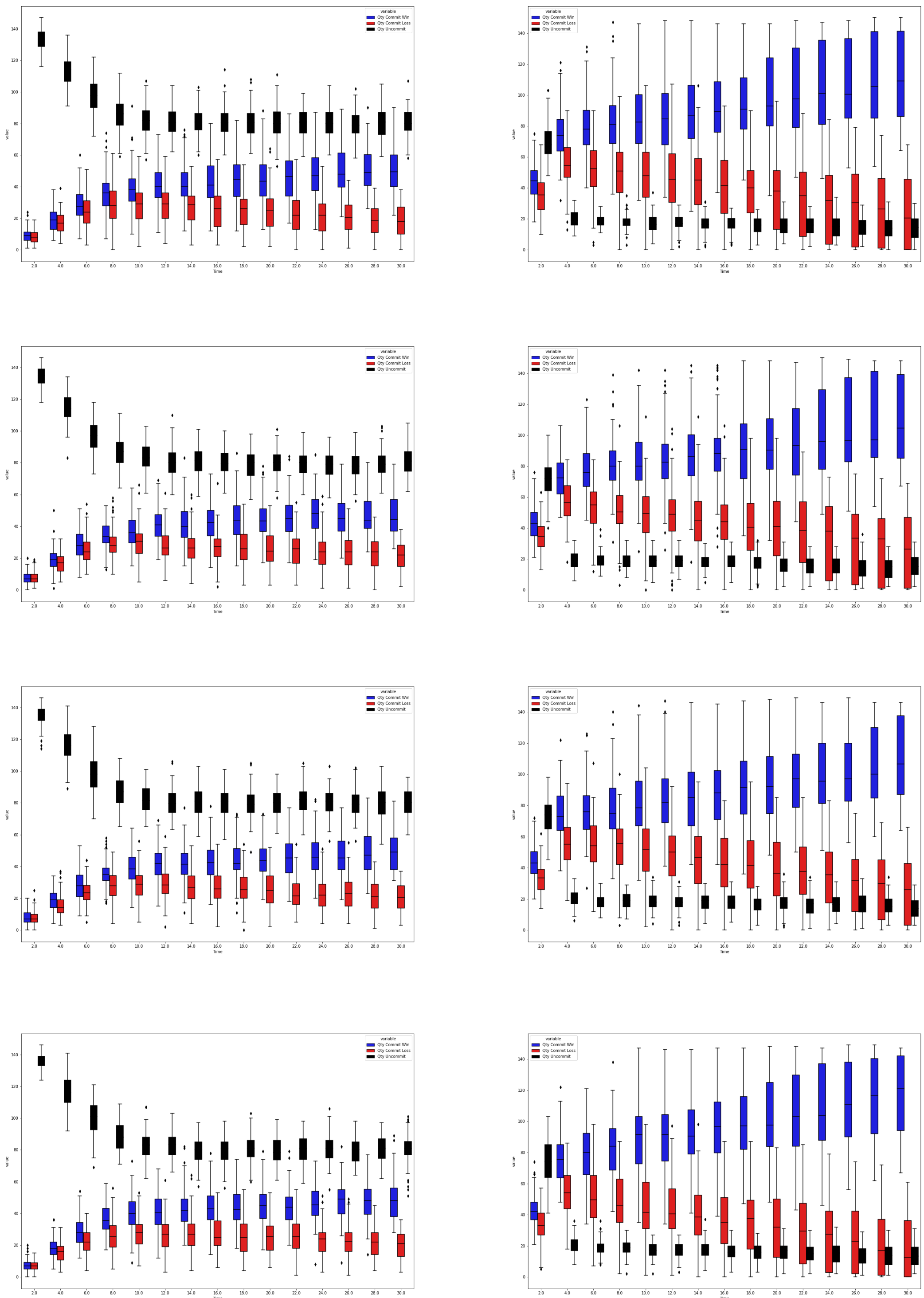


Figure 10: Diagrams on the left corresponds to results at $\nu = 1.5$ and on the right at $\nu = 5$. Each diagram shows amount of agents committed to the target option that won (blue) and the other (red). Vertically down corresponds to changing the distance from top to bottom: 0.66, 0.76, 0.86, and 0.96.

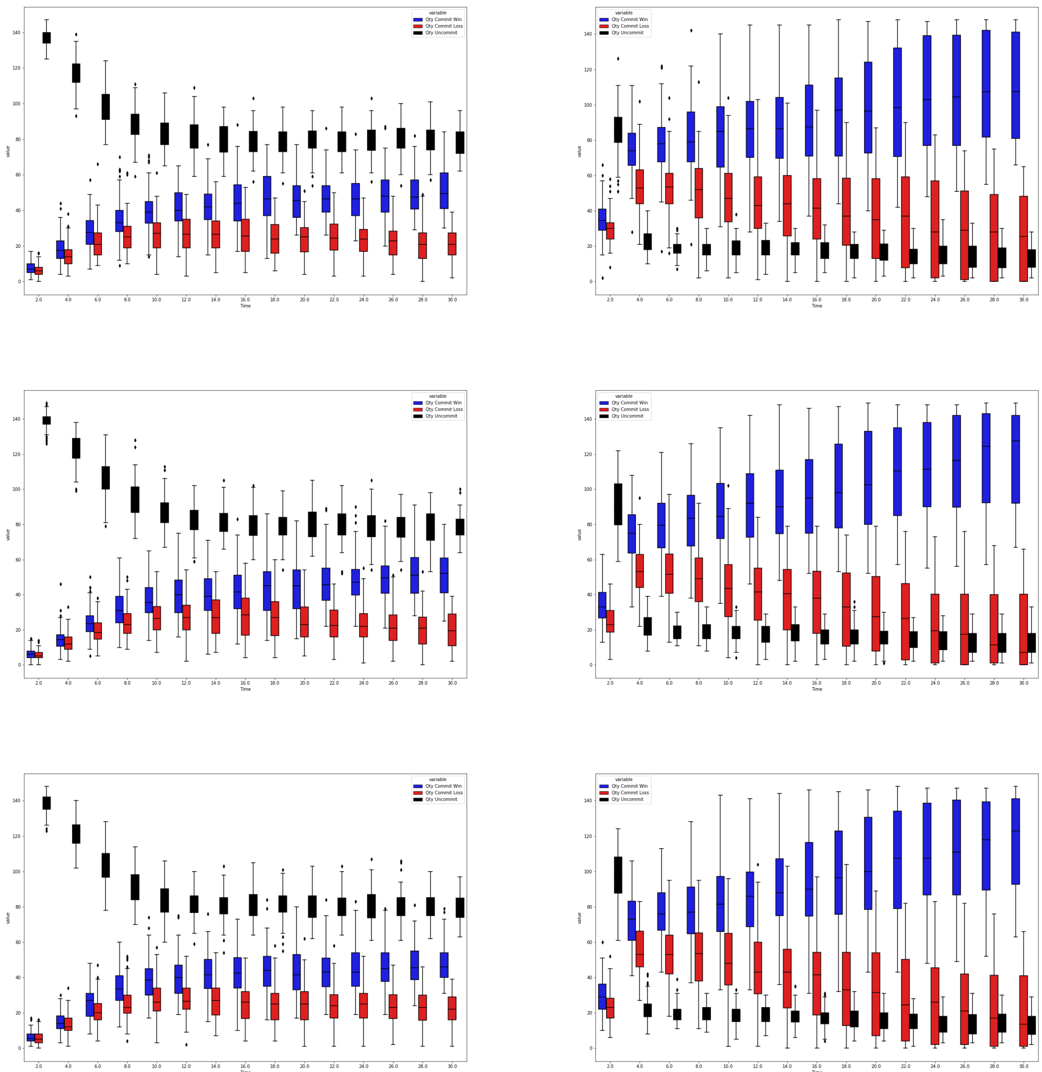


Figure 11: Diagrams on the left corresponds to results at $\nu = 1.5$ and on the right at $\nu = 5$. Each diagram shows amount of agents committed to the target option that won (blue) and the other (red). Vertically down corresponds to changing the distance from top to bottom: 1.06, 1.16, and 1.22.

For the diagrams comparing the results when changing the CRW and Levy exponents, go to the Appendix. Figures 21 and 22 show these behaviors using a constant distance of 0.76 and $\nu = 5$.

To analyze better these diagrams, plots were merged to show how the behavior changes with increasing CRW and maintaining everything else constant. Figures 12, 13, and 14 show this analysis for $levy = 1.2$, $levy = 1.6$, and $levy = 2$ respectively. In general, the plots seems to show that as CRW increases from 0 to 0.6 the upper quartile (75%) increases at each time saved; while the lower quartile (25%) decreases at each time saved. For all $CRW = 0.9$ does not show any trend compared to the rest.

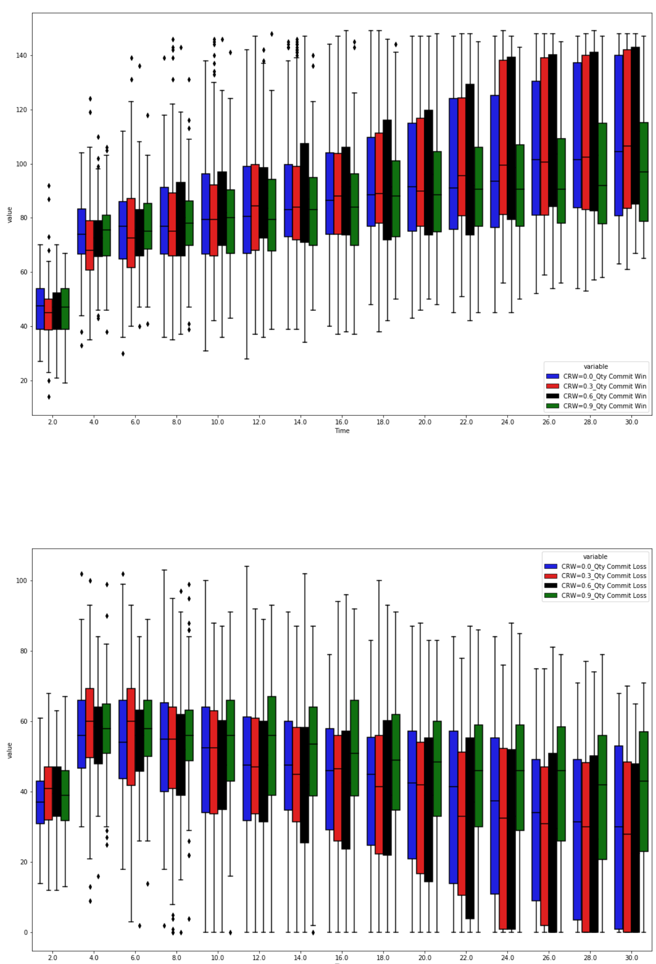
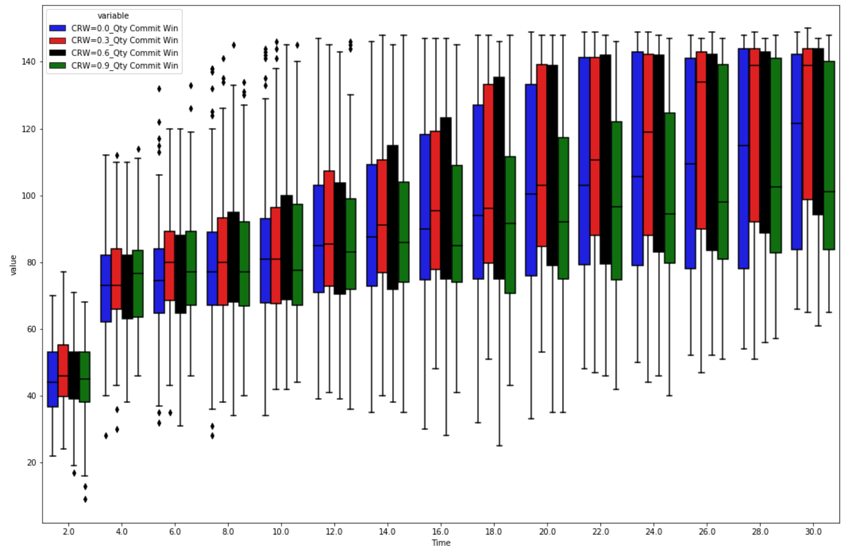


Figure 12: Diagrams are at $\nu = 5$, $distance = 0.76$, and $levy = 1.2$. Both diagrams shows amount of agents committed to the target option that won (top) and the other (bottom) with increasing CRW (0, 0.3, 0.6, 0.9).



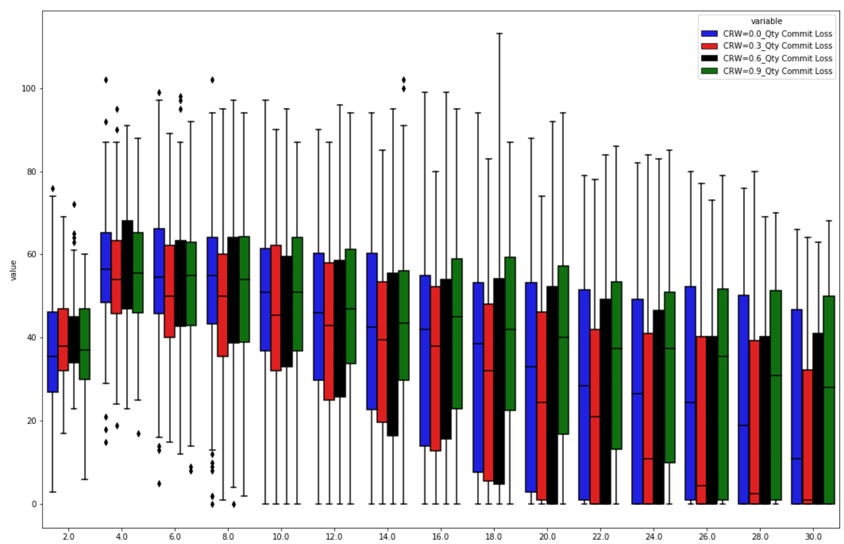
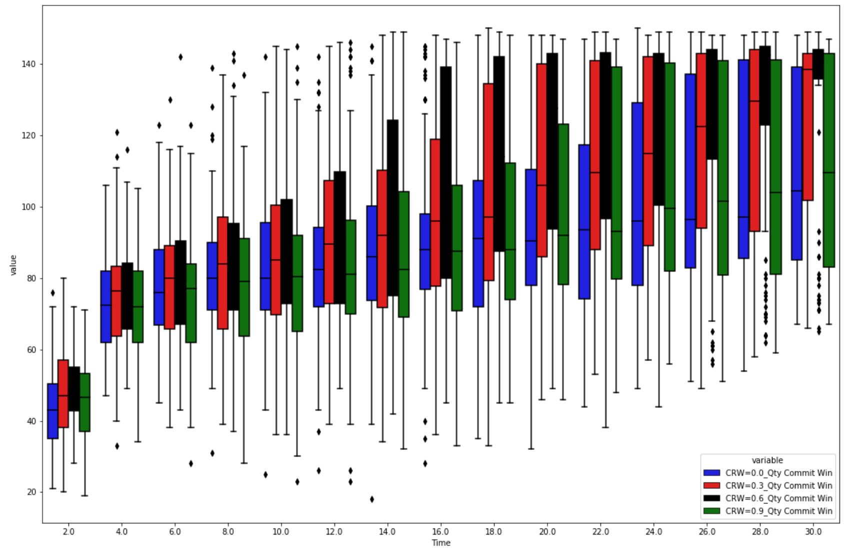


Figure 13: Diagrams are at $\nu = 5$, $distance = 0.76$, and $levy = 1.6$. Both diagrams shows amount of agents committed to the target option that won (top) and the other (bottom) with increasing CRW (0, 0.3, 0.6, 0.9).



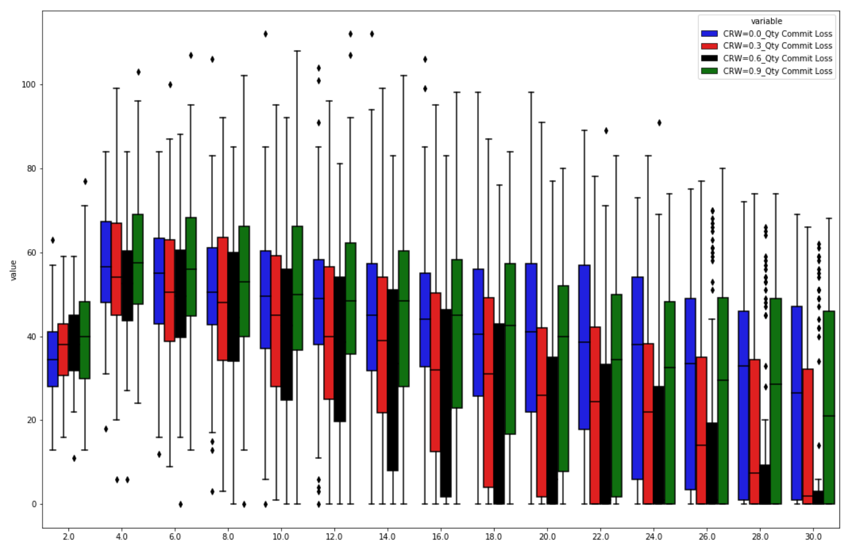


Figure 14: Diagrams are at $\nu = 5$, $distance = 0.76$, and $levy = 2$. Both diagrams shows amount of agents committed to the target option that won (top) and the other (bottom) with increasing CRW (0, 0.3, 0.6, 0.9).

The amount of runs converged depends on the quality value of the target options. For $\nu = 1.5$, there were no runs the successfully converged. Therefore to understand the behavior of distance to a lower quality value, another quality value of $\nu = 2$ was analyzed through running these additional experiments.

Figure 15 shows the percentage of convergence given a change in distance for $\nu = 2$ and $\nu = 5$. The results show that for $\nu = 5$, the change in distance does not change significantly the percentage of convergence. The distances that gave the highest percentage converged runs were at 0.26, 0.96, and 1.16. This therefore does not show a specific trend since both the lower and higher range of distances show an increase of converged runs. One the other hand for $\nu = 2$, as the distance increases the percentage decreases for the runs converged. The distances that gave the highest percentage converged runs were 0.28, 0.36, and 0.46 as high as 45%. This is a significant difference compared to $\nu = 1.5$ when there were 0% runs converged. Also, a higher quality value gives a much higher percentage of runs converged. On average, $\nu = 2$ has 30% converged compared to $\nu = 5$ that has 60%.

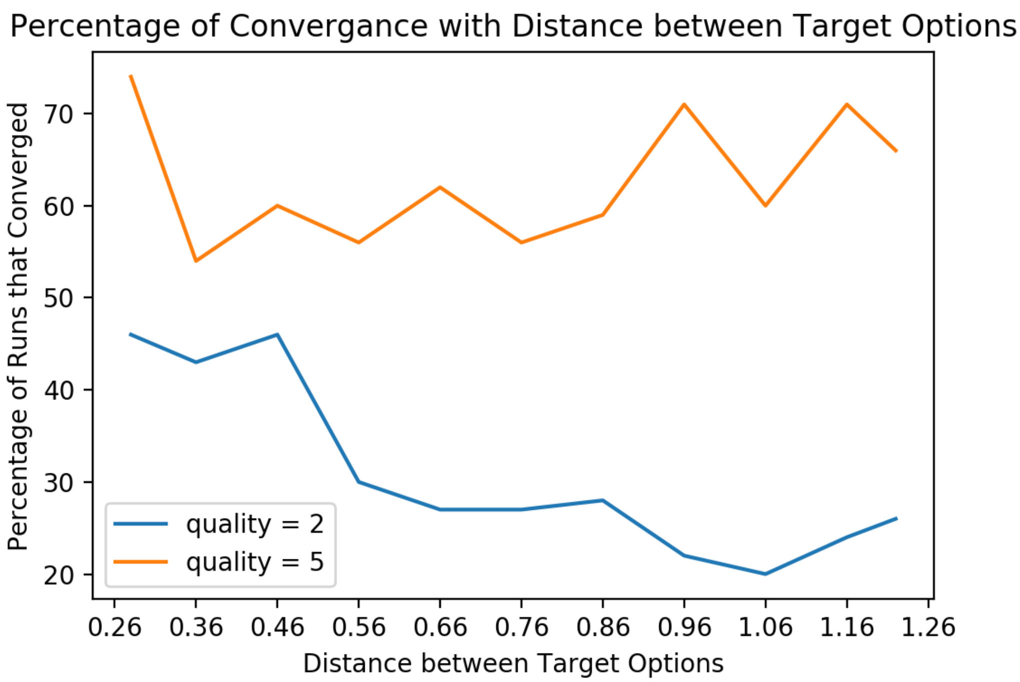


Figure 15: Diagram shows the percentage of runs that converged as it changes with distance between target options.

Figure 16 shows the average time of convergence given a change in distance for $\nu = 2$ and $\nu = 5$. The results show that for $\nu = 5$, the change in in distance increases the time of convergence. Between 0.28 to 0.46 the lowest convergence time is 13 minutes and then increases to 18 minutes at its peak when distance is 0.76 and otherwise constant at about 15. Similarly, for $\nu = 2$ the average time of convergence slightly increases. At 0.28 distance, the lowest convergence time is 20 minutes; while at 0.76 distance shows the highest convergence time at 25 minutes; everywhere else, the average time is about 23 minutes. This shows that the distance chosen by the original distance of 0.76 is the slowest convergence for either quality value. Also, a higher quality value gives a much lower average convergence time. On average, $\nu = 2$ has 23 minutes to converge compared to $\nu = 5$ that has 15 minutes.

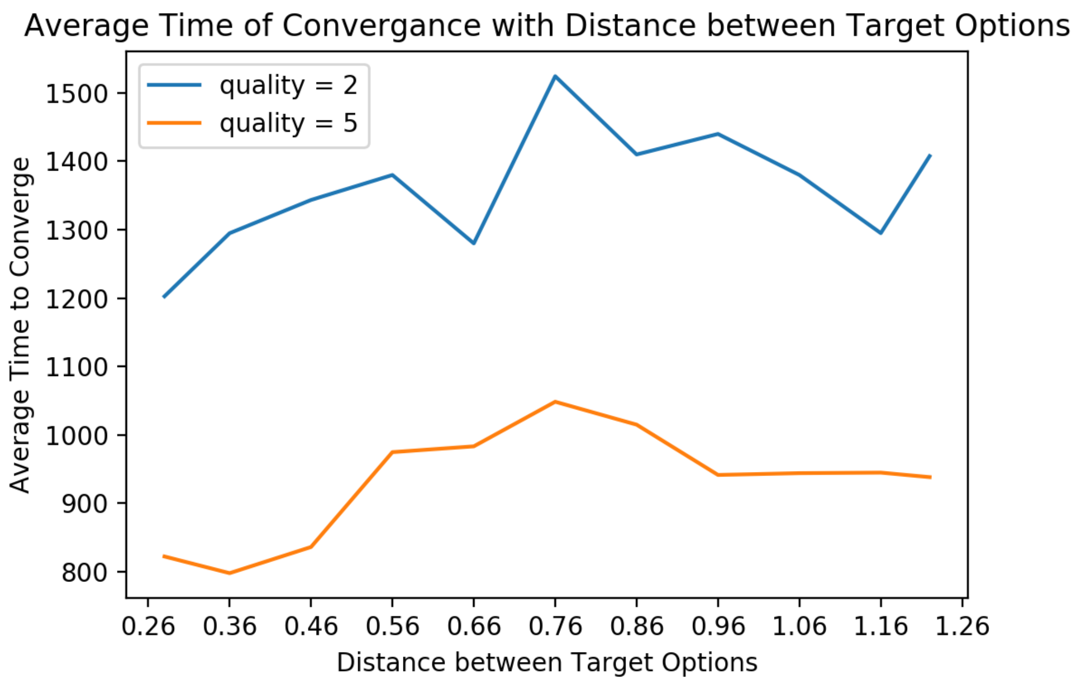


Figure 16: Diagram shows the average time of convergence as it changes with distance between target options.

5.3 Results from Experiments with CWR and Levy Exponent Parameters

The following subsections describe the results of the two set of experiments: CRW and Levy Exponent parameters using a constant distance of 0.76, as specified in the original paper experiments. CRW stands for Correlated Random Walk, which is commonly used for simulating non-oriented animal movement patterns. This agent's trajectory is represented by a sequence of distinct, independent randomly oriented 'moves' [3]. If $CRW = 0$, the rotation angle of the agent is a value chosen at random from a uniform distribution between $0 - 2\pi$. Otherwise for other values of CRW, the rotation angle of the agent is a value chosen from a wrapped Cauchy distribution.

Similarly, Levy walk (LW) is used to simulate movement patterns given the levy exponent parameter. In this LW models, displacements and turning angles draw from a power-law and a uniform distribution, respectively [1]. For LW, longer displacements are much more probable than Gaussian random walks. Recent works hypothesize that LW may represent an adaptive behavioral response to search in highly unpredictable and resource-poor environments, therefore increasing the likelihood of locating resources in space and time [1]. For the simulation, this random value is calculated using both exponential and uniform distributions given the Levy exponent parameter and the standard motion steps.

5.3.1 Results for CRW Exponent

The results shown in Figure 17 given the percentage of runs converged given a constant levy exponent. For all levy values, at $CRW = 0$ the percentages are very similar at about 56%. When $levy = 1.2$ and as CRW increases from 0 to 0.6, the percentage remains constant at 56%. However, when $CRW = 0.9$, there is a decrease to 46%. When $levy = 1.6$, at $CRW = 0.3$ the percentage increases to 75% but later decreases to 70% at $CRW = 0.6$ and 55% at $CRW = 0.9$. For $levy = 2$, it has a similar percentage at $CRW = 0.3$ however it increases to 88% at $CRW = 0.6$ and as expected decreases to 58% at $CRW = 0.9$.

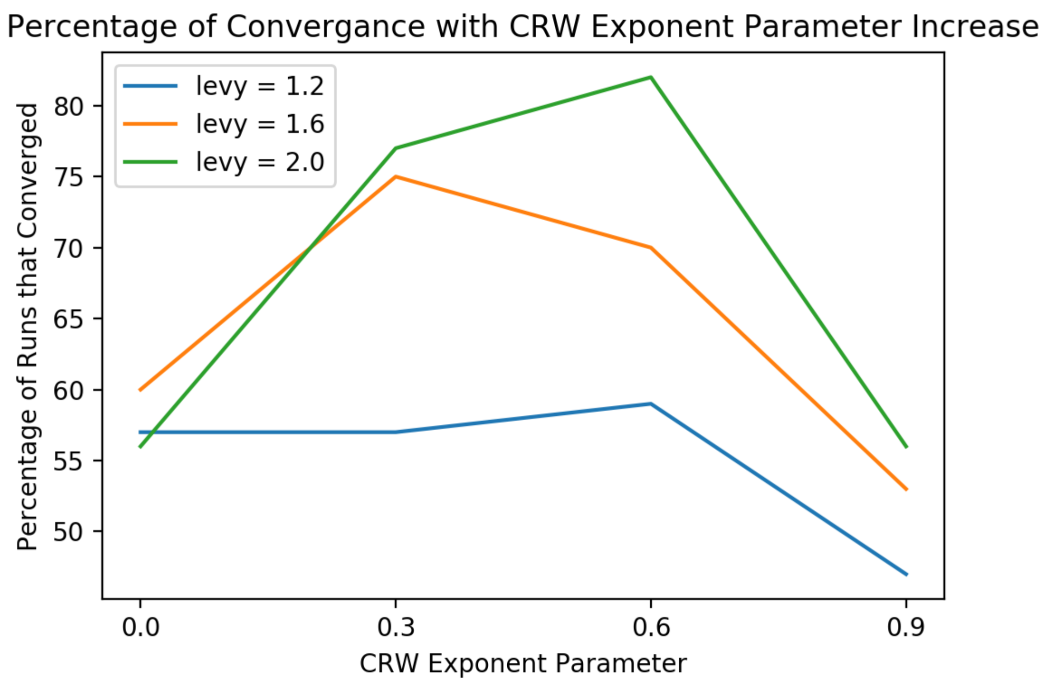


Figure 17: Diagram shows the percentage of runs that converged as it changes with CRW Exponent parameter.

The results of Figure 18 show the average time for convergence keeping levy exponent constant. Just as $CRW = 0.6$ and $levy = 2$ gives a better performance in percentage, it also does for average time which gives the lowest quantity: 15 minutes. Also, the worst performance is for $CRW = 0.9$ and $levy = 1.2$ with 18 minutes. The only quantity that provides a low average convergence time for $CRW = 0$ is when $levy = 1.2$. For $CRW = 0.3$, any levy amount provides the same time of convergence.

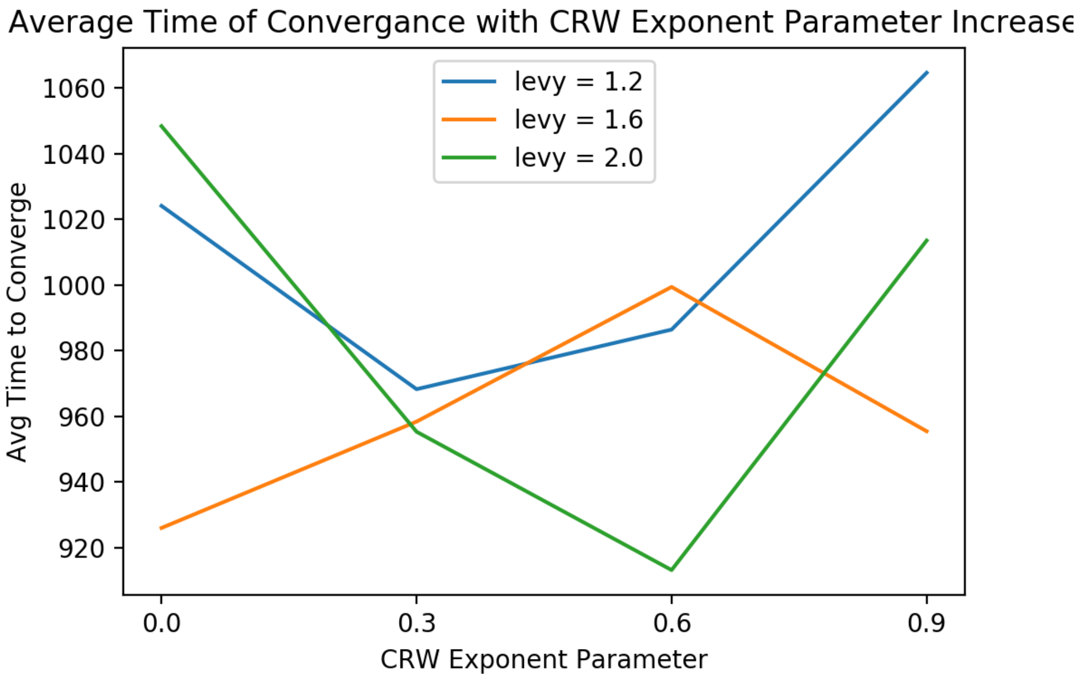


Figure 18: Diagram shows the average time of convergence as it changes with CRW Exponent parameter.

5.3.2 Results for Levy Exponent

Compared to results from CRW, the levy experiments for percentage converged shows a trend, as seen in Figure 19; as the levy exponent increases from 1.2 to 2, the percentage of convergence mostly increases. For $CRW = \{0.3, 0.6, 0.9\}$, this trend works. On the other hand for $CRW = 0$, the percentage maintain consistent and decreases slightly when $levy = 2$. The worst and best parameters are consistent with that seem previously seen when described above.

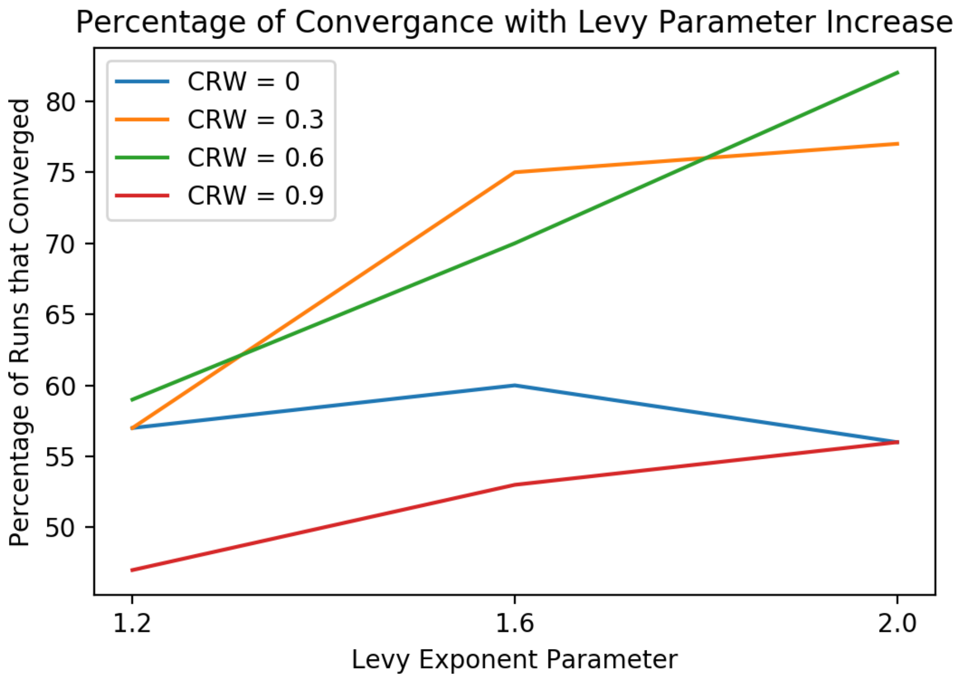


Figure 19: Diagram shows the percentage of runs that converged as it changes with Levy Exponent parameter.

Figure 20 shows no particular trend for all different CRW parameters, similar to the plot of CRW described above. Both $CRW = \{0, 0.9\}$ show a similar trend where at $levy = 1.2$ shows a high convergence time, then decreases at $levy = 1.6$ until it increases again at $levy = 2$. For $CRW = 0.3$, the average time of convergence stays mainly consistent. Similar to $CRW = 0.6$, except when $levy = 2$ which decreases and becomes the best performance.

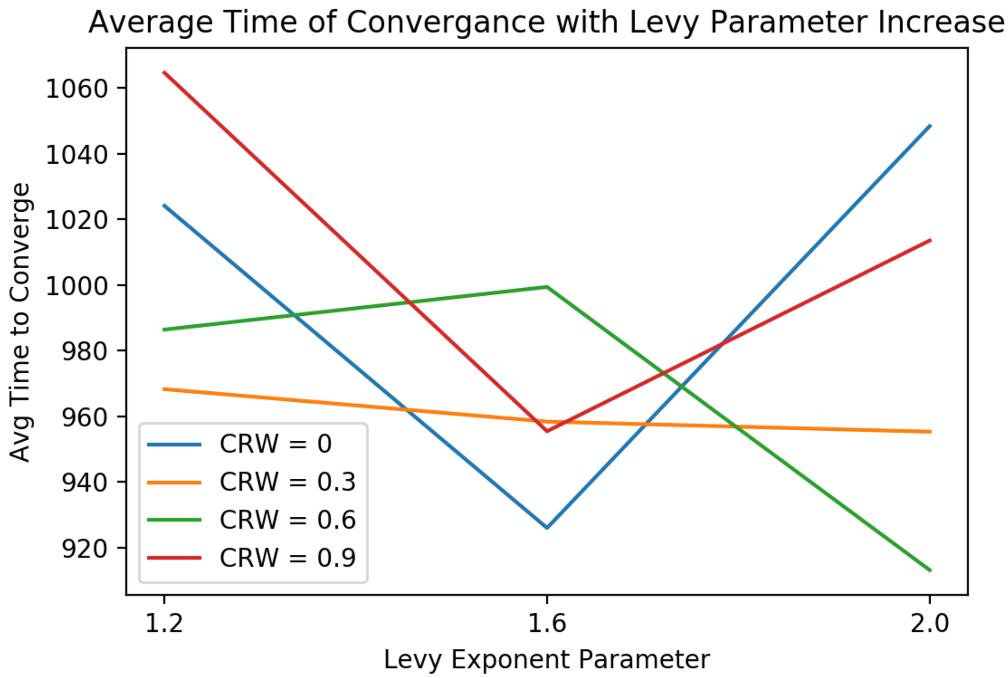


Figure 20: Diagram shows the average time of convergence as it changes with Levy Exponent parameter.

6 Discussion

The experiments ran to compare the agent commitment behavior in this simulation implementation and that of the original paper show that:

- General behaviors are as expected for both $\nu = 1.5$ and $\nu = 5$.
- As expected, results for $\nu = 1.5$ is that the amount of agents committed to both target options stabilize to a constant with time. Neither target option therefore also never reaches the quorum to converge.
- As expected, results for $\nu = 5$ is that the amounts of agents committed to the target options diverge. The target option that won increases while the target option that losses decreases. The target option that increases reaches the quorum to converge.

The experiments ran by the changing distances with constant $levy = 2$ and $CRW = 0$ show the following conclusions for both quality values $\nu = 2$ and $\nu = 5$:

- Smaller distances increase the percentage of convergence.

- Smaller distances decrease the average time for convergence.
- Distance of 0.76, as used in the original paper, shows the slowest (worst) convergence time of 25 minutes when $\nu = 2$ and 18 minutes when $\nu = 5$.
- Percent of Runs Converged: best performance when $\nu = 5$ and any of the following $distance = \{0.28, 0.96, 1.16\}$; worst performance when $\nu = 2$ and $distance = 1.06$.
- Average Convergence Time: best performance when $\nu = 5$ and any of the following $distance = \{0.28, 0.36, 0.46\}$; worst performance when $\nu = 2$ and $distance = 0.76$, as stated in the item above.

The experiments ran by the CRW and Levy exponents with constant target options' $distance = 0.76$ show the following conclusions for quality value $\nu = 5$:

- Increasing levy exponent from 1.2 to 2, mostly increases the percentage of convergence
- All other trends do not have a clear definition.
- Percent of Runs Converged: best performance when $CRW = 0.6$ and $levy = 2$ at 88%; worst performance when $CRW = 0.9$ and $levy = 1.2$ at 46%.
- Average Convergence Time: best performance when $CRW = 0.6$ and $levy = 2$ at 15 minutes; worst performance when $CRW = 0$ and $levy = 2$ or almost equally when $CRW = 0.9$ and $levy = 1.2$ at 18 minutes.

7 Conclusion

Analyzing all the results of the simulation experiments show that the best performance for both percentage of converged runs and average time to converge are using $\nu = 5$ and as follows:

- $distance = 0.28$
- $CRW = 0.6$ and $levy = 2$

For avoiding the worst performance, do not use $\nu = \{1.5, 2\}$ and the following parameters:

- $distance = 1.06$ for percentage of runs converged
- $distance = 0.76$ for average convergence time
- $CRW = 0.9$ and $levy = 1.2$

The lower the quality value, the worse the performance in both percentage of converged runs and average time to converge.

8 Bibliography

1. Bartumeus, Frederic. (2009). Behavioral intermittence, LEvy patterns, and randomness in animal movement. Oikos. 118. 488 - 494. 10.1111/j.1600-0706.2009.17313.x.

2. Reina A., Bose T., Trianni V., Marshall J.A.R. (2018) Effects of Spatiality on Value-Sensitive Decisions Made by Robot Swarms. In: Groß R. et al. (eds) Distributed Autonomous Robotic Systems. Springer Proceedings in Advanced Robotics, vol 6. Springer, Cham

3. Reynolds, Andy. (2010). Bridging the gulf between correlated random walks and Lévy walks: Autocorrelation as a source of Lévy walk movement patterns. Journal of the Royal Society, Interface / the Royal Society. 7. 1753-8. 10.1098/rsif.2010.0292.

9 Appendix

The following diagrams show the plots that compare the general behavior of the amount of committed agents for each target option, while changing the CRW and Levy exponents.

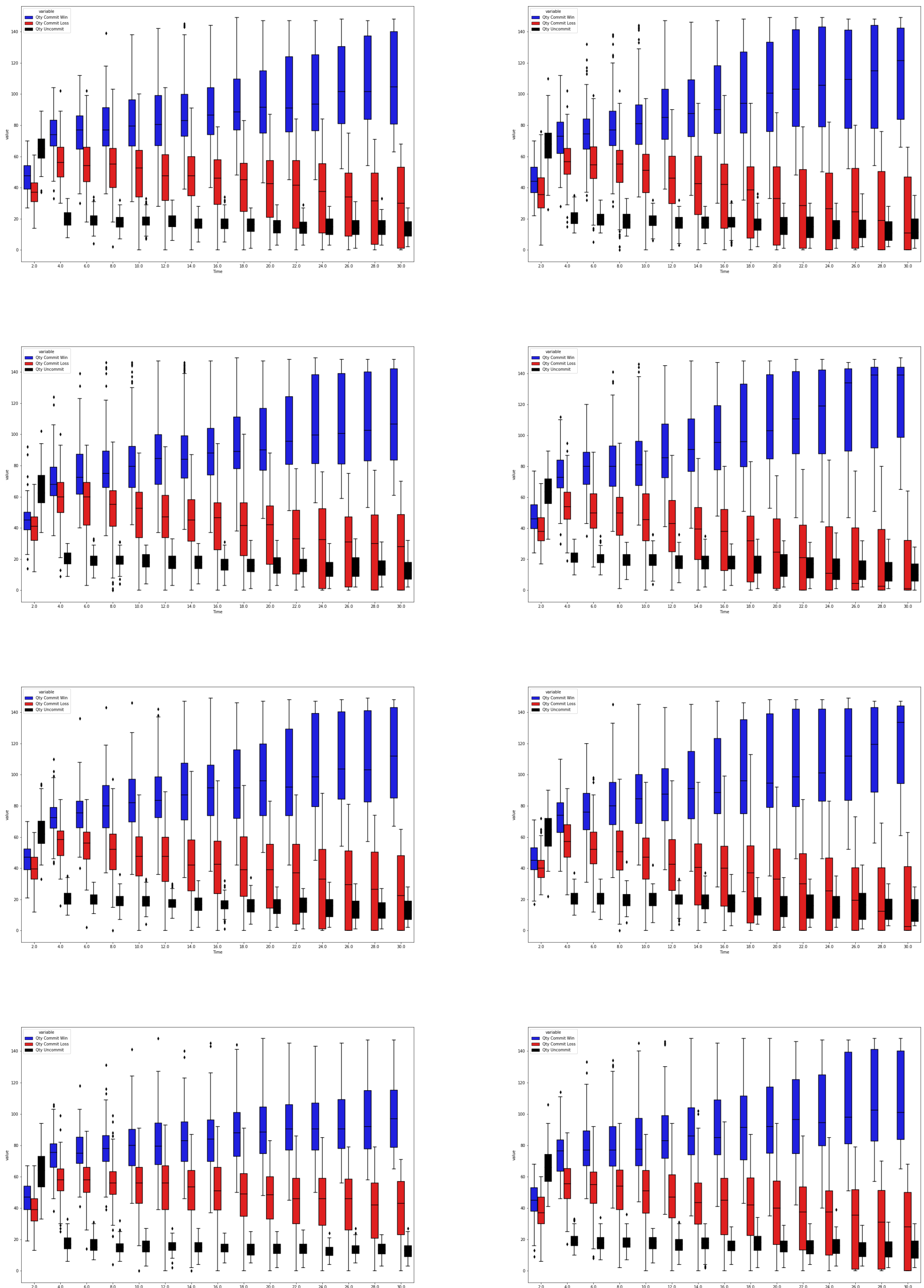


Figure 21: Diagrams for all is at $\nu = 5$; on the left corresponds to results at $levy = 1.2$ and on the right at $levy = 1.6$. Each diagram shows amount of agents committed to the target option that won (blue) and the other (red). Vertically down corresponds to changing the CRW exponent from top to bottom: 0.0, 0.3, 0.6, and 0.9.

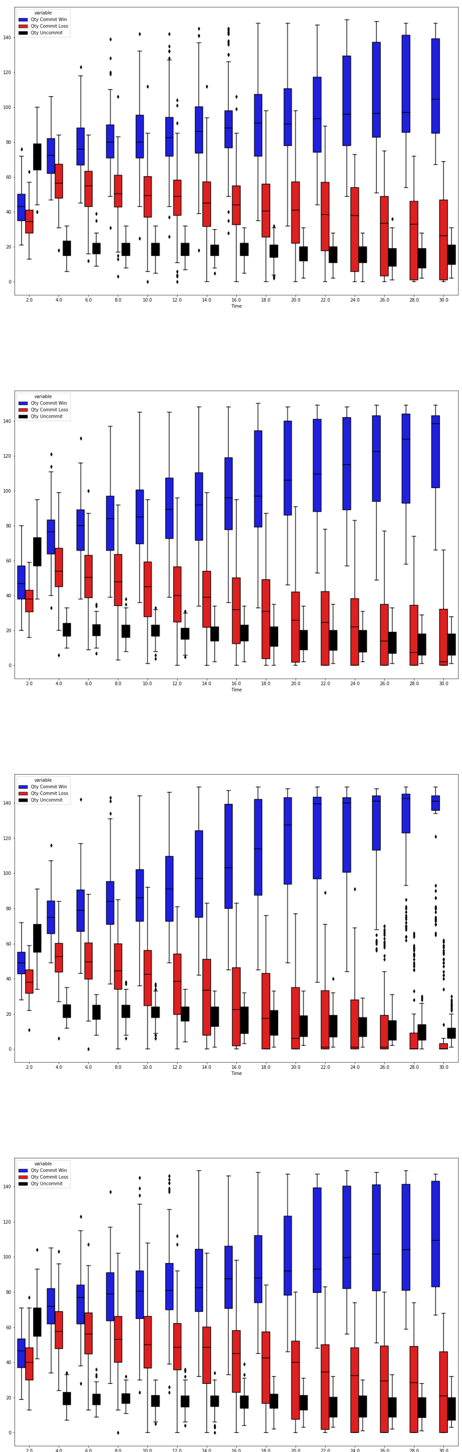


Figure 22: Diagrams for all is at $\nu = 5$ and corresponds to results at $levy = 2.0$. Each diagram shows amount of agents committed to the target option that won (blue) and the other (red). Vertically down corresponds to changing the CRW exponent from top to bottom: 0.0, 0.3, 0.6, and 0.9.