Agents Survivability in Logistics Problem

Daniel Correia
IST MEIC-A, 80967 (Group 17)
daniel.b.correia@tecnico.ulisboa.pt

Pedro Miguel Orvalho
IST MEIC-A, 81151 (Group 17)
pedro.orvalho@tecnico.ulisboa.pt

ABSTRACT

This report provides a study of agent survivability in the logistics decision-making problem using a multi-agent system operating in an hostile environment. We modelled three types of agents: companies, trucks and clients. Trucks perform transportation routes that may pass by multiple clients. Companies are responsible for participating in Clients' auctions of offers and processing the offers they win. Our experimental results show that the companies' throughput is mostly influence by its ability to process offers and by its location in the network. The level of hostility of the environment can be a key factor in forcing weak companies to die out faster, allowing companies in favorable positions to monopolize the system over time.

Keywords

Multi-Agents, Survivability, Hostile Environment, Logistics, Travelling Salesman, Auction

1. INTRODUCTION

This project was developed in the context of the Autonomous and Multi-Agents Systems course at Instituto Superior Técnico. The main goal was the development of an autonomous multi-agent system inserted in a logistics problem domain (transportation of goods) to study the evolution of agent survivability in an hostile environment.

The framework developed during this project can be extended to be applied in real-life problems that require risk analysis in a volatile network environment. For example, we can study which type of parameters have the most impact in agent survivability: network structure, agent characteristics or environment hostility.

The problem of decision making in logistics would be too complex if we tried to study all decision levels [1], thus we decided to focus on the execution level (e.g. movement of the trucks) and operational (e.g. distribution of offers to a company's trucks).

For the purpose of this project we will focus on the following questions:

1. Which parameters have the largest impact on agent survivability?
2. How relevant is the network structure relatively to agents characteristics?
3. Is there an optimal configuration that maximizes survivability?

2. PROBLEM

The main goal of this project is the study of agent survivability in a volatile network environment using a logistics problem domain. Thus, we modeled our problem by considering which real-life characteristics would be most interesting to adapt into our multi-agent system's features.

Environment

We begin by modelling the environment where our agents will operate. Consider a network environment represented by a weighted undirected graph where each node corresponds to either a company or a client and each edge has a weight that represents the cost of using the edge for transportation routes.

On top of this network, we applied several random hostile environment events that either change the structure of the network (edge explosion), impact the agent's survivability directly (existence tax) or remove an agent from the system (truck explosion).

Formally, we can define our environment using the following properties:

- Accessible - all agents can obtain complete and up-to-date information about the network.
- Deterministic - each action taken by an agent has a single guaranteed effect (there is no execution error factor).
- Static - any random events that may change the environment are applied before the agents deliberate and act.
- Discrete - there is a finite set of possible actions and perceptions in the system (e.g. trucks are only able to take actions related to their autonomous movement in the network).
- Episodic - each simulation ran in our system is independent of each other.

Agents

In our multi-agent system we considered three types of agents (companies, trucks and clients) representing real-life entities in a logistics problem domain.

The most important agent type in our system is the company since it contains the property we want to study - survivability. Companies are responsible for several behaviors in the system:

1. Distribute offers by the trucks they own

2. Send trucks into the network when they are ready to start a transportation route

3. Bid on auctioned offers by clients

4. Handle events such as receiving profit from a truck that completes a route or processing a truck explosion

The "Company" agent type is characterized primarily by its current money (representing survivability), position in the network and set of owned trucks. It is also further parametrized by properties that influence its behaviors:

- Initial Money

- Truck Threshold: decide when a truck is full enough to leave

- Profit Margin: to compute a profitable bid on an auctioned offer

- Tax: to be paid for each offer the company wins on an auction

The second agent type we modelled in our system was the truck which represents a truck owned by a company. A "Truck" agent is assigned a set of offers by its owner (a company) and is responsible for delivering its offers to the respective clients by calculating the cheapest route that passes by all the clients.

Each truck is allowed to move along only one edge per time-step of the system and is characterized mainly by its capacity (maximum and currently used) and its current status (free for usage or occupied/"en route").

Finally, we defined the "Client" agent type representing an entity in the system that wants to deliver a certain amount of units between two points in the network. The client does so by choosing the amount and the target position of the offer and then using a auction with the companies to find the cheapest offer based on the companies' bids.

3. SOLUTION

For our project, we decided to build a Python implementation where each agent is represented by a class containing the respective characteristics and behaviors. This allowed us to then build a simulator over a network structure using these agents to do experimental simulations and evaluate agent survivability in multiple scenarios.

Network Representation

For the sake of simplicity, we decided to use the NetworkX [2] library to streamline the process of setting up a network structure environment where our agents will act. This had the additional benefit of supporting trivial parametrization of the network structure.

Thus, our developed solution supports parametrization of the number of nodes, weight range of the edges and network type (e.g. random or scale-free).

Truck - Autonomous Movement

Regarding the truck movement in the network, we decided to model its behaviors by taking inspiration from the traveling salesman problem and applying the key principles to the truck agent.

A truck agent can contain multiple offers to be delivered to specific locations in the network (e.g clients) and based on its current set of offers, it tries to build the route that passes by all the target locations and where the total cost is minimal.

In the context of the simulations, this means that in each time-step, if a truck is in an "Occupied" state (currently delivering offers), then it builds an approximation of the cheapest route and moves to the first node of the calculated route.

Finally, to avoid making the project too complex, once a truck has finished all its deliveries successfully, it teleports back to its company owner and gives the profits obtained on the deliveries route to the company (instead of planning a return route to its company owner).

Company - Offers Distribution

Regarding the companies offers distribution, given a set of offers that were obtained after the clients auctions (further explained in the next section), a company agent goes over its owned trucks that are free for usage and distributes the offers based on a greedy cost criteria.

In practice, this cost criteria consists of calculating the updated route cost of a truck if it is assigned a certain offer and selecting the truck with the minimum cost.

If there is no truck that is able to receive a new offer (e.g. not enough capacity left), then the offer is kept until its able to assigned to a truck or the end of the simulation.

Finally, besides being responsible for distributing offers to its trucks, each company is able to control when a truck is full enough to start delivering offers to clients using a truck threshold parameter, that represents the minimum used capacity necessary for a truck to be marked as "Occupied" and to start moving along a delivery route in the next time-step.

Client - Offers Auctions

A client agent is responsible for generating offers and managing an auction where each company bids on their price for delivering an offer. Considering the auction mechanisms used in our project, the auction is characterized as first-price, sealed bid and single-shot.

An offer is composed of three elements: the quantity the client wants, the location of the delivery (the client itself) and the total price (to be determined by the auction).

In each time-step of simulation, each client decides if it will generate a new offer based on a random risk factor initialized once when the simulation is started. Assuming a new offer is generated, the offer enters the auction process and each company bids on it.

A company bid is obtained by first considering the quantity desired in the offer, the company's cost for each unit of product and the company's profit margin using the following formulas:

$$baseCost = (quantity * unitCost) + bestRouteCost$$

$$baseBid = baseCost * profitMargin$$

where profitMargin > 1 and minRouteCost is the minimum cost for a company to distribute the new offer to one of its trucks.

After calculating this base bid, a tax is applied and, thus the final bid is given by:

$$bidTax = baseBid * tax$$

$$finalBid = baseBid + bidTax$$

To conclude the auction process, the client that started the auction takes all the bids from all the companies and multiplies each bid by the client's preference for each company. This preference can represent several real-life characteristics in this domain like company reputation, client habits and trust.

The minimum bid is selected from this updated set of bids and assigned to the winning company. The company that wins the offer pays immediately the associated bidTax for the offer.

Environment - Hostile Events

To increase the hostility of our environment and evaluate the evolution of survivability in such an environment, we decided to implement three types of events, each modifying different aspects of the system.

Existence Tax

The existence tax is an event that impacts the agent's survivability directly by reducing the company's current money by a constant value using the following expressions:

$$MoneyTax = Comp_i.InitialMoney * ExistenceTax$$

$$Comp_i.Money = Comp_i.Money - MoneyTax$$

This event is executed, in each time-step, before the company agents act on the environment.

Edge Explosion

Edge explosion is an event that changes the structure of the network by randomly removing an edge from the network representation. This event allows us to simulate edge failures in the network, for example, a road accident, blocking access to a path.

A low value for this probability parameter is recommended to avoid making the network too disconnected, which would result in anomalous behaviors in the system.

Truck Explosion

Truck explosion is an event that randomly removes an agent from the system, specifically, a truck agent. This event represents spontaneous agent failure (e.g. a truck that stops working due to mechanical issues).

As was noted for edge explosion, it is recommended to choose low values for the parameter that controls this event's probability. Otherwise, the system will eventually reach an anemic state where company agents are unable to survive due to the lack of trucks.

4. RESULTS

For the purpose of the simulations executed to evaluate the evolution of agent survivability given different configurations of the available parameters in our system, we decided to only change one parameter in every attempt and fix the remaining parameters.

We decided to use each company's money at the end of the simulation to evaluate agent survivability, since this is the key characteristic that is changed during the multi-agent system's execution.

4.1 Default Parameters

To simplify our experimental analysis, we used the following default parameters for each parameter type in our system.

Network Parameters

Number of Nodes	15
Graph Type	Random
Graph Parameter	20
Edge Min Weight	1
Edge Max Weight	10

Agents Parameters

Number of Companies	5
Number of Trucks	7

Company Parameters

Initial Money	2500
Truck Threshold	100
Unit Cost	1
Profit Margin	1.5
Tax	0.05

Client Parameters

Risk	] 0, 1 [
Preferences	0.2 (all companies)
Offer Min Quantity	25
Offer Max Quantity	80

Events Parameters

Existence Tax	0.05
Edge Explosion	0.01
Truck Explosion	0.01

4.2 Money vs Time

The first configuration used was the default configuration to evaluate the evolution of survivability (or money) over time.

This analysis allowed us to observe the standard patterns of survivability at each time-step of the simulations.

The results presented in Figure 1 were obtained by running 30 iterations of the simulation with 100 time-steps and each data point is the average of all iterations.

By analyzing Figure 1 and the respective graph (Appendix A.1), we can extract the following conclusions:

- On average, only half of the agents in our system survive at the end of the simulation.
- Companies that survive are usually in very favorable positions in the graph (e.g. high degree, high closeness centrality, low weight neighboring edges).

Finally, the negative spike of Company C's money observed in Figure 1, can be a result of the state of the system at time-step t=20. One possibility is that at t=20, only Company C had trucks available to bid on offers, which means it won all the offers, resulting in a large sum of bid taxes to be subtracted from Company C's money.

4.3 Random vs Scale-free

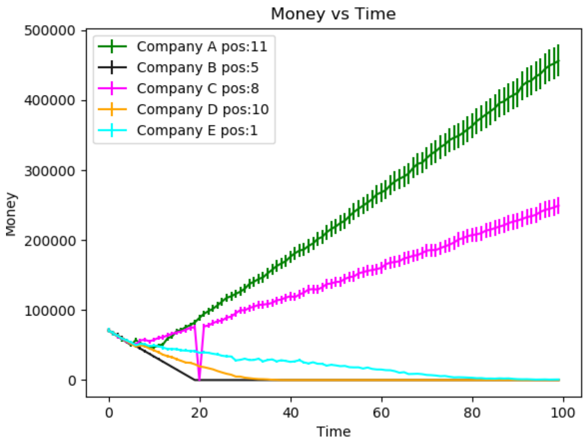

Figure 1: Companies money evolution over time for graph 1 (Appendix A)

The first variation we used was the graph type selected to generate our system's environment. Thus, by relying on NetworkX capabilities, we decided to compare the performance of the winning agent if the graph is random or scale-free.

From network theory concepts, our expectation is that since scale-free networks naturally contain "hubs" [3], the cost and delay of the truck's delivery routes should be lower w.r.t the respective random graph counterparts.

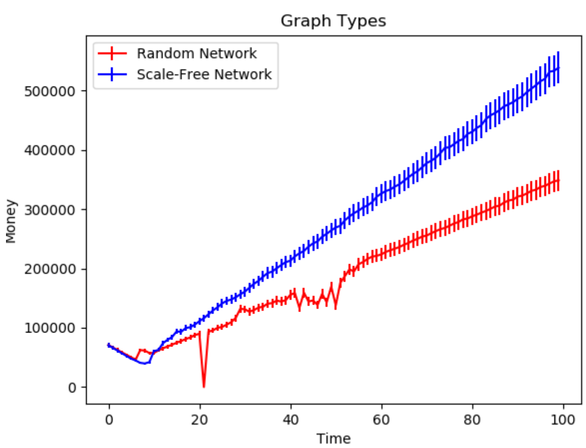

Figure 2: Winning company money evolution on random and scale-free graphs

From the results in Figure 2, we were able to confirm our expectations for this parametrization choice and obtain the following conclusions:

- Scale-free network structures provide cheaper routes due to the existence of hubs that work as shortcuts to move from one node to another.
- Although the winning agent performs better on a scale-free network environment, the difference to random networks is small since the money evolution of both configurations seems to be linear over time.

4.4 Number of nodes

A key network parameter we can change is the number of nodes in the underlying graph structure. In theory, the impact on survivability should be minimal if every other parameter is fixed, since even though the amount of clients increases with the number of nodes, the cost of the routes also increases, resulting in an expected stable survivability rate.

In Figure 3 we present the results for a simulation where we increased the number of nodes. We also changed the number of trucks (8 and 16) to validate an expected increase in profit. The results presented are w.r.t the company that had the most money at the end of each simulation.

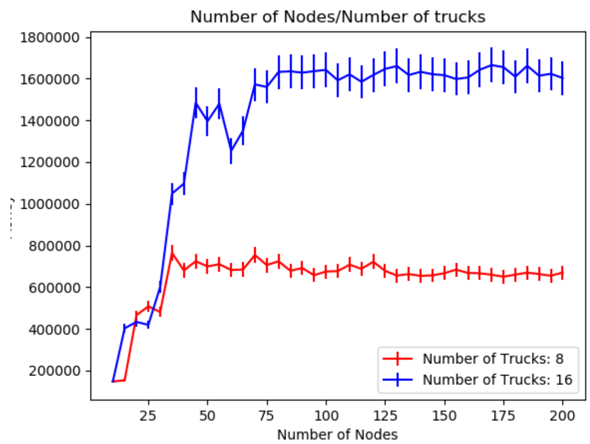

Figure 3: Winning company money evolution with varying number of nodes with 8 and 16 trucks

Analyzing Figure 3, we can extract the following conclusions:

- Increasing the number of nodes for fixed parameters does not result in significant changes to the survivability of the winning agent. In fact, the final money of the winning company eventually converges to a stable state, as we increase the number of nodes.
- The number of trucks is a very likely bottleneck to the maximum profit of a company, since the higher the number of nodes, the longer it takes a for a truck to finish a delivery route. This implies there is higher delay between the dispatching of a truck and its return to owner company for higher number of nodes.

4.5 Number of Companies

Regarding the "Agents Parameters" group, we decided to experiment with increases of the number of company agents for a fixed size graph. Our expectation is that by increasing the number of companies, there is higher degree of competition in our system and thus, the winning company final money will converge to lower values.

By observing the results in Figure 4, we can conclude that:

- For high number of companies, the environment becomes saturated with company agents and the winning agent's money decreases. This characteristic is similar to the "tragedy of the commons"[4] associated with overexploitation contexts.

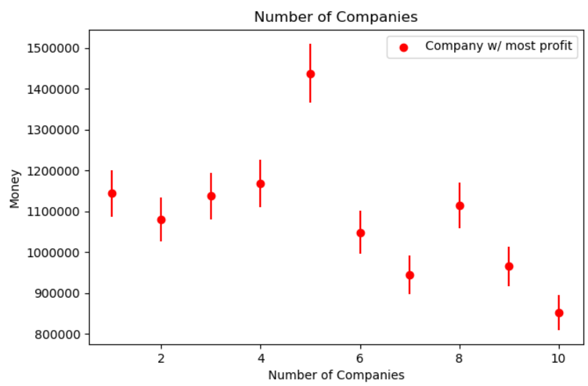

Figure 4: Winning company money evolution for increasing number of companies

- There appears to be an optimal configuration for the generated graph, where there is just enough competition for the offers that in the end, the winning agent has the highest money. This happens for n=5.

4.6 Truck Threshold

From the Company agent parameters set, we decided to change the truck threshold to evaluate if the frequency of truck dispatching has a significant impact on the agent's survivability (or company money).

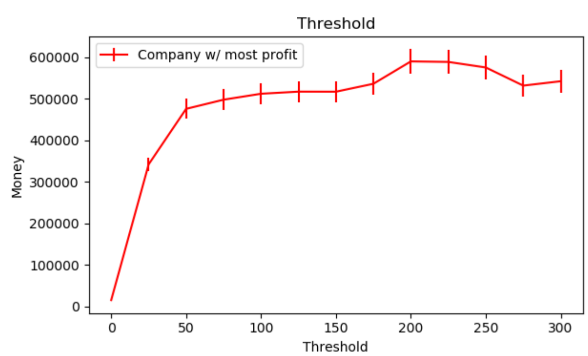

Figure 5: Winning company money evolution for increasing truck threshold values

From the results in Figure 5, we can extract the following conclusions:

- Increasing the threshold results in a higher profit for the winning agent since, for low threshold values, the company waits a long time for the truck to be sufficiently full with offers before dispatching, which results in a higher wait time for the company to receive delivery routes profits.

- Interestingly, for threshold values close to the maximum of 300 (which means the company dispatches trucks as soon as they have some offer), the winning agent's profit remains stable without any significant changes.

From this we can conclude that, for small finite size random graphs, there seems to be no significant strategic advantage between dispatching as soon as possible (maximum threshold) and dispatching only after a threshold is satisfied.

4.7 Truck Explosion

Consider now the "truck explosion" event present in our system's environment. We can evaluate the impact of this event on agent survivability by increasing this event's probability of occurrence in each time-step of the simulations.

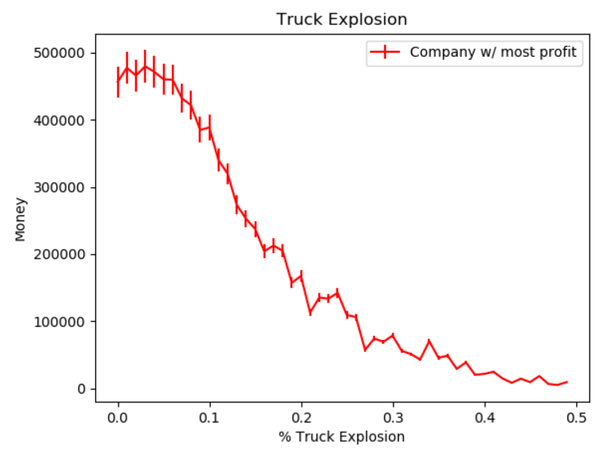

Figure 6: Winning company money evolution for increasing truck explosion probability

Based on Figure 6, we can observe that increasing the truck explosion event probability results in a direct negative impact on the winning agent's survivability, until eventually the agent becomes unable to survive.

4.8 Edge Explosion

The other event we parametrized was the edge explosion event probability. Increasing this parameter results in a decrease of network connectivity, which in turn affects the agents capability of defining and completing delivery routes in the system's environment.

From Figure 7 and comparing with the results obtained for truck explosion (Figure 6), we can conclude that edge explosion events require higher probabilities to have a significant effect on agent survivability.

This is mostly supported by the initial configuration of underlying graph structure and agents modelling decisions. While removing trucks from a company agent reduces their ability to complete offers by a constant factor, removing edges is a slower process because it takes several random edge removals for a random graph to become completely disconnected.

4.9 Profit Margin

The final "Company Parameter" we decided to experiment with was Profit Margin. This parameter is responsible for controlling how good an offer's profit needs to be for a company to "want" to complete it.

Since profit margin affects the value of a company's bid, the expectation is that higher profit margin values will lead

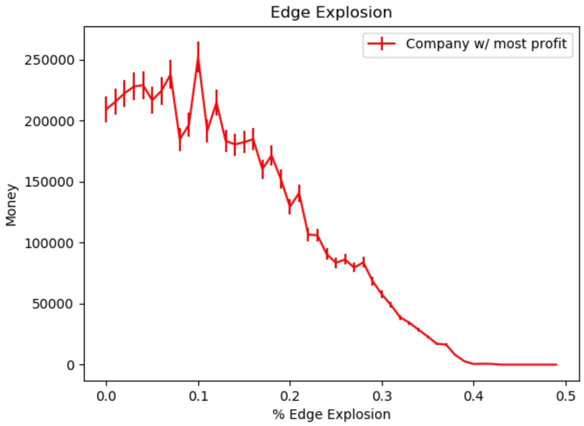

Figure 7: Winning company money evolution for increasing edge explosion probability

to a smaller set of offers completed on average, which may have a positive or negative effect on agent survivability depending on the remaining parameters of the system (environment, agents and network).

To evaluate this parameter we decided to run an initial simulation where each company had the same profit margin value of 1.5. Then we selected the company with the worst profit (e.g. a company that failed to survive) and reran simulations where we increased the profit margin of that company.

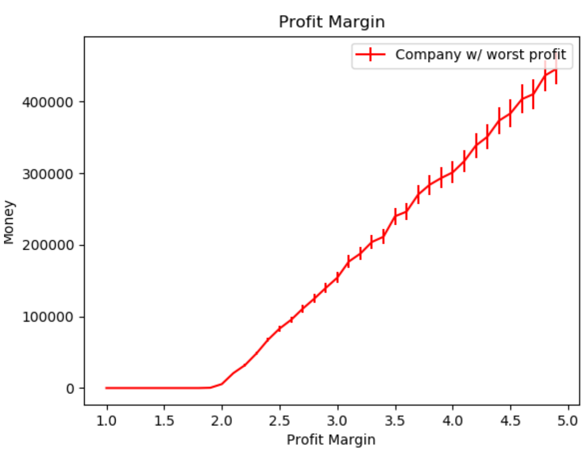

Figure 8: Example 1 of worst company money evolution for increasing profit margin (positive effect)

Figure 8 shows a situation where the increase of the profit margin had a positive effect on agent survivability. The results indicate that this company was unable to survive for low value of profit margin because the offers it completed weren't profitable enough to counterbalance the consistent applications of existence tax at each time-step.

By increasing the profit margin, even though the company completed less offers in total (since its bids became higher), each offer provided more profit for the company, resulting in a higher survivability.

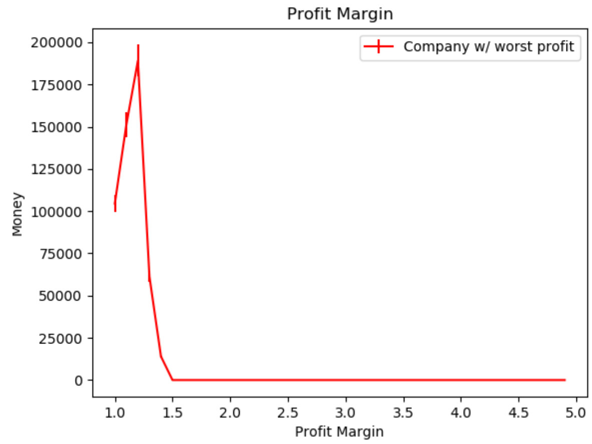

Figure 9: Example 2 of worst company money evolution for increasing profit margin (negative effect)

The effects of increasing profit margin depend on too many parameters of the system to be predicted consistently. In Figure 9, we present a repetition of the same methodology on a different random graph and the results are the opposite - increasing profit margin eventually led to the death of the agent we were trying to improve.

If we observe the results of Figure 9, we can note that initially the increase in profit margin had a positive effect and actually increased the company's money. However, greedily going past a certain limit seems to break this specific agent's equilibrium between its ability to process offers for income and the existence taxes from the environment.

4.10 Client Preferences

To conclude our experimental analysis, we decided to try change of the Client parameters, in particular, the preferences parameter.

Preferences represent how a client perceives the reputation of each company and influences the decision of which bid the client should select at the end of the auction process. Without this parameter, the decision would be trivial and disconnected from the agents in the environment - select the minimum bid.

For example, if a company is assigned a high preference value its bids are more likely to be selected, even their "pure bid value" isn't the minimum.

To evaluate the impact of this parameter, we used the same methodology as in previous section ("Profit Margin") and tried to improve a company that performed worse in an initial simulation.

The results in Figure 10 are aligned with our expectations for this parameter: increasingly higher preference values lead to higher survivability rates.

We can also note that changing this parameter has an impact on survivability much sooner than increasing the profit margin.

5. CONCLUSIONS

During our experimental simulations, we tried to evaluate the impact of a subset of our system's parameters w.r.t the target parameter of the study - agent survivability (repre-

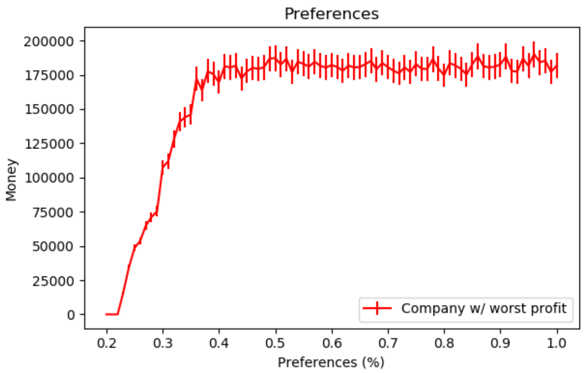

Figure 10: Example of worst company money evolution for increasing client preference

sented by the company's money).

Our results show that the parameters that have the largest impact on agent survivability are of two types:

1. Influence the ability of the agent to process requests quickly: number of trucks, truck threshold, favorable network location and structure.

2. Regulate the hostility of the environment or the competition for requests: truck explosion, edge explosion, number of companies.

Secondly, from the obtained results, it is not clear whether network structure is the dominant parameter group comparatively to agents parameters. For example, while usages of scale-free appear to result in better agent survivability performances, careful management of the profit margin and client preferences parameters seems to have largest and fastest impacts on agent survivability.

Finally, due to the wide range of parameters combinations, we were unable to find an generic optimal configuration that promotes agent survivability.

5.1 Future Work

There are several extensions and improvements that can be done on the work developed for this project, both to introduce more features to the multi-agent system and to extend the experimental analysis with new results:

- Test simulations for pairs/triplets of parameter variations to evaluate interdependence and correlation between parameters.

- Update the client's preferences over time using learning strategies (e.g reinforcement learning) or public reputation (e.g indirect reciprocity).

- Introduce the possibility for companies to negotiate with other companies, for example to trade offers or to share the profits of an offer that is too expensive to be done a single company.

- Consider adding the concept of velocity and return trips to the truck's movement on the network.

5.2 Related Work

In this project we tried to study agent survivability in a multi-agent system within an hostile environment, using logistics as a domain problem. We evaluated agent survivability by experimenting either with network structure, agents individual parameters or the hostility environment.

However, this is a very small subset of the whole problem of agent survivability. Other approaches for survivability have been studied by either introducing higher-level cooperation mechanisms between the agents or making the agent's behaviors more adaptive to changes in the environment [5] [6].

REFERENCES

[1] N.A. Lang, J.M. Moonen, F Srour, and Rob Zuidwijk. Multi agent systems in logistics: A literature and state-of-the-art review, 01 2008.

[2] NetworkX: Python software for complex networks. Networkx. `https://networkx.github.io/`.

[3] Scale-free network — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Scale-free_network#Characteristics`.

[4] Tragedy of the commons — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Tragedy_of_the_commons`.

[5] Egis Vincent and R'egis Vincent Bryan Horling. Survivability simulator for multi-agent adaptive coordination. In *In Proceedings of the First International Conference on Web-Based Modeling and Simulation*, 1998.

[6] N Gnanasambandam, Seokcheon Lee, Soundar Kumara, Natarajan Gautam, Wilbur Peng, V Manikonda, M Brinn, and Mark Greaves. Survivability of a distributed multi-agent application - a performance control perspective, 09 2005.

APPENDIX

A. GRAPHS

A.1 Money vs Time

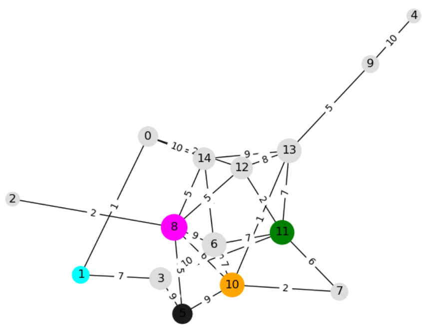

Figure 11: Graph 1 from Money-Time Simulations

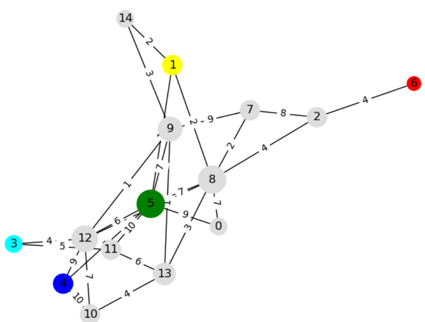

Figure 12: Graph 2 from Money-Time Simulations

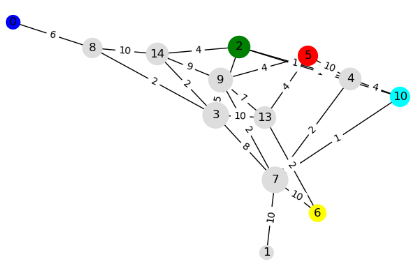

Figure 13: Graph 3 from Money-Time Simulations