Second Cycle Degree in
Automation Engineering

Reputation management in open robot swarms through crypto-economy

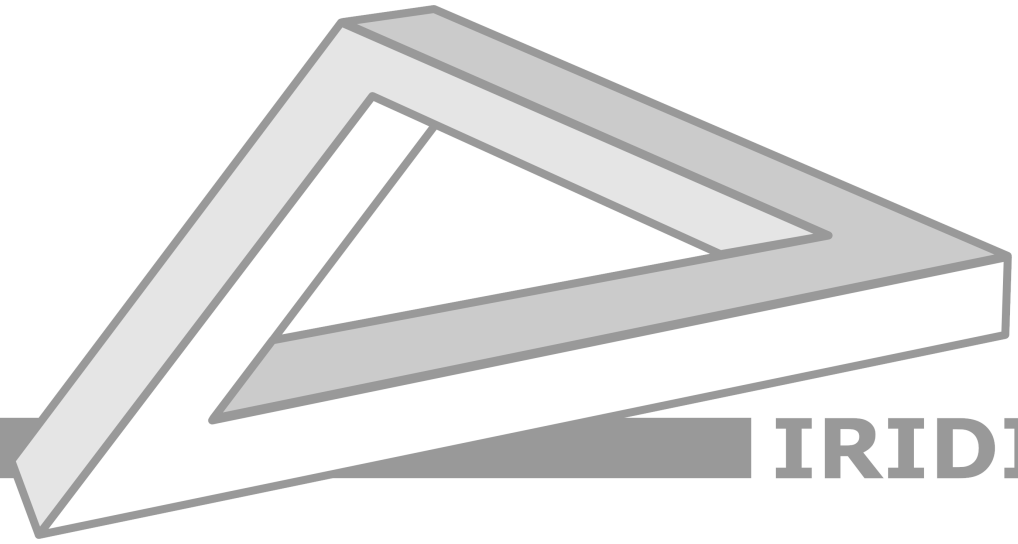
Supervisor:
Prof. Michela Milano

Author:
Francesco Cerri

Co-supervisors:
Prof. Marco Dorigo,
Dr. Andreagiovanni Reina,
Dr. Volker Strobel,
Eng. Alexandre Melo Pacheco

Graduation Session: December 2023

Academic Year: 2022-2023



IRIDIA

Abstract

In an open robot swarm task, different parties can introduce their machines to participate at any time, preventing the possibility to assume the willingness to cooperate of the entities, paramount due to the restriction imposed by local information. Using a blockchain economy, it is possible to establish an information market that promotes cooperation by rewarding the sharing of useful data, while punishing any attack of Byzantines, robots that harm the mission by spreading, either with malice or incompetence, deceitful information. This protection mechanism is not robust when a robot receives information that is locally shared among colluding robots, albeit evidently plagued by heavy position noise that makes it harmful. In this work I analyzed how is it is possible to prevent this by using a global reputation management system, based on a metric directly proportional to the wealth that the robots gain cooperating, and that is stored and visible on the distributed ledger of the blockchain. I used a simulation of a foraging mission, where robots must cooperate to locate a resource patch and transport the gathered items to a deposit site. I introduced a strategy able to compare the reputation of the information sources, and reject the ones that are too noisy, prioritizing higher reputations. I designed specific market rules which are able to harm the Byzantines, using the decentralized economy to extract wealth from the former and redistribute it to the cooperating robots, depleting the deceivers of resources needed to access the market. I introduced the possibility for the robots to request for debit wealth, created by a market that is specifically planned to exploit the tendency of the Byzantines to borrow money, rendering the disparity of wealth between the former and the cooperating robots more evident. I tested the market's stability and robustness against two waves of Byzantine attacks, at the beginning of the simulation and after a specific amount of time. My results showed that, by relying on the the globality of the wealth-related reputation, the swarm of robots is able to simultaneously defend itself against noisy and deceitful information. My study is the first to entirely rely on economic metrics to protect itself against multiple forms of non-cooperation. I envision that the field of swarm robotics could soon apply the theories of markets and economic inequality to furtherly harness the consolidating blockchain technology. **Keywords**: open swarm robotics, blockchain economy, reputation management.

Abstract in lingua italiana

In uno sciame robotico aperto, differenti soggetti possono aggiungere robot in qualsiasi momento, impedendo ipotesi sulla volontà di cooperare delle macchine, essenziale a causa delle restrizioni imposte dalla località delle informazioni. Con un'economia basata su blockchain, è possibile istituire un mercato che promuove la cooperazione premiando la condivisione di informazioni utili, punendo allo stesso tempo eventuali attacchi di Bizantini, robot che diffondono informazioni ingannevoli, sia per malizia che per incompetenza. Questo meccanismo di protezione non è robusto quando un robot riceve informazioni confermate localmente da robot che cospirano, sebbene visibilmente dannose. Ho analizzato come sia possibile prevenire ciò con l'utilizzo di un sistema di gestione globale della reputazione, direttamente proporzionale alla ricchezza che i robot guadagnano cooperando, memorizzata e visibile sul registro distribuito della blockchain. Ho simulato una missione di foraggiamento, in cui i robot devono cooperare per individuare una risorsa, raccoglierla e trasportarla in un sito di deposito. Ho introdotto una strategia in grado di confrontare la reputazione delle fonti di informazione e respingere quelle troppo rumorose, dando priorità alle reputazioni più elevate. Ho progettato specifiche regole di mercato in grado di danneggiare i Bizantini, utilizzando l'economia decentralizzata per impoverire questi ultimi e redistribuire la ricchezza ai robot cooperanti, privando i truffatori delle risorse necessarie per accedere al mercato. Ho introdotto la possibilità per i robot di richiedere debiti, creato da un mercato progettato per sfruttare la tendenza dei Bizantini a richiedere prestiti, rendendo più evidente la disparità di ricchezza tra questi ultimi ed i robot cooperanti. Ho testato la stabilità e la robustezza del mercato contro due ondate di attacchi di Bizantini, all'inizio della simulazione e dopo un periodo specifico. I miei risultati hanno mostrato che, basandosi sulla globalità della reputazione legata alla ricchezza, lo sciame di robot è in grado di difendersi simultaneamente contro informazioni rumorose ed ingannevoli. Il mio studio è il primo a fare affidamento esclusivamente su metriche economiche per proteggersi da molteplici forme di non cooperazione. Prevedo che il campo della robotica di sciame potrebbe presto applicare le teorie dei mercati e delle disuguaglianze economiche per sfruttare ulteriormente la tecnologia blockchain, in fase di consolidamento. **Parole chiave**: sciami robotici aperti, economia su blockchain, gestione della reputazione.

Contents

1 | Introduction

1.1. Swarms

A few research lines in robotics have focused in implementing a single high-performance and high-computational capable machine, able of effectively dealing with any task. However, this research direction may not be the only feasible solution to many real world application scenarios. Instead, the task could be completed by a large number of simple, cooperating agents, working together to fulfill the goal of the collectivity.

Remarkably, this swarm behavior has been evolved in nature [30] by some species of bees, ants and termites, that are in fact called social insects [26]. Swarms are composed of individuals that have limited capabilities, with respect to actions they can perform, communication range between the swarm members, typically local, and knowledge of the environment [30]. By interacting, the agents exchange information and are able to tackle otherwise difficult tasks: cooperating, the collective possibilities of the swarm are far beyond the ones of the single entity. In natural swarms this allows for a very high level of task parallelization, with huge increments in efficiency and coordination, even if no centralized authority to control the agents is present [30].

Swarm robotics studies how to exploit interactions among a large number of embodied and relatively simple robots, to design intelligent collective behavior [13, 14, 22]. This framework has been formalized by Dorigo [11] it must have the following characteristics:

1. scalability of performance as the number of robots increases;
2. it must be composed by many individuals, which can be either homogeneous, heterogeneous, or homogeneous per groups [12];
3. robots' performance should improve with cooperation. Robots themselves could be simple individuals having difficulty to carry out or complete the task on their own;
4. communication and sensing can only be limited and local, causing the single robot to have noisy and partial knowledge.

The design process often emulates social animals for the behavior of the single robots, with the goal of emerging cooperation from the collective interaction.

1.2. Interaction Between Robots

By ensuring frequent communication, the swarm can counterbalance the limitations imposed by local and partial knowledge, and noisy sensing or a possibly dynamic environment. Moreover, the decentralised aspect of the system gives tolerance towards failures of individuals.

In nature, many species tend to use a specific form of indirect communication called stigmergy [36], which consists of modifying the environment to signal other components of the swarm. For example, ants do this by leaving a trail of chemical compounds (pheromones) that are sensed by other ants; this leads to the formation of networks of trails. Since ants are attracted to bigger concentrations of pheromones, they will converge to the path with the highest frequency of deposition, which has been shown to eventually be the shortest one: the trail network is hence iteratively optimized over time [20, 49]. This example shows that communication is fundamental for the swarm to manifest distributed intelligence in robotics [28, 34, 51].

Many attempts have been made to recreate stigmergy in robotics because it is a very scalable way to communicate. For example, previous work has used stigmergy-based communication using chemicals to modify the environment [19, 34], or a smart environment using radio-frequency identification [29], augmented reality [43] and virtual pheromones [53]. However, since it requires specialized hardware, some researcher started using direct communication instead [3, 6, 15, 16, 23, 44, 46], which can be easily obtained usually using electromagnetic wave transmission. The easiest way to simulate the effect of pheromones with direct communication is to build a static chain of robots acting as beacons for the ones conducting the physical task [6, 15, 23]. While being very simple to realize, a major drawback of it is that part of the potential workforce is devoted to a role that is not directly performing the physical task.

An evolution of this method that deals with the problem is to use a dynamic chain of robots [16, 46], where each of them acts as a moving beacon, while also participating in performing the task: in this way the number of workers is maximized, and the physical interference of robots is reduced, since no static robot is present as an obstacle on the task path. A possible way to implement this, offered by social navigation, is for robots to share information about the last time they encountered a location. By being attracted to robots that encountered a site more recently, the swarm will eventually form a dynamic

chain between the sites, even in an environment with obstacles [16]. The robots will not only be able to localise the sites, but will also be able to estimate their position in the environment: this is particularly useful in environments that lack proper communication or localisation infrastructure, like Global Positioning System (GPS) [21, 46], or where communication redundancy is required for safety reasons.

1.2.1. Cooperative Foraging

Foraging is, generally speaking, the act of researching and gathering resources. In the particular scenario of central-place foraging, robots explore the environment to find resource patches; once one or multiple resource sites are found, robots exploit them, bringing the resource back to an accumulation site usually called deposit or nest [24, 36]. The importance of foraging derives from the great number of applications it is able to model: agriculture [1], garbage collection [2], search and rescue [35]; all of these tasks could require concurrent exploration and exploitation of the environment.

All the aforementioned setups can be efficiently solved using robot swarms, since task parallelization can allow the execution of both exploration and exploitation at the same time [4, 15]. Practically speaking, robots start without any global knowledge and must explore the environment to acquire information about where the sites of resources and deposit are. Once a food site has been found robots are able to start exploiting it; robots which already found it are able to communicate its position to others: there are, hence, two different methods to discover the sites. As time passes, information of the sites diffuses in the swarm, and, since new information is acquired in the meantime, robots reach a consensus, obtained by iteratively averaging personal and received information following a specific protocol.

1.3. Cooperation Among Robots

Since most swarms take inspiration from eusocial animals, in many works cooperation and correctness is assumed as a given constraint for each robot [14]. In real-world applications this assumption is too tight since non-cooperative behavior can manifest in many ways: if a robot, for example, malfunctions and starts to send wrong data, it can harm the operation of the entire swarm, without even knowing it; another example is where one robot is infected with malicious code, forcing it to behave in a non-cooperative way. Both malfunctioning and malicious robots could not be easily told from a cooperative one, and hence that behavior can be generalized using the Byzantine one: when a Byzantine fault occurs, there is no real consensus on whether a robot has, or not, failed. The name comes

from the Byzantine Generals Problem [31], a classic in computer science.

It has been proven by Strobel that even a single Byzantine robot could harm the whole swarm by spreading false information, if the robots do not filter the received information and hence act in a naive way [47]. Moreover, there could be cases where robots' goals differ from the ones of the rest of the swarm from the beginning. Albeit counterintuitive, this possibility is in fact more than reasonable in the case of open swarms. A swarm is defined as open when the robots can join and leave it at any time; obviously, no assumption on the behavior of the participants can be made: they may also have the sole intent of disrupting the operations of the swarm.

1.3.1. Blockchain Based Approach

To allow consensus on the real position of the sites to be resilient in the presence of Byzantines, a blockchain-based solution is possible, as it has been proven by Strobel using a blockchain-based token economy [48]. This makes the swarm robust against both zealot and Sybil attacks; the first being a common type of malicious attack where the robot always applies the same forgery to the data, irrespectively of the situation, while the second consists in forging a large number of new identities to prevent the traceability of the attacker or to flood the system with messages for a potential Denial-of-Service attack [27, 38, 50].

Blockchain can also be used to deal with another great challenge in swarm robotics: given the bottom-up approach used, it is hard to foresee what will be the actual collective behavior given a particular individual one, and results can be reached only by using careful design [52]. Obviously, it would be preferable and more practical to design a centralized controller to orchestrate all the activities of all the robots, however this would defeat the decentralised nature of swarm robotics. Pacheco proves that it is possible to use blockchain-based smart contracts, relying on a private Ethereum network [5], to coordinate robots during a foraging task, introducing robustness to Byzantines at the same time [37].

Van Calck proves that is possible to create a market in which information is exchanged to regulate the robot activities. Each robot owns some personal wealth, stored on the blockchain, that it trades with other robots. They show that it is possible to design market rules that lead to higher accumulation of wealth by cooperative robots and a loss of wealth by Byzantine robots. Using a behavior who takes inspiration from economic agents, cooperation may emerge even in a context where robots behave in a selfish way [54]. To detect and reject Byzantine behavior, a two-layer protocol is used: systemic protection,

based on the market rules and economic incentives, and an individual one, improving the decision-making strategy of the single robot. A robot decides to acquire information from other robots using a so-called skeptical logic where new information is accepted only when it's not too different, considering a given threshold, from previously received information. However, this could lead to the rejection of good information coming from honest robots when it is not confirmed by others. This issue is especially visible when no Byzantine robots are present or already identified, and it is very depending on the selection of the threshold, and the degree of attack performed by the Byzantines. Instead of relying on local information only, to detect the Byzantines, a systemic collective estimation of the robots' reliability can prevent the unjust penalization of honest robots.

1.4. Reputation and Trust Systems

Trust and reputation models present a convenient solution in multi-agent systems where the selection of a partner is fundamental, like in the case of the presence of Byzantine agents [39]. Trust towards an agent is defined as the subjective belief other agents have about its degree of reliability or competence. In the particular case of a market-based economy, it will represent the will of other agents to accept its data. Reputation represents the recorded history, or perception, of an agent past behavior and performance; within a multi-agent system, reflecting the agent's trustworthiness [39].

Traditionally, there is no consensual differentiation in the definitions of trust and reputation models, but in this work I will rely on the definition of Pyniol & Sabateur-Mir [39] that trust implies a decision about interacting or not with a partner, which is based on Castelfranchi & Falcone's idea that trust implies delegation [8, 18]. This implies that trust could be considered as a Boolean information. In the scenario of open swarms, standard reputation models cannot be readily used because they fail to foresee the changes in the system caused by openness [25]. Moreover, if cheating is present, more actions shall be taken in consideration.

The AFRAS model, by Carbo [7], uses both personal and social information to compute a subjective trust measure. Although including reliability measures on all the sources of information used to compute it, this value is only based on the last interactions the partner had with both the agent and third parties. At this point, the work does not provide any mechanism to decide based on the obtained value.

Rasmusson & Jonson's [41, 42] model relies on specific rating agents, acting as external evaluation agencies, to compute a subjective reputation measure that it then provides to all other agents. In addition, partners can invest wealth to encourage agents to remember

them in a more positive way, believing that this monetary incentive will improve truth telling, without allowing any bribery. In Sen & Sajja's [45] model, the computed value is private and relies on different types of direct experience, a noise-less one, and a noisy one, that may differ from reality. At partner decision time, the agent may also query other ones, inside its communication range, aggregating their reviews about the partner.

1.5. Original Contribution

Van Calck proves the fundamental hypothesis that, in a market that rewards actions benefiting the swarm and punishing the Byzantines, the wealth of a robot is proportional to its degree of cooperation [54].

Considering this, I conjecture that a candidate metric for reputation and trust could be the wealth of the robots, more specifically, either the current wealth or its history of variation could be used. By using a blockchain-based market, where all rules are enforced with tamper-proof smart contracts, the desired wealth evolution for the participants can be obtained. Moreover, these rules act as a centralized authority, offering an easy way to control the swarm. Each robot's wealth is stored in the blockchain, which serves as an external memory storage for the swarm. Since the wealth of a robot is the result of all its past interactions with other robots, the value of trust is created in a distributed way by all the participants of the swarm, but relying on a centralized infrastructure, hence removing the need to account for reliability of the sources of information.

The focus of my work is to provide an in-depth analysis of how wealth could be used as a metric for a reputation system with the purpose of rejecting Byzantines as possible trading partners. To do so, I develop economic and inequality metrics to interpret the evolution of the multi-agent economy. I relate those metrics with task-related performance measures [9], regarding the swarm physical job and compare them with a naive and sceptical reference robot behavior.

I prove that wealth-based selection is a "cost-free" protocol, having always better performances than scepticism in the absence of Byzantines, and obtaining better performances than robots that do not filter information, acting in a naive way, in the presence of Byzantines. As part of my thesis, I also introduce and test an additional protection layer for naive robots, using the reputation for weighting the information for the reliability of its source during the combination and averaging process.

My simulation results show that trust can be used to tune the outlier detection system, improving performances while still obtaining the desired market evolution.

Moreover, I explicitly consider an open swarm where new members join after a given time, and therefore discuss entry reputation and show their effect on the stability of trust system, market and performances.

1.6. Thesis Structure

The remaining chapters of this thesis are organized as follows:

- in Chapter 2, I outline the characteristics of the simulation, namely the robots' physical and logical properties, the environment where the foraging task takes place, the rules of the market and how information exchange in the market takes place, and describe the metrics I will use to interpret the results;
- in Chapter 3, I use the developed metrics on the different markets and protection strategies to show the results;
- in Chapter 4, I comment on the obtained results and discuss limitations and possible future improvements.

2 | Methods

Trust protocols are tested in a simulation of central place foraging, with the goal of the swarm being to collect resources from the food site and transport them to the nest; during this process robots tend to get lost, because of the odometry noise that plagues them. The most efficient way to not get lost is to build a dynamic chain of robots, going back and forth between the two sites, which however is fragile to the attack of Byzantines. I test a system using reputation, based on the market, to imbue the chain with robustness against Byzantines' attacks.

The swarm is composed of 25 robots, with a varying number of them as Byzantines; the other ones are considered honest. A simulated market, based on a blockchain that permits the contracts to be issued, is used by the robots to build their trust and reputation measures for the others. It is constituted of a data structure, shared between all robots, and relative procedures to operate on it. The simulation may last from 15000 to 50000 steps.

2.1. Environment

The experiment is performed in a rectangular-shaped ground, shown in Figure 2.1, with smooth surface and without any obstacles in it. The chosen values for width and height are respectively (1200, 600) units. Two round-shaped patches, of fixed size and position, are present, representing the foraging and the nest sites: the former is where robots gather resources, while the latter is where they deliver and deposit the collected items. I choose equal radius for both sites, r_f=r_n=50 units, and coordinates P_f=(200, 300) and P_n=(1000, 300) units, respectively for horizontal and vertical positions of centers of resource and nest sites; sites' placement is such that they are symmetric with respect to the vertical median axis of the environment.

Resources consist of generic items, with no size; no real action is required for picking and dropping: robots can do that by simply reaching a specific position inside the specific site. To randomize the time the robots spend to pick up and drop an item, the locations where

items are generated inside the resource site and must be dropped in the nest are decided randomly, with uniform distribution, on the whole surface of a site; this also prevents the formation of large robots assemblies, moving very close to each other, and forces diversification of the chain so that no robot gains an advantage thanks to a reiterated shorter path.

Since the resiliency to destruction of the chain is the main research focus, I decide to use this simple environment; a more complex environment, in fact, could produce fluctuations and complex responses that could mask the effectiveness of the protocol.

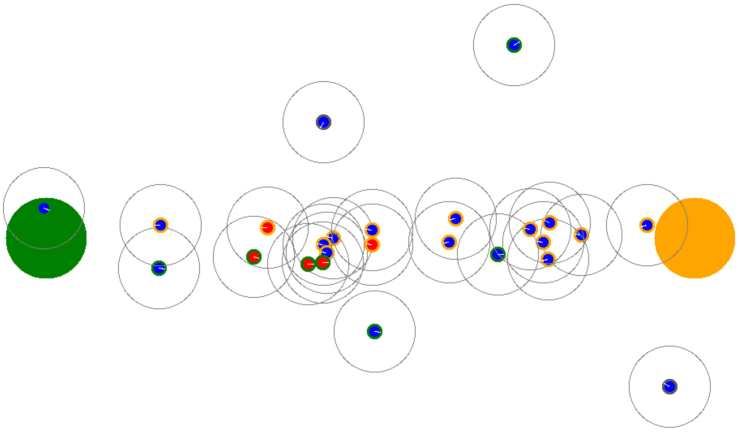


Figure 2.1: Robots creating the dynamic chain in the foraging environment: resource patch is shown in green, nest in orange. Honest robots are blue, while Byzantines are red; the color of their outline indicates their current target site. The grey circle around the robots is the communication and sensing range, r_c=r_s.

2.2. Robots

Robots are represented by point-like particles that can freely move in the environment and cannot collide with each other. They do not possess any physical nor dynamical property, and can hence accelerate instantaneously, without any slippage; the translation velocity is limited to V_{MAX}=2.5 units/simulation step, while the rotation is not, meaning they can instantaneously change direction.

The robots are spawned randomly, with uniform probability, in the environment. At the beginning of the experiment, each one of them is assigned a static ID, starting from 0 and increasing with the spawn order, and Byzantine status. Irrespective of the number of Byzantines, the honest robots will always have the lower IDs, while the Byzantine will

be assigned the higher ones. I decide the number of Byzantines to be in the range $[0-8]$; the amount is bounded by the theoretical limit of dishonest robots that this protection protocols can deal with, 33% of robots' population.

I define as Byzantine robots the ones that systematically try to harm the task every possible time they interact with others. To protect itself from the Byzantines, each robot is also assigned a static behavior, a strategy to use the computed trust value to estimate if a possible partner is untrustworthy.

Robots are tasked to collect items and transport them back to the nest site. They will act differently if they are searching for resources or if they are carrying one, representing the two situations during its round trip between the sites, to the foraging site and back, as presented in Algorithm 2.1. In the former case, if the position of the resources is known, the robot will try to reach them; otherwise, it will start an exploration phase, activate the site sensor and open the communication channel, for the possibility to exchange precious data with partners. Sensing and communication ranges are equal and defined at r_s=r_c=50 units. The latter case is similar, but the decision is made upon the nest site being known or not. Since there is no possibility of collision, the navigation strategy is simply to drive towards the current target in a straight line.

Algorithm 2.1 Robot State Machine

```
1:  if not carrying food then
2:      if food location is known then
3:          go to food
4:      else
5:          activate sensing
6:          open communication
7:          begin random exploration
8:      end if
9:  else
10:     if nest location is known then
11:         go to nest
12:     else
13:         activate sensing
14:         open communication
15:         begin random exploration
16:     end if
17: end if
```

To enable each robot to trade and access the market, each one has a blockchain wallet where its wealth is stored; this wealth is the currency used in the market and it is exchanged on the blockchain. Each robot's secondary goal is to maximize its wealth: apparently, this may seem selfish.

2.2.1. Odometry and Exploration

Robots arrive in the environment without any knowledge of it or their position. They do not have access to any centralized nor external position system (e.g. GPS), so they will localize themselves using odometry. From the start of the simulation, for each time step, the robots keeps track of its linear motion and change in orientation, estimating the cumulative distance from a given starting point.

When within its sensing range, a robot records the precise position of each site separately, as shown in Table 2.1. It uses the Cartesian distance in its solidary reference frame computed from the odometry, and overwriting any preexisting older record of it, setting the age of the information at 0. At each time step, the robot updates the distance of each know site, and increments the age attribute of 1.

	Distance	Age	Trusted
FOOD	(200, -5)	78	True
NEST	(1000, -35)	421	False

Table 2.1: Navigation table example

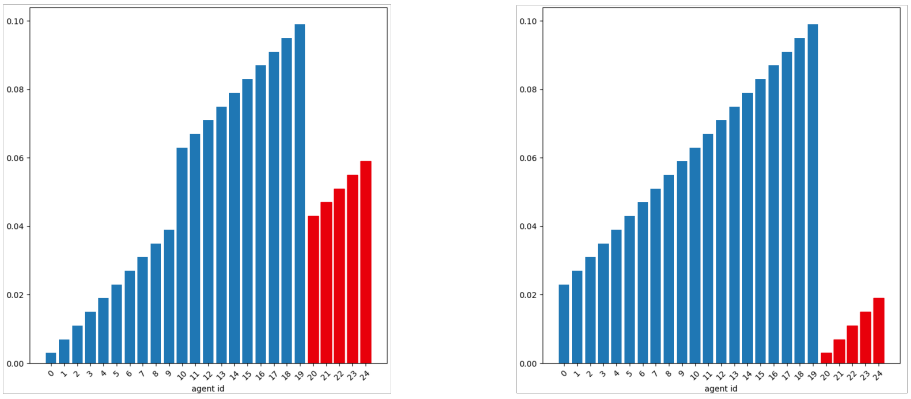
When willing to find a site, the robot starts a Levy random walk, inspired by nature [17] and that can result in optimal search when the targets are sparse [40]. When the current target site is known, the robots will compute the linear and angular distance it must move its own odometry. In the case it reaches the expected location of a site and does not sense it, it will label the data as not known and start the exploring phase.

To simulate imperfection of sensing, each robot's odometry is plagued by noise, consisting in an additive rotation to its intended movement. The cumulative effect of these errors causes the robots to move in circular trajectories, with radius depends on the inverse of the intensity of their odometry noise, whilst they believe to be moving in straight lines. By moving, each robots contaminates sites' data in its possession with it. From this derives the rationale behind the preference towards newer data of the sites.

Noise is drawn from a uniform probability distribution, in the interval $[\mu - \sigma, \mu + \sigma]$, with

$\mu = 0.051$ and $\sigma = 0.05$. This value is such that an averaging protocol is needed, but still sufficient to work against it: if the chosen noise was too low, exchanging information will be useless since all robots could be able to navigate without problems, if it was too high, even with perfect data, the robots would not be able to reach the site. The mean μ has a value such that the best robot has $noise_{best} > 0$. Each robot is assigned a fixed noise intensity at the start of the experiment, increasing with their ID, but also depending on the odometry performance of Byzantines η with respect to the honest robots: if $\eta = average$, the mean of Byzantine group noise equals the mean of the whole distribution, and half of the honest will be better than them, the rest worse. If $\eta = perfect$, Byzantines will have the lowest odometry noise possible, and the honest will always be worse. Figure 2.2 shows an example of noise distribution, in the case of five Byzantine robots, for both the possible odometry noise performance. Equation (2.1) shows the procedure to select the noise for each robot. The index i is selected considering the robot ID and the noise performance for Byzantines; $|P|$ represents the population numerosity.

$$noise_i(\mu, \sigma, P) = \mu + 2\sigma \frac{i - \frac{|P|}{2}}{|P|} \tag{2.1}$$



(a) Average Byzantines. (b) Perfect Byzantines.

Figure 2.2: Noise distribution by ID for different Byzantine noise performances. Red color indicates Byzantines.

I assign the same value of noise for each robot in all the experiments, instead of randomly selecting it, to understand directly their reliability: it can be proven that the foraging performance depends on the navigation drift. I want to considered higher level of noise to be somehow equal to being Byzantine such that the security protocol will punish,

to a smaller extent than Byzantines, robots spreading highly noisy information, leaving unpunished the honest ones with lower level.

2.2.2. Communication and Social Navigation

Cooperation not only consists in participating to the physical task with all the other robots, but also in sharing information about the sites of importance for the mission: to obtain the missing or newer data, robots can also rely on others selling it, upon agreeing to become commercial partners.

Whenever two robots are within range, they will both try to sell the information to each other. If one agrees to become partner of the other, the transaction is written on the blockchain. Then, the seller is added to the list of contributors that gave information to the buyer for this round trip, from nest to foraging patch and back; this happens for as many instances as the number of times it sold the information to the buyer. The list of its partners is cleared every time it drops an item. When a robot buys information, it must add the distance with the seller, both linear and angular, before being able to use it. Every time two robots meet, they also update their version of the blockchain with the longest one between the two.

Once the data is referenced to a robot frame, it may decide to use it to update its navigation table: if the relative site is not known, it will simply use it as it is, otherwise it will combine it with the one it already possesses using an averaging protocol for both the distance data and the age. I test multiple methods, considering only age and/or reputation of the seller: if only age is used, age of combined data equals the algebraic mean of the preexisting and new one, otherwise both data and age will be updated using the same weighted average. In the case of reputation, it may use different quantities: it could either be by comparing the buyer reputation with the seller's one, or comparing this last one with the average of all the robots. Data is always rejected if $\text{age}_{seller} \geq \text{age}_{buyer}$. Since robots have different levels of odometry noise, there may be cases where older data from robots with low drift is better than newer data from noisy measurement. The requirements to consider this is for the robots to estimate their own odometry noisy, which has been proved to yield similar results than simply using age as a quality for the information, as has been proven by Van Calck [54].

Equation(2.2) shows the test to assess the validity of the messages

$$buy()\ if\ age_{seller} \leq age_{buyer} \tag{2.2}$$

Then, if a robot decides to combine the information, using a Boolean valued function $combine\_info()$ specific for each individual behavior, the update of local information follows the steps of Equations (2.3) and (2.4)

$$\vec{V}_{new} = w^{\vec{V}}_{buyer}\ \vec{V}_{buyer} + w^{\vec{V}}_{seller}\ \vec{V}_{seller} \tag{2.3}$$

$$age_{new} = w^{age}_{buyer}\ age_{buyer} + w^{age}_{seller}\ age_{seller} \tag{2.4}$$

Table 2.2 presents the different method I tested for computing the weights, with R representing a specific reputation used; $\mathcal{A}_a$, $\mathcal{A}_{R_{mean}}$ and $\mathcal{A}_{aR_{mean}}$ use respectively age of the information, reputation of source and both, in a running average procedure. $\mathcal{A}^{now}_{aR_{buyer}}$ and $\mathcal{A}^{now}_{aR_{mean}}$ instead use the current value of the reputation; every time a new piece of information is acquired, the procedure also updates the weight of all previously bought data, that is stored in the vector τ. Element $\vec{R}^{now}_{seller}$ is a vector storing all updated reputations, and in this case, elements $\vec{V}_{seller}$ and age_{seller} in Equations (2.3) and (2.4) take a vectorial form, containing all velocities and ages of all data acquired and stored in τ, and participate in a scalar product with the relative weight.

	$w^{\vec{V}}_{buyer}$	$w^{\vec{V}}_{seller}$	w^{age}_{buyer}	w^{age}_{seller}
$\mathcal{A}_a$	$\frac{age_{buyer}}{age_{buyer}+age_{seller}}$	$\frac{age_{seller}}{age_{buyer}+age_{seller}}$	0.5	0.5
$\mathcal{A}_{R_{mean}}$	$\frac{R_{mean}}{R_{mean}+R_{seller}}$	$\frac{R_{seller}}{R_{mean}+R_{seller}}$	$w^{\vec{V}}_{buyer}$	$w^{\vec{V}}_{seller}$
$\mathcal{A}_{aR_{mean}}$	$\frac{age_{buyer}R_{mean}}{(age_{buyer}+age_{seller})(R_{mean}+R_{seller})}$	$\frac{age_{seller}R_{seller}}{(age_{buyer}+age_{seller})(R_{mean}+R_{seller})}$	$w^{\vec{V}}_{buyer}$	$w^{\vec{V}}_{seller}$
$\mathcal{A}^{now}_{aR_{buyer}}$	$\frac{R^{now}_{buyer}}{R^{now}_{buyer}+\sum_{j\in\tau} R^{now}_{seller_j}}$	$\frac{\vec{R}^{now\ T}_{seller}\cdot}{R^{now}_{buyer}+\sum_{j\in\tau} R^{now}_{seller_j}}$	$w^{\vec{V}}_{buyer}$	$w^{\vec{V}}_{seller}$
$\mathcal{A}^{now}_{aR_{mean}}$	$\frac{R^{now}_{mean}}{R^{now}_{mean}+\sum_{j\in\tau} R^{now}_{seller_j}}$	$\frac{\vec{R}^{now\ T}_{seller}\cdot}{R^{now}_{mean}+\sum_{j\in\tau} R^{now}_{seller_j}}$	$w^{\vec{V}}_{buyer}$	$w^{\vec{V}}_{seller}$

Table 2.2: Different methods for computing averaging weights

By spreading their information, robots try to average all odometry errors with the goal of creating and maintaining a dynamic chain of robots coinciding with the shortest path between the foraging and the nest sites, that the robots must follow to work in the most efficient way. Information exchange takes a single time step in the simulation, so they are able to go back to the physical task right away, without having having to sacrifice themselves to act as a static beacon.

Byzantine robots send a modified version of the data they posses and use, applying to it a rotation of a given fixed angle. This has the intent of bring the victim of the attack to get far from the chain, losing the possibility of obtaining honest data and getting lost, having

then to roll back to the exploitative behavior, with an effect on the swarm productivity. Applying a 90° rotation is the worst possible case, since it could push the victim on a perpendicular path with respect to the chain.

A preliminary study of all the combination methods based on reputation show no promising results in protecting honest robots using a naive behavior. Possibly, this implies that Byzantine information should be, more conveniently, stopped before it reaches the other robots. Considered its simplicity ad effectiveness, method $\mathcal{A}_a$, based on age of information alone, is the only one worth using in this work.

None of those methods are able to protect the honests, possibly meaning that the Byzantine information and I hence omit them. I omit the results on the different combination methods, used as a protection for naive robots: none of the methods using reputation is able to protect the honest, implying that Byzantine information must be stopped before it reaches the other robots.

2.3. Transactions Regulations

Whenever a robot receives navigation information from a seller, this is effectively an economic transactions. The seller expects a compensation for the data it provides. All the transactions exchanged on the blockchain use smart contracts, which have coded-in regulation which is inherently tamper-proof. These rules act as a systemic protection for the robots, being able to control the evolution of the wealth of each member of the swarm, according to its degree of honesty.

2.3.1. Penalization of Outliers

This system relies on the assumption that the Byzantines are a minority of the total population, meaning that their information, considerably different than the one of honest, is the outlier in the distribution. This can be used to value less that data.

When a robot completes a round trip, the smart contract distributes the reward among the robot that transported the item and all robots that contributed useful navigation information. The more similar each piece of information sold is, compared to all the data provided by the partners, the higher the reward is. The similarity is assessed considering a window of 30° on the normalized angular distance of the navigation vectors, computed

by the function $similar(t_i, t_j)$, as shown in Equation (2.5).

$$similar(t_i, t_j) \triangleq \begin{cases} 1, & \text{if } |o_{t_i} - o_{t_j}| < 30^\circ \text{ and } location_{t_i} = location_{t_j} \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

Computation of the weight w_{t_k} of a transaction t_k follows as shown in Equation (2.6).

$$w_{t_k} = \sum_{j \in \mathfrak{T}} similar(t_k, t_j) \tag{2.6}$$

where $\mathfrak{T}$ is the set of all transactions t_k. Transactions will be valued more the higher the number of similar transactions among the ones recorded is, using the weight w_{t_k}. Since $similar(t_i, t_i) = 1$, then $w_t \geq 1 \forall i$.
For a swarm composed only by honest robots, each one of them will have equal wealth $\frac{1}{|P|} W_{total}$ of the total one, where $|P|$ is the population's size; in the case of 25 robots, this value is $W^{\infty} = 4\%\ W_{total}$. When a single Byzantine is present, this scheme will force its wealth to be, on median, $W^{\infty}_{Byzantine_i} \approx 3\%\ W_{total}$, while the one of the honest to be $W^{\infty}_{honest_i} > 4\%\ W_{total}$.

2.3.2. Penalization Mechanism with Staking

The effect of the previous scheme reduces with the number of Byzantines, with the difference in wealth of the two groups becoming smaller. Introducing staking, each robot that wants to sell information must deposit some tokens on the blockchain as a guarantee for the reliability of the information it wants to sell. A robot selling information will be considered only if it's able to stake the required amount, and will regain it once the buyer is able to finish a round trip. The cumulative amount of all stakes gathered by buyer will take part to the redistribution of reward, using the same shares presented in Section 2.3.1.

Despite the penalization mechanics being not sufficient for outlier identification, the Byzantines will still have lower median wealth than the honest. By choosing as the staking amount the median wealth the robot would have if only honest would be present presented in Section 2.3.1, the Byzantines will see their wealth eroded with time, until they reach a level where they're not anymore able to stake. When that happens, they are excluded from the information market. This promotes the diffusion of honest information only: it is in fact $r_{k,seller_i} - stake_{k,i}$, the difference between $r_{k,seller_i}$, the positive rewarded quantity for the transaction t_k relative to a concluded contract, and $-stake_{k,i}$, the negative quantity staked for same transaction by $seller_i$, that drives the wealth evolution: honest robots tend to have a positive difference, and will see their wealth increase with

time. For 25 robots, the base stake amount is selected to be $stake_0 = 0.04$ tokens.

2.4. Market

Items and information are sold in the distributed market, exchanged at a fixed price payed using tokens stored in the blockchain; each robot starts with a fixed and equal amount of tokens depending on the market which is used in the experiment. The value of the items is set at a fixed amount of $p_I = 1$ token. Using smart contracts it is possible to program the payment of rewards and use the information market to control the swarm in its foraging task.

All markets rely on the delayed payment system: even if all robots would measure and disclose their own odometry noise, it's not possible to trust what they say about the quality of their information before buying it, given the hypothesis that any robot could be Byzantine. Partners that offer the information for free and accept to be payed once the item is delivered to the nest, will foster the diffusion of honest information with actions that reward more the improvement of probability of to reach the nest.

2.4.1. Information and Foraging Market

In this market, robots have two forms of reward: firstly, they receive an amount for selling the foraged item in the site. Secondly, every time a robot deposits an item in the nest, it shares a part of the value of the item with the partners that helped it complete the last round trip, following the scheme shown in Section 2.3.1. The selected share of reward given to the partners $\sigma = 50\%$, leaving the rest to the robot depositing the item. The reward payout for a certainty transaction t_k is shown in Equations (2.7) and (2.8), respectively for the buyer and the sellers

$$r_{k,buyer} = (1 - \sigma)\ p_I,\ \forall t_k \in \mathfrak{T} \tag{2.7}$$

$$r_{k,seller_i} = \frac{w_{t_k}}{\sum_{j \in \mathfrak{P}_i} w_{t_j}} (\sigma\ p_I + \sum_{j \in \mathfrak{P}_i} stake_j),\ \forall t_k \in \mathfrak{T} \tag{2.8}$$

where $\mathfrak{T}$ is the set containing all transactions of a buyer, related to the set of all partners $\mathfrak{P}$ of a buyer, $\mathfrak{P}_i \subseteq \mathfrak{P}$ is the subset where seller_i is involved, and $stake_j$ is the amount staked by it for a given transaction. The wealth of a $robot_i$ is therefore updated following

the rule of Equation (2.9)

$$\begin{cases} W_i^0 = W_0 \\ W_i^{t+1} = W_i^t + r_j, \quad \forall\ rewards\ r_j \in \mathfrak{R}_i \end{cases} \tag{2.9}$$

the collection of rewards received by $robot_i$ and not yet accounted. Every time a robot sells an item, new wealth is introduced in the systems: this market must be considered open, since it's only lower bounded, and not zero-sum. For this reason, and since the robot that deposits the item is always rewarded before it splits its reward, there is no possibility that it won't have any money to pay the creditors.

It must be noted that paying the robots for their physical work is the most natural thing to do to adhere to a simulation of a human society and attribute to the robots the behavior of an economic entity. To enable the staking mechanism, each robot starts with an initial wealth of $W_0 = 1$ token.

2.4.2. Information Market

Rewarding foraging introduced the issue that the Byzantines could use the physical work to counteract the effect of the staking penalization mechanism, and be able to access the market again. I decide hence to test a market where there is no reward for foraging: once a robot delivers an item, it won't receive any compensation for the physical work done, but it will still have to pay the share for the received information to the partners. Moreover, the reward share for the partners, $\sigma = 100\%$, hence omitted in Equations (2.10) and (2.11) showing the rewards for the buyer and the sellers respectively

$$r_{k,buyer} = -p_I \tag{2.10}$$

$$r_{k,seller_i} = \frac{w_{t_k}}{\sum_{j \in \mathfrak{P}_i} w_{t_j}} (p_I + \sum_{j \in \mathfrak{P}_i} stake_j),\ \forall t_k \in \mathfrak{T} \tag{2.11}$$

where the negative $r_{k,buyer}$ represents the fact that the buyer must pay with its own wealth the information received, since no foraging reward to split is given. Wealth update follows the same rules as in Equation (2.9).

No new wealth is introduced in the system, making this is a closed market, upper and lower bounded, and a zero-sum game. Despite apparently being against the rationale of a fair reward for the work done, this system is conceived so the honest thrive on Byzantines, depleting them of their resources. If a buyer robot can provide on average good enough information, it will recover this wealth once it becomes a seller itself: honest robots will

settle to a level of wealth proportional to the quality of its information, while Byzantines will keep losing wealth and this excess will be eventually earned by the honest. In the case no Byzantine is present, the honest will on average oscillate around the average of all the robots, which is the starting wealth.

Since no new wealth is introduced with time, I set the starting value of initial wealth $W_0 = 7$ tokens, with this value coming from the analysis of the mean wealth in the experiments that reward foraging.

2.4.3. Default and Debit

When delayed payment and staking penalization are paired with the closed market, no new wealth is introduced in the market before the payment of creditors. It may happen that the debtor finds itself without enough wealth to pay all the creditors. When this happens, the debtor defaults, with the effect that it receives information for free. This is a necessary evil: I test a payment scheme where a financial audit is performed on the information buyer, preventing robots without any wealth to receive information, proving that it yields a great drop in performances. This happens because inhibiting the spreading of information harms the effectiveness of the averaging protocol, and harms the honest the most since they are the biggest part of the population: robots that finds themselves without tokens to pay for information would more probably be honest starting from unfortunate situations.

For what it concerns Byzantines, the important thing is that they still cannot create a new contract if they do not have any wealth, being still prevented to spread false information. Having them receiving information for free is not a big deal, since they cannot recover wealth with physical work, even if the are still contributing to the task of the swarm. Despite that, this is an unfair practice that damages the honest robots that have the better information, potentially reducing their possibility to stake and spread the best information.

I introduce a mechanism for a fairer commerce: I remove the closed market assumption; here the wealth is injected by the debtor creating actual debt: this guarantees that the creditors will always receive compensation, but opens the possibility of having negative wealth; in this situation markets will not be bounded anymore. This somehow opposes to the sense of blockchain tokens as a regulated tender, but the foraging market is not upperly bounded to begin with; anyway, this will help honest robots with good information to stake more. Since this works also with wealth necessary for staking, this may also help honest starting with bad initial conditions to have a second chance, and build up some

good reputation with this supplementary wealth; the Byzantines anyway will anyway keep losing wealth with the staking mechanism, plunging way below the zero, helping the differentiation from honest.

2.5. Econometrics

In the previous section I showed the market and the mechanics that force it to evolve in a way that is beneficial to using wealth as reputation. Before showing the different strategies the robots use to make deals with partners, I show how to analyse the market and the tools that I use to verify the benefits of having wealth as a form of reputation.

There are three major metrics which yield the higher informativity for what it concerns the creation of the chain, the rejection of deceitful information, and the availability of resources in the system. To better analyse my hypothesis, I divide the robots in three noise groups: honest will be *good* or *bad*, depending if their noise level is above or below the mean of the honest; Byzantines will always belong to the *byzantine* group.

2.5.1. Wealth

The evolution of wealth in the system shows if the reputation protection is interfering with systemic protection based on contracts. Ideally, as in the previous work [54], while wealth of Byzantines should drop to zero, or below in the case shown in Section 2.4.3, honest should have steady or, steadily increasing wealth, proportional to their drift: the *good* noise group should perform better than the *bad* one, but since the averaging protocol can deal with noise, I would like not too have too much variance in the performance of the honest.

I consider two different types of wealth: available wealth, or reward, W^t is the quantity of tokens that is, at a certain moment t, in the wallet of a robot. It represents the possibility of investing in staking of a robot and a bigger potential of sharing information with the other robots, ultimately yielding the possibility of higher personal return. In the long run, it's higher in the robots that obtained more returns on investments, indicating possibly a better quality of the information sold.

Total wealth $\mathscr{W}^t$ instead accounts also for the values of investments of a robot, and it is equal to the $\mathscr{W}^t = W^t + \sum_{j \in \mathfrak{P}_i} stake_j$, respectively available wealth and the sum of all staked wealth a robot entrusted to partners, at a given time t. The difference $\mathscr{W}^t - W^t$ is able to show the state of the formation of the chain: in fact, a large difference means that the staked tokens are still frozen, and can indicate an incomplete or broken chain for

a Byzantine attack, causing many robots to get lost and not be able to drop the items, and unfreeze the staked funds. A little difference, steadily or slowly varying, shows that the stake are released quickly, by robots following the shortest path between sites.

2.5.2. Foraging Performance

The amount of items collected by a single robot indicates the quality of information it possess: if it is able to forage at a high pace, it must be able to find the sites with good efficiency. Extending this reasoning to the whole swarm, if the good information is able to spread and the dynamic chain is stable and robust to Byzantine attacks, then the number of items foraged by the swarm must be high; instead, if the Byzantines succeed the performance must drop. The best protection system is the one that guarantees the maximum efficiency at completing the task.

For this, I relate the previous metric of wealth with the foraging performance, and use the mean number of collected items at a given time as a method for comparing different strategies. In this case I consider separately honest and Byzantines groups: the best situation would be for honest group to forage at the same rate of a swarm composed of only naive robots without any Byzantine present, because in that situation information spreads completely and without any threat; Byzantines, depending on the used market, should collect the same amount, if they cannot recover lost wealth with it, or the less possible. The number of items collected could be measured at the start, to understand how fast the robots start the task, or at regime to understand the resilience against a Byzantine attacks.

2.5.3. Acceptance Rate of Transactions

All transactions are represented as messages between two parts, $robot_i$ and $robot_j$. A transactions has to cross three stages: when the seller i tries to sell the data, it is considered accepted, increasing its count q_i^A; then, if the buyer j accept the data, considering its recency requirements shown in Equation (2.2), it is considered validated, increasing the count q_i^V. Finally, if $robot_j$ decides to combine the data, following its behavioral strategy, it is considered combined, resulting in the actions shown in Equations (2.3) and (2.4), and it counted in q_i^C.

To understand the resiliency of the strategy against Byzantines, I consider the amount of data they try to sell which is being accepted. The ideal situation is that the data of the Byzantines is rejected all the time, while the data of each honest robot is rejected proportionally to its noise, with the information coming from the best honest robot having

the highest acceptance rate. I compute the acceptance rate of information coming from a $robot_i$ as $q_i = \frac{q_i^C}{q_i^V}$, using the proportion of the number of messages combined into local information by another $robot_j$ $\forall j \neq i$, over the number validated messages sold by $robot_i$. It must be noted that, on average, all robots experiences a rate of validated over accepted messages $\frac{q_i^V}{q_i^A} = 10\%$.

2.6. Reputation and Robot Strategies

After the information passes the validity test on its age, the robot must decide if it should use it. A behavior describes the rules that a robot follows to complete this step.

The goal of this work is to establish these rules using a metric of reputation shared among all robots and based on wealth and use this value in a trust management system to lessen the impact of Byzantines. I develop multiple behaviors that use different functions of the wealth. Each robot has a fixed deterministic behavior for the whole duration of the experiment. I study an homogeneous swarm: honest and Byzantines follow the same rules, although the latter lie on the information.

2.6.1. Computing Reputation

I consider both reward and total wealth to compute reputation. The former is intended to punish the Byzantines: due to staking, a robot which gets lost due to dishonest information will freeze for a longer time the staked amounts of the Byzantines and cause them to have lower mean available wealth, while if a buyer receives honest information, it will more likely release stakes at a higher pace. The latter instead should benefit robots with a long history of cooperation, good information quality and a lower odometry noise; this last cause could initiate a limited time snowball effect, since these robots can navigate better when the chain is not formed yet.

Regardless of the chosen metric, reputation always assumes an integer value in the range $[-10, +10]$, where 0 represents the neutral reputation given to the robots at the start of the experiment. After normalization, following an heuristic based on the type of wealth used, an hyper tangent function filter is used to restrict the value in the chosen interval.

Equation (2.12) shows how reputation of $robot_i$ at time t R_i^t is computed

$$\rho_i^t(w) = \begin{cases} W_i^t - 0.25\dfrac{7500}{wallet\ age_i^t}, & if\ w := W \\ \mathscr{W}_i^t - 3\ \min\limits_j(\mathscr{W}_j^t), \quad \forall j \in P, & if\ w := \mathscr{W} \end{cases} \tag{2.12}$$

$$R_i^t = 1 + 10\ tanh(\rho_i) \tag{2.13}$$

The heuristic using available wealth is able to penalize poor robots not able to invest, robots not able to get their stake back and also deter the use of Sybil attacks, since it penalizes robots with wallets with smaller age. The second heuristic penalizes being poor and uses the distance with the poorest robot in the population. Initially, I test heuristic based on mean wealth, but the ones presented here achieve higher performances.

The reputation of a partner is computed individually by each robot every time it's needed by reading the value of the chosen metric on the blockchain. Then, the buyer robot uses this value to compute the trust measure specific of each behavior, deciding if it should use the bought information.

2.6.2. Behaviors Using Instantaneous Value of Reputation

In this part, I show behaviors using the current value of the reputation. The advantages are the simplicity, and the fact that the value is always available from the start.

$\mathcal{C}$ Behavior

In this simple strategy, only sellers with higher reputation than the buyer will be considered as good trading partners. This is completely based on the hypothesis that the quality of information and odometry noise are proportional to the wealth, and hence to the reputation. Moreover, the better the quality of information a robot sells is, the higher its reputation will be; hence, it will use less often information coming from other sources, preventing *good* robots from being contaminated by the noisy information of mediocre sellers. Equation (2.14) and Table 2.3 show the information combination method and the parameters used for this behavior

$$combine\_info()\ if\ R_{seller}^t \geq R_{buyer}^t \tag{2.14}$$

	Reputation metric R
$\mathcal{C}$	reward, total wealth

Table 2.3: Parameters and used values for $\mathcal{C}$ behavior

Using buyer's own reputation may be dangerous. For example, if a Byzantine robot is able to amass wealth by exploiting very *bad* robots, it will obtain high reputation; at that moment it will be able to attack even robots with *good* reputation. Those may not be able to recover, if they only meet robots with reputation lower then their own, even if that reputation may be good enough compared with the mean of all population.

$\mathcal{T}$ Behavior

In this mechanism, seller reputation is compared to a reference reputation value; only sellers with reputation higher than the threshold are considered. That derives from an heuristic to aggregate the value of reputation of the whole population, and compare that to the individual performance, setting a value below which the seller is considered untrustworthy. Equation (2.15) and Table 2.4 show the information combination method and the parameters used for this behavior

$$combine\_info()\ if\ R_{seller} \geq c_s A(R) \tag{2.15}$$

	Aggregation method $A()$	Reputation metric R	Scaling c_s
$\mathcal{T}_a$	$mean\|_{swarm}$	reward, total wealth	[.5, .8]
$\mathcal{T}_M$	$max\|_{swarm}$	reward, total wealth	[.3, .5]
$\mathcal{T}_m$	$min\|_{swarm}$	reward, total wealth	[2, 3]

Table 2.4: Parameters and values range for $\mathcal{T}$ behavior

I test three different methods to compute the threshold, and use a specific real scale factor in the range $[0, 3]$.

Selecting parameters that are able to obtain a correct separation for all the market conditions is difficult. Despite that, the threshold approach can work very well when the separation between Byzantines and honest is very net, like in the debit market presented

in Section 2.4.3, or where *good* and *bad* honest have similar performances. Moreover, this requires to read all the wealth from the blockchain, for each computation.

$\mathcal{R}$ Behavior

In this behavior, the buyer reads the values of reputation of the entire population and ranks them in an ordered list; it will use seller's information if the rank of the seller is above a certain value. This is equivalent to checking if the seller belongs to the highest ranks of wealthy robots. Equation (2.16) and Table 2.5 show the information combination method and the parameters used for this behavior

$$combine\_info()\ if\ ranking(R_{seller}) \geq L \tag{2.16}$$

	Reputation metric R	**Rank Level L**
$\mathcal{R}$	reward, total wealth	top 80%, top 50%, top 30%,

Table 2.5: Parameters and values range for $\mathcal{R}$ behavior

The advantages offered by this system are a faster response: in fact using a threshold at a certain numeric value could delay the rejection of Byzantine robots; instead, if the market is already evolving in the manner pursued by the systemic protection, the robots will be already ordered by wealth in the intended way, and the separation will already be evident. Moreover, the selection of a numeric threshold may be difficult in certain markets, for example when new wealth is introduced in the system, or when the values of wealth for different classes are similar.

Similarly to the $\mathcal{T}$ behavior in Section 2.6.2, it requires to read all wealth values in the blockchain, for each transaction.

2.6.3. Staking based on Reputation

A faster evolution of wealth metrics for the different groups is advisable since the reputation mechanism is based on it. Since the difference between rewarded and staked amount for each transaction drives them, as shown in Section 2.3.2, I use the reputation to vary the amount each robot needs to stake, with the intent to accelerate the divergence between robots obtaining a positive and negative outcome from a contract.

For the robots with negative reputation I test a penalization scheme that imposes a

payment directly proportional to the absolute value of their reputation: the rationale is that a lower reputation will imply information of lower quality, hence the risk of buying it must be covered by an higher stake deposit. Moreover, this could represent the cost of significantly improving a robot's reputation.

While neutral reputation (Section 2.6.1) pays the base amount, some considerations must be taken into account about changing the stake amount for the robots that have positive reputation: rewarding the better robots could cause snowballing effect, making the rich even richer, causing a condensation of wealth. I test a variable penalization systems that saturates at the fixed base value for positive reputations.

The amount each robot must stake consists in the base stake amount $stake_0$ shown in Section 2.3.2, multiplied by a scaling factor, as presented in Equation (2.18). As scaling factor I use a monotonically non-increasing functions of reputation defined by parts, represented by Equation (2.17) and shown in Figure 2.3

$$c_s(R_i^t) = \begin{cases} 0.5^{\frac{2R_i^t}{10}}, & if\ R_i^t < 0\ or\ rewarding\ R_i^t > 0 \\ 1, & otherwise \end{cases} \tag{2.17}$$

$$stake_i^t(R_i^t) = c_s(R_i^t)\ stake_0 \tag{2.18}$$

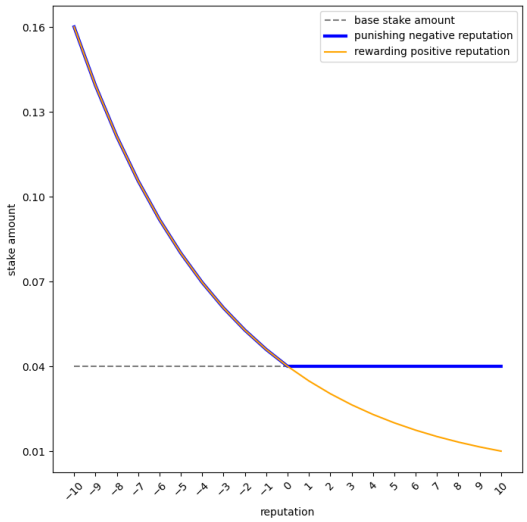


Figure 2.3: Variable stake amount based on reputation, punishing negative reputation, with and without reward for positive reputation.

The other variable reputation coefficient, awarding positive reputation with a proportional and smooth decrease in the amount to invest, has been tested, but left outside of this work.

2.7. Opening to Newcomers

My interest is to migrate the discussed protocols to an open system. When new robots join to the system after some time, it must be considered that the robots already participating have built a reputation, that could also be non neutral (Section 2.6.1), but positive or negative. It may not be possible to offer the newcomers the same conditions that the robots had at the beginning of the simulation.

In my analysis, I introduce three new robots in the environment. To simplify the test, I neglect the randomization of their spawn position, but instead I place them close to the nest, in a way that they are able to work as they are create. The reason for this is their low number with respect to the population. The most important aspect is the initial reputation they are introduced to the swarm with: since the identity of the robots is only ensured by their ID, which in the real case would be substituted with their blockchain address, if the initial reputation of newcomers is neutral, Byzantine may use this strategy to perpetrate the Sybil attacks shown in Section 1.3.1, creating a new identity to reset its reputation and hide its past behavior. There are multiple practical ways to deal with this, depending on the market conditions: in a foraging market, newcomers may be assigned negative reputation, for example the average of negative on, the lowest permitted or the lowest among robots; in this case they would be forced to take part of the physical task of swarm to recover it, but this practice would be unfair towards honest newcomers, especially in a market without foraging. However, I will not test this proposition in my work. I'm also interested in the possible instability effects caused by the introduction of new wealth in the system, brought by the new-coming agents.

I test the contemporary introduction of three Byzantine agents, with neutral reputation, to identify the possible instability on the market and prove that the reputation management system is not perturbed by that. The introduction happens after 20000 steps of the simulation.

3 | Results

In this chapter I summarise the characteristics I use to tell if wealth can be a good metric for reputation. I analyse if those are common in all the different robot behaviors that I have developed, to understand if they are inherent features of reputation based on wealth. Moreover, I want to understand if such protection system is compatible with the systemic protection based on staking.

The following characteristics are considered, and compared with the reference naive and sceptical behaviors of robots.

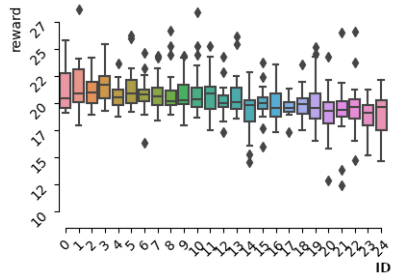
- how fast the robot can build the chain, and how long it resists to Byzantine attacks;
- the proportionality between the acceptance rates of the honest and their information quality;
- the proportionality between wealth of the honest and their information quality;
- the rejection rates of Byzantines;
- the robustness against an increasing number of Byzantines;

Moreover, I consider the same characteristics applied to an open population. Finally, I also address the wealth distribution and inequality. For a precise description and justification of these characteristics, the reader should refer to Sections 2.3.2, 2.4.1, 2.4.2, 2.4.3, 2.5, 2.6, and 2.7.

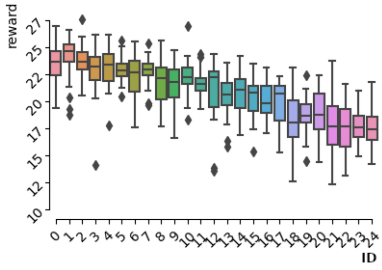
The results use the aggregation of a batch of 32 to 128 experiments, depending on the case, with the same conditions and the same odometry noise assigned to a robot between the experiments. To compare different batches, I introduce a fixed, integer, random number generator seed for the first experiment in the batch, incremented by 1 for the following. Instead of showing all the combinations of parameters for the behaviors, to ease the discussion I will present the most significant values.

3.1. Uniform Odometry Noise

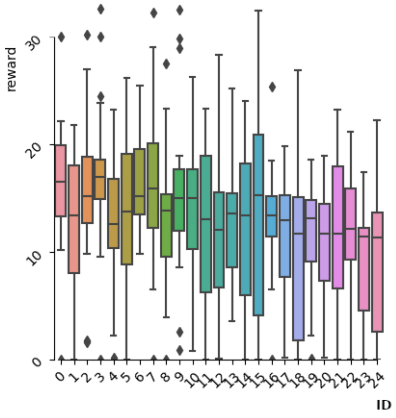
Firstly, I consider the dependence of reward W_i^t with respect to the odometry noise of the robots in a market without foraging reward, for the whole duration of the experiment of 15000 steps. No Byzantine is present in this experiment, hence I plot the robot IDs on the horizontal axis: the relative noise increases linearly, and it is similar to the one presented in Figure 2.2.



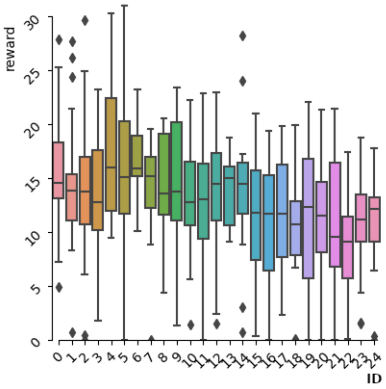
(a) R behavior, non-staking penalization.



(b) T behavior, non-staking penalization.



(c) R behavior, staking penalization.

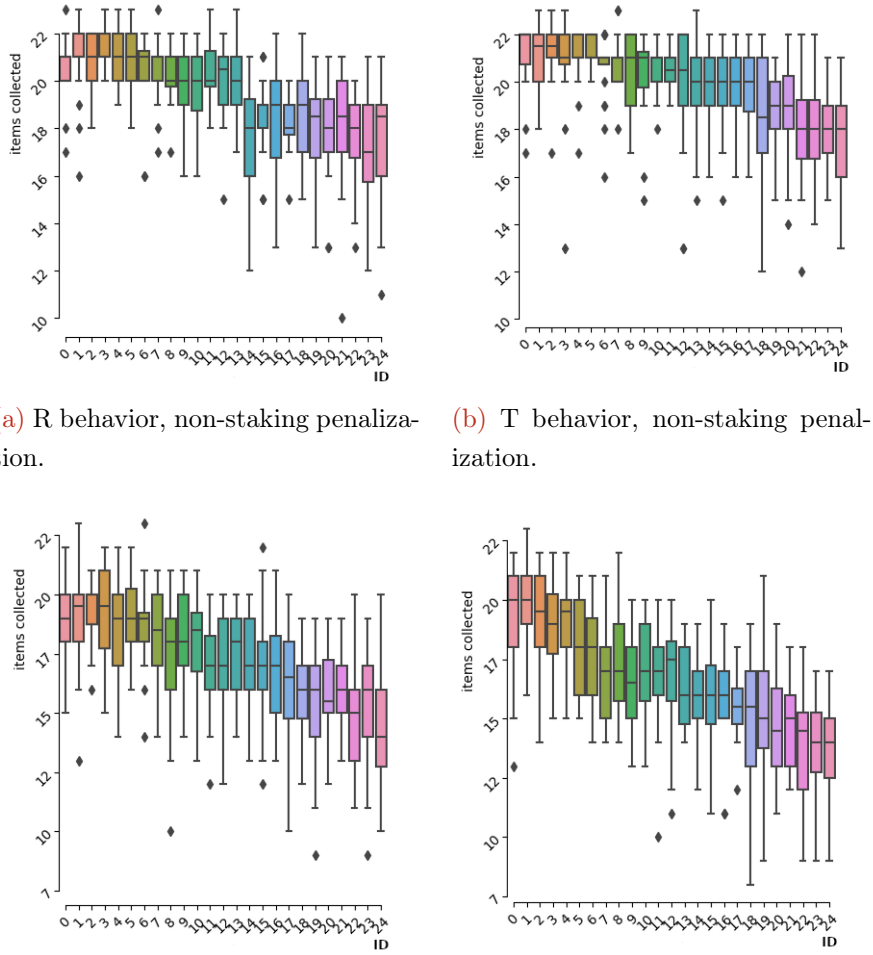


(d) T behavior, staking penalization.

Figure 3.1: Wealth distribution with respect to odometry noise in absence of Byzantines. The noise is linearly increasing with the robots IDs. All the behaviors use reward as reputation metric.

Figure 3.1 shows R and T behavior in a market without foraging, for a aggregated result of 128 experiments. Albeit the staking penalisation introduces high variance in the amount of wealth of each robot, due to the exclusion of staked wealth in this plot, wealth presents inverse proportionality with respect to odometry noise in both penalisation cases. The relation is not properly linear, but in the case it uses the non-stake penalisation presented in Section 2.5.2, it is very close to it.

I proceed to show the relation between foraging performance and odometry noise, for the same behaviors and market shown in Figure 3.1.



(a) R behavior, non-staking penalization.

(b) T behavior, non-staking penalization.

(c) R behavior, staking penalization.

(d) T behavior, staking penalization.

Figure 3.2: Collected items distribution with respect to odometry noise, in absence of Byzantines. The noise is linearly increasing with the robots IDs. All the behaviors use reward as reputation metric.

Distribution of collected items with respect to noise shows a similar dependence of wealth. In the case where no staking is considered in the outlier penalisation system, it is possible to see, for the robots with lesser noise, the saturation around 22 items per experiment on average: that is the "slack" physical limit for this combination of environment and setup. Interquartile range is similar for case (a) and (c), and (b) and (d) respectively, showing a similar dispersion for both penalization cases. Proportionality between collected items and odometry noise is close to a linear one.

Moreover, in the previous figures it is possible to observe a constant difference of collected items in the case with stake penalisation. This is due to a slower dynamics causing a delay in performances: the amount of invested wealth at the beginning of the experiment is consistent and has an impact of investing more. Once the chain is built, investments are recovered and robots are able to invest in an efficient way, foraging at the same pace of the non-stake penalisation method, as shown in Figure 3.3

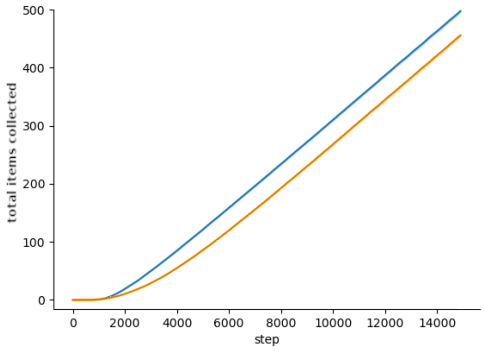


Figure 3.3: Foraging performance delay introduced by stacking, in absence of Byzantines: in orange, the system with stacking starts slower with respect to the one without, in blue. On the long run, gap stops increasing, since both penalisation methods let the swarm forage at the same pace.

For this reason, from this point onwards, I will present results referred only to the stable part of the experiment, usually the last 5000 steps; for such period of time, robots are expected to forage 8 items each. From this result, one may expect that the stake penalisation does not introduce any advantage in the system: in reality, staking improves wealth evolution and foraging performances of all honest robots in presence of Byzantines.

To summarize in this section how show that robots with higher reward have an history of good quality information. Moreover, the odometry noise directly influences the number of foraged items, with a proportionality very close to linear. Considering the two altogether, this could indicate that, using as reputation a function of wealth, this is directly related to the foraging performance.

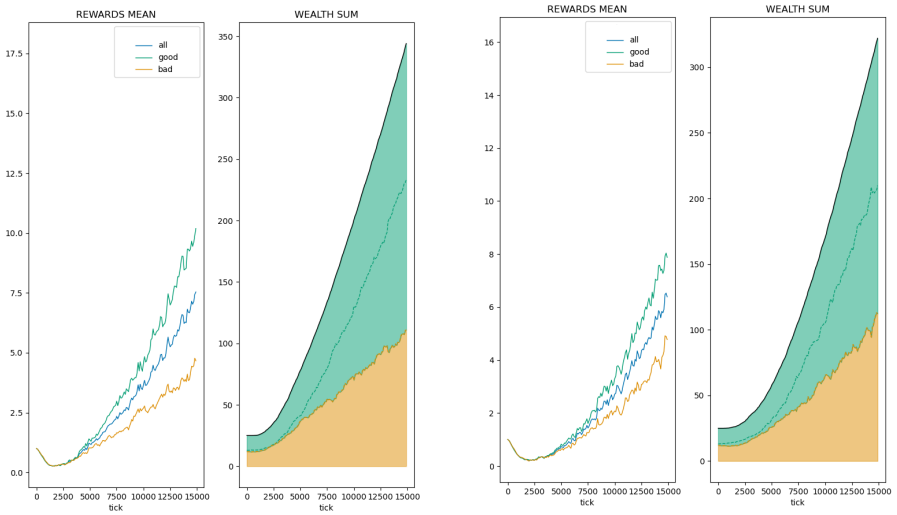
3.2. Market with Foraging

In this section I analyse the results using a market where foraging is rewarded, both in absence and presence of Byzantines; they lie on the vector's angle by increasing it of certain value. I test 90°, which is the theoretical worst value for this environment, and 25°, which instead has been proven to be the worst for the current implementation of the

sceptical behavior. I compare the results with it and with a naive strategy.

Firstly, I show the effect of the different wealth types, reward wealth W and total wealth $\mathscr{W}$, on the system performances. I focus on the differences they present when used with the behaviors and justify the assumptions I made to explain their use. To simplify the discussion, I only show the results of a single combination of parameters for a behavior, resulting from a batch of 32 experiments. Nevertheless, the effect of each wealth type used as reputation metric that I present is observable in other combinations of parameters. I will define for each different case to which extent those effect are observable.

In Figure 3.4 I present the effect on C behavior. On the left, it is visible that the effect of W is to increase in the difference between *good* and small honest groups. Albeit small, this effect is present in all experiments in absence of Byzantines and could helps in the differentiation of the honest. Moreover, an increment in the total wealth of the *good* group is visible.

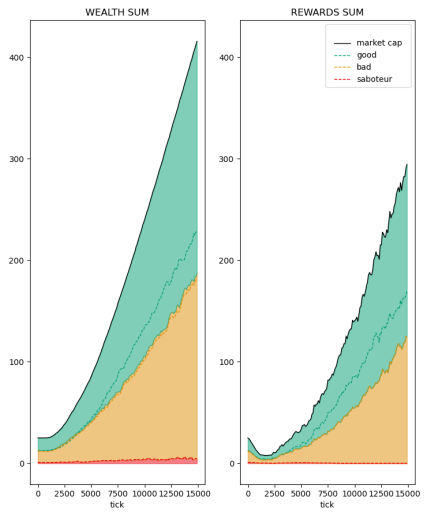


(a) C behavior, using reward reputation metric.

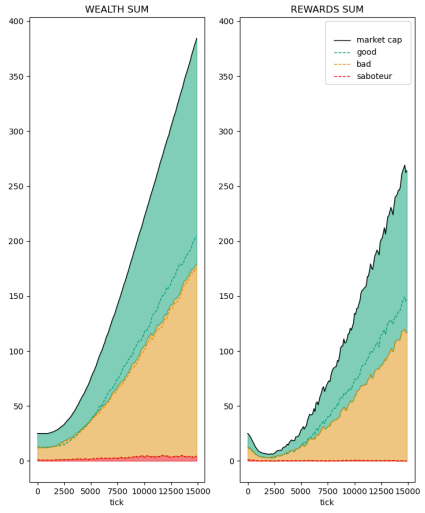
(b) C behavior, using total wealth reputation metric.

Figure 3.4: C behavior improves with the use of W as reputation metric. The effect is limited to the cases where no Byzantine is present, and tends to vanish when their number rises.

In Figure 3.5, similar effects are observable in the R behavior when the number of Byzantines is low: the cumulative wealth is improved and the difference between the two honest groups is more visible, and causes the mean value of wealth of the *bad* group to deviate more noticeably from the whole population's. Unlike the C behavior, the effect disappears rapidly when the number of Byzantines increases.

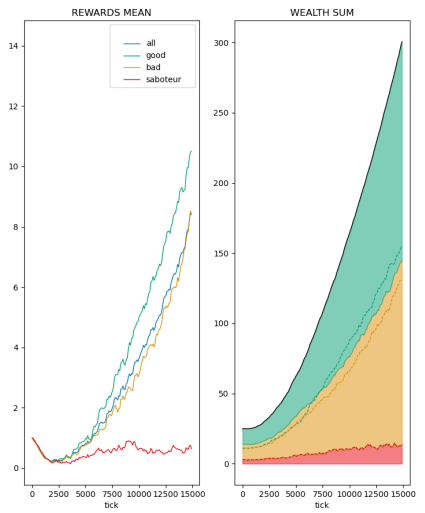


(a) R behavior, using reward reputation metric.

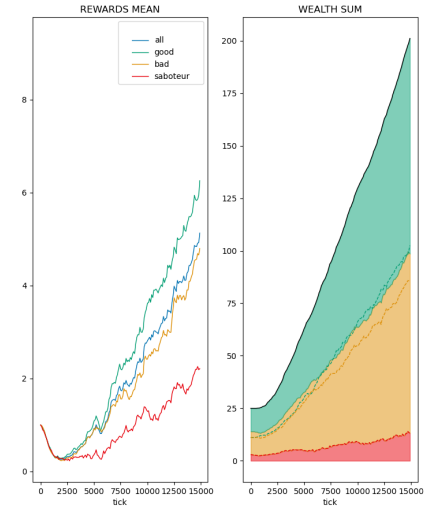


(b) R behavior, using total wealth reputation metric.

Figure 3.5: R behavior improves with the use of W as reputation metric. In this case the swarm is attacked by a single Byzantine, but the improvement rapidly vanishes with their number increasing.



(a) T behavior using reward reputation metric.



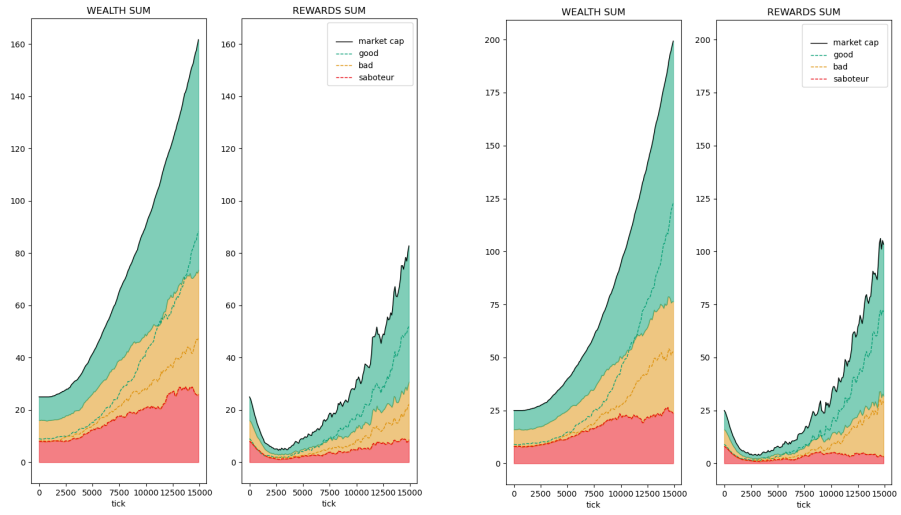
(b) T behavior using total wealth reputation metric.

Figure 3.6: T behavior using W as reputation metric is able to heavily penalise the Byzantines, despite this effect vanishes with their number. The *bad* group does not seem penalized, albeit its mean odometry noise is higher than the *good* one.

In Figure 3.6 the positive effect of W on the cumulated wealth is also present in T behavior,

despite being more limited than the previous cases. This reputation metric significantly impacts the mean available reward of the Byzantines; this advantage slightly diminishes with their number.

Using $\mathscr{W}$ as reputation metric in combination with $\mathcal{C}$ behavior shows good results when the number of Byzantines reaches $[5-8]$, displaying an higher robustness to Byzantine than in the cases using W. The cumulative wealth is considerably higher for the honest, with a marginal increase for the Byzantines, as shown in Figure 3.7



(a) C behavior, using reward reputation metric.

(b) C behavior, using total wealth reputation metric.

Figure 3.7: Honest using C behavior see a considerable improvement when using $\mathscr{W}$ as a reputation metric, while the Byzantines are not able to take advantage of it since their individual wealth is lower.

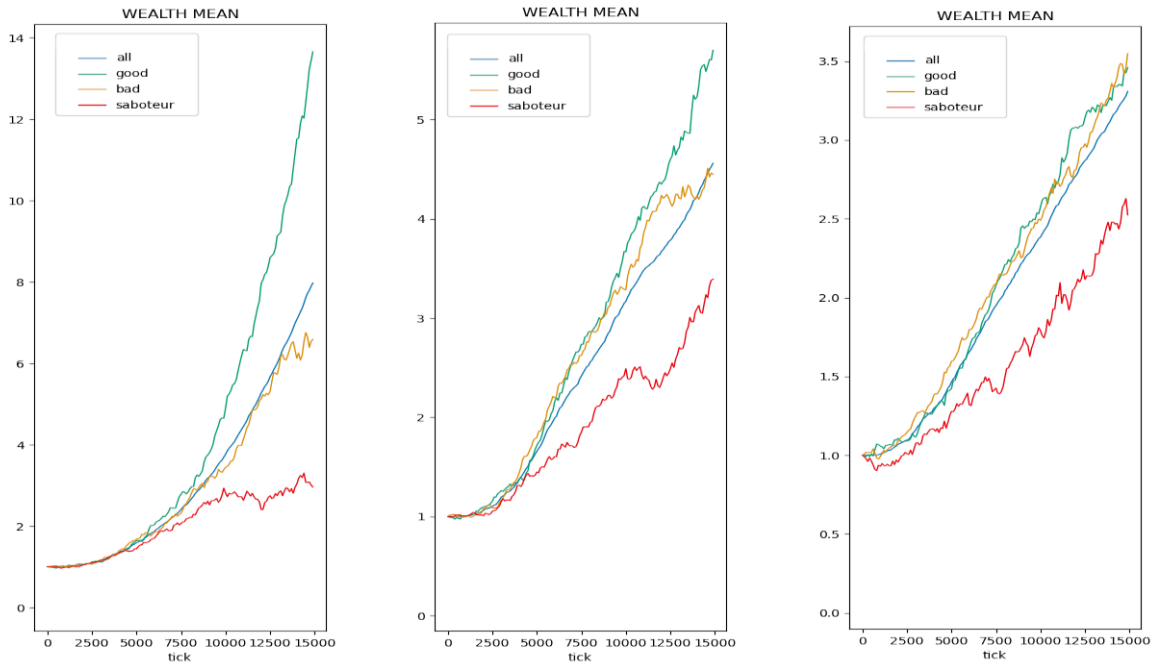
A deeper look should be taken to understand if the acceptance rate increases for all groups, of just for the honest.

With all this considered, it appears that reward yields higher returns when used as a reputation metric, not only for the *good* group, but also for the *bad* one. It does not appear to keep the staked amount frozen for a longer time, since there is no substantial difference between the evolution of wealth of groups, except in the case of the T behavior. When I defined $\mathscr{W}$ as a reputation metric, I hypnotized that it would advantage robots that have a long history of cooperation and good information, but that does not appear to be the general case. Nonetheless, that effect is visible in the $\mathcal{C}$ behavior.

Considering all the previous figures, another important element that it possible to observe

is that the moment where the greatest difference between the two different reputation metrics, $\mathscr{W}^t - W^t$, coinciding with the moment where the curve of W has a minimum, is also the moment in which the swarm is able to build a dynamic chain. Each behavior is able to build a chain in the same interval of $[1500 - 2500]$ ticks. As stated before, this metric can be used, alongside with the foraging performance, to asses the integrity and stability of the chain.

It appears clear that each behaviors respond differently to the system conditions, and each one has cases where it performs better than the other. My focus now is to show the different behaviors in more detail, and compare their performance, with each other and with the reference behaviors.



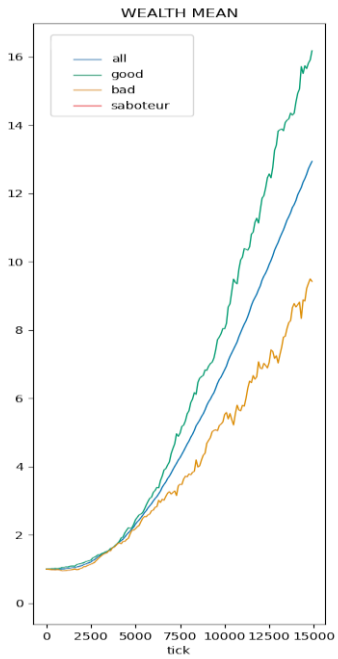
(a) C, 8 *average* Byzantines, total wealth metric.

(b) R, *average* Byzantines, reward wealth metric.

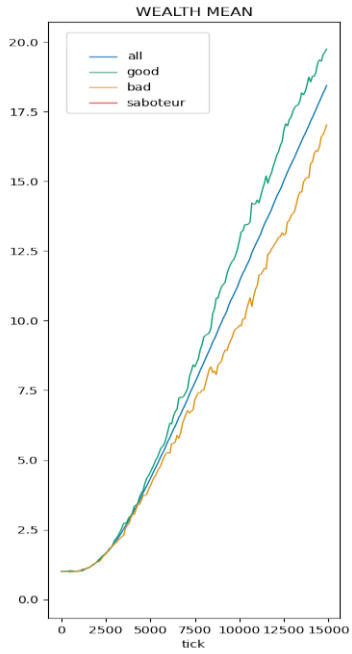
(c) C, 3 *perfect* Byzantines, total wealth metric.

Figure 3.8: Comparison with different number of Byzantines and reputation metrics

C behavior is able to impose an high degree of penalization on the Byzantine agents even in the case where 8 of them are present, as shown in Figure 3.8. Figure 3.9 shows the results when no dishonest robot is present, it is also able to penalize the wealth of the *bad* group, with respect of the *good* one, showing a great spread between the their evolution, but this is reduced using total wealth as a reputation metric. Comparing it to the R behavior shows that the degree of penalization is also higher for the *good* group, with respect to the same one in the latter strategy. The effectiveness against the byzantines



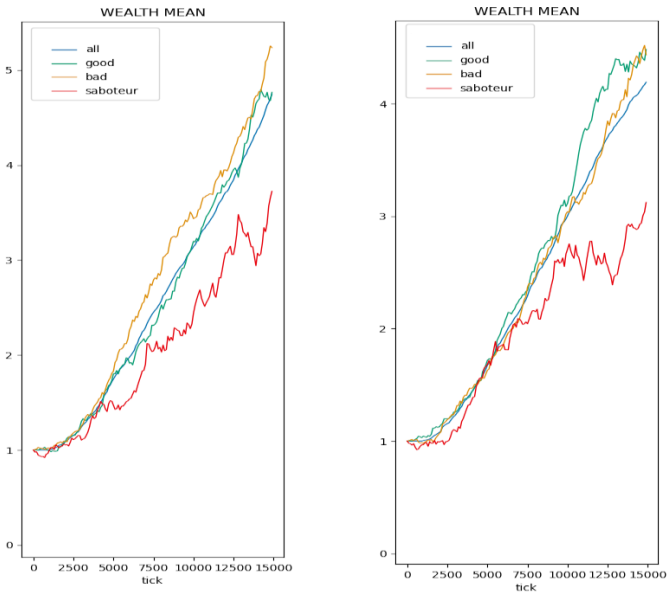
(a) C 25 honest, total wealth metric.



(b) R 25 honest, reward wealth metric.

Figure 3.9: Comparison with no Byzantines and different reputation metrics

comes at a cost of an higher rejection against both the honest groups, as shown with the acceptance rates in the Figures 3.16, 3.16, 3.13, and 3.14, resulting in globally lower wealth for all the robots.



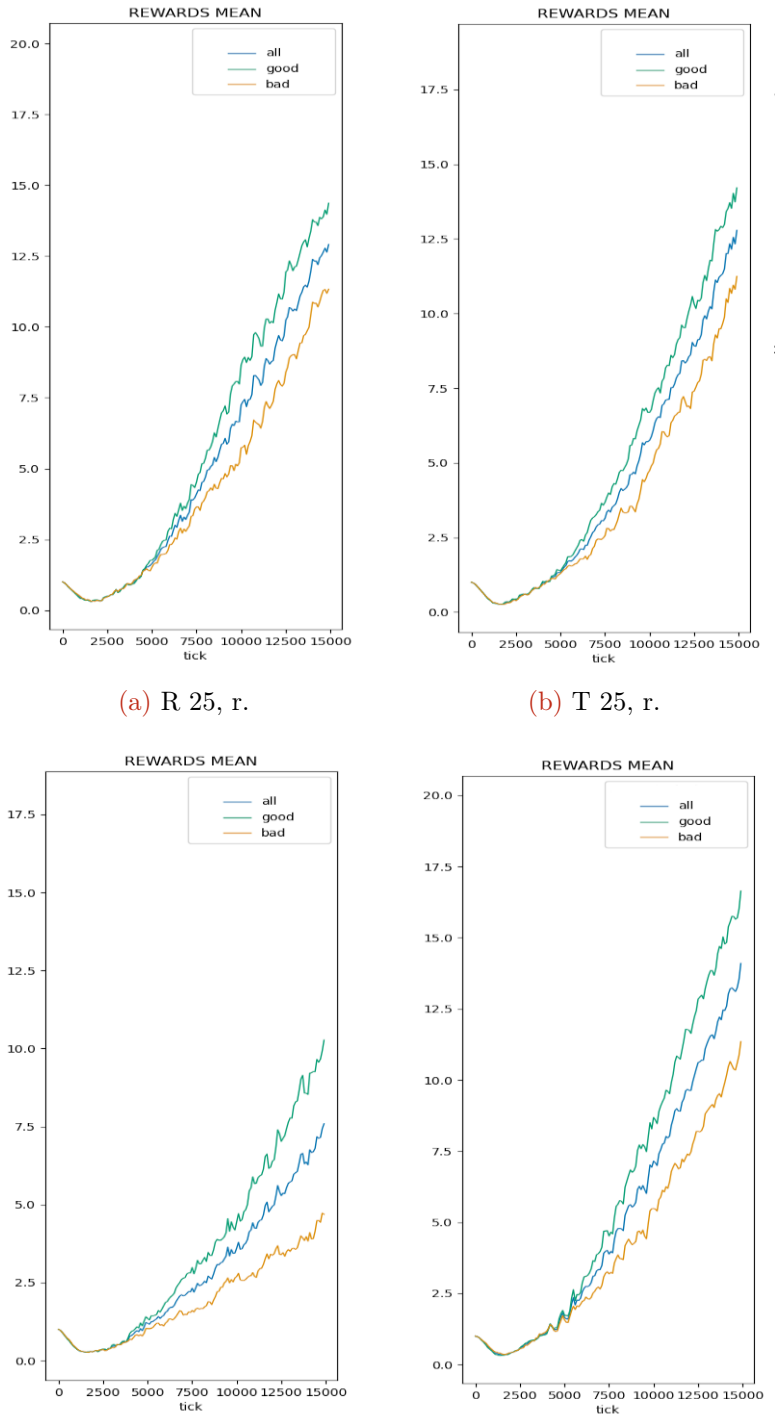
(a) T, 8 *perfect* Byzantines, r.

(b) T 8 *perfect* Byzantines, t.

Figure 3.10: Group Inversions for $\mathcal{T}$ Behavior

T Behavior presents group inversion, namely the *bad* group performs better, with respect of wealth, than the good one, as shown in as shown in Figure 3.10. The group inversion is less visible using the available wealth reputation metric, but still harmful. Since the group that has more noisy information is able to sell more, this impacts the performances of the swarm. When no Byzantine robot is present, the behavior does not present the inversion, with a beneficial impact on foraging performances, as shown in the Figures 3.16, 3.16, 3.13, and 3.14.

Differently from my initial hypothesis, R behavior does not provide a much faster evolution of wealth metrics, compared to T and C Behaviors, as shown in as shown in Figures 3.11, where no Byzantine robot is present. Nonetheless, its evolution is comparable with the one of Sceptical Behavior. Group separation is present, but not as good as the one of C or S Behaviors.



(a) R 25, r.

(b) T 25, r.

(c) C 25, r.

(d) S 25.

Figure 3.11: Speed at which different Behaviors build the chain is not different

The simultaneous analysis of foraging performances, shown in the left plot in the Figures 3.16, 3.16, 3.13, and 3.14, and acceptance rates, alongside mean reward, shown on the right, is able to shed more light on how the strategies behave and what is effectively

their performance for what it concerns the foraging task. I only show the best combination of parameters, for both wealth reputation metrics.

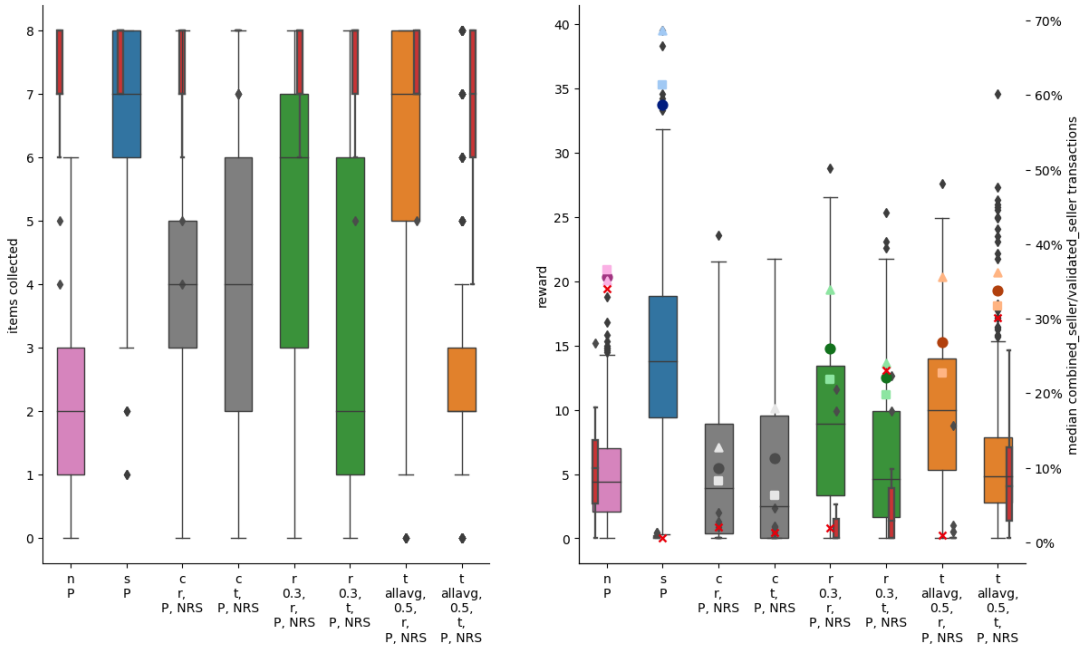


Figure 3.12: Foraging FIM, 1 *perfect* Byzantine

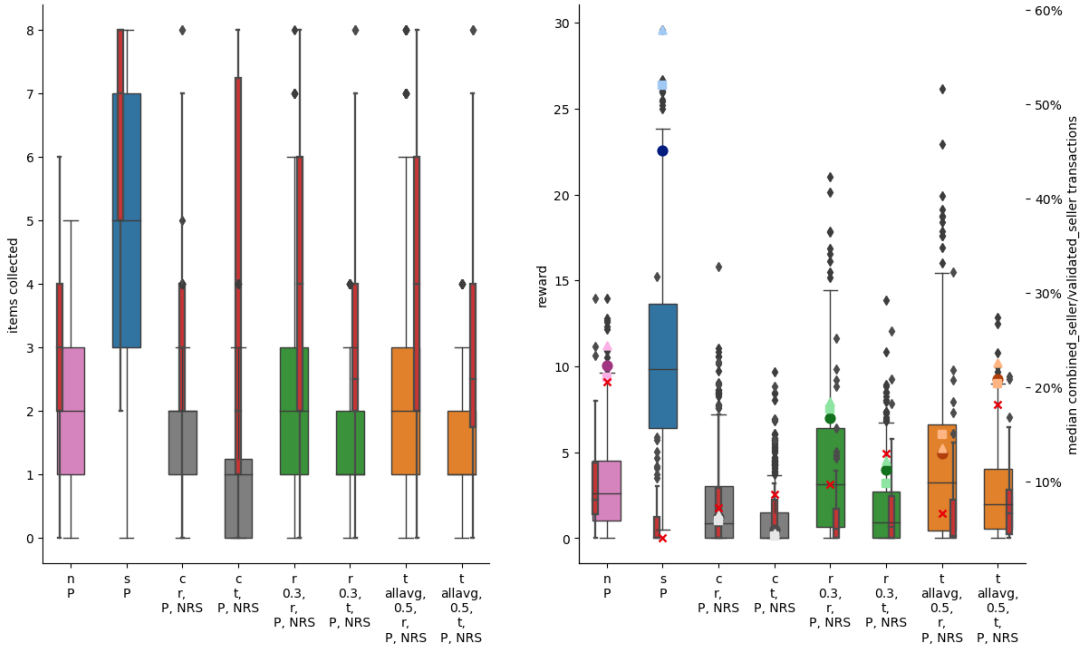


Figure 3.13: Foraging FIM, 3 *perfect* Byzantines

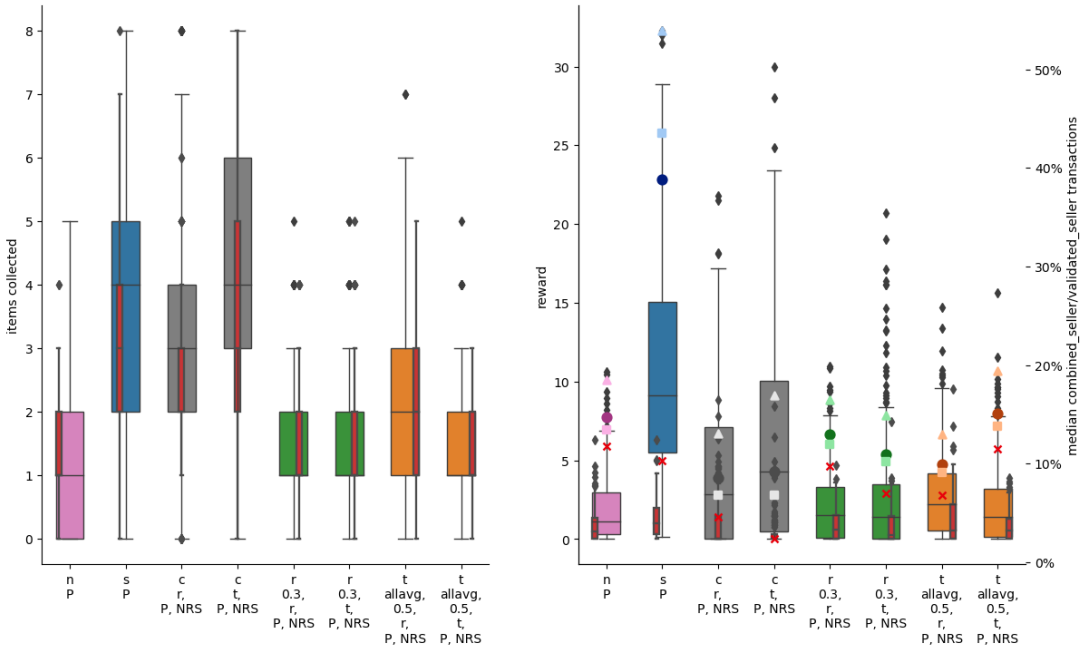


Figure 3.14: Foraging FIM, 8 *avegage* Byzantines

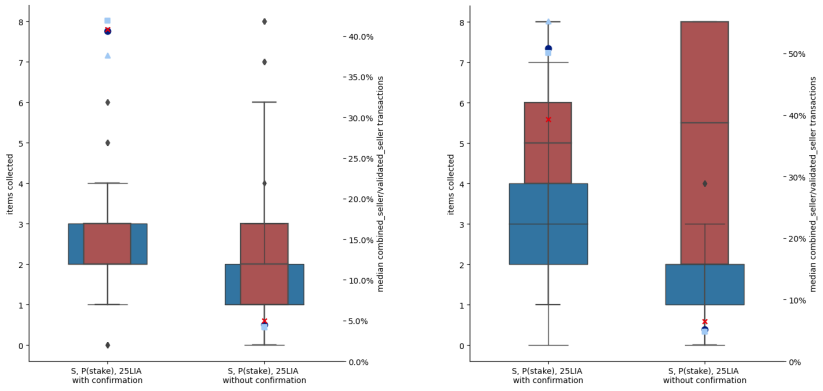
In the previous figures, it is possible to see how acceptance rates of the groups relate with the foraging performances of the groups. An higher rate for the Byzantines, represented by the a red X symbol in the right plot, causes a massive drop in foraging performances and rates for the honest, but also a decrease of foraging ability of the former. The huge increment in the foraging performances of the Byzantines, like in Figure 3.13, is in fact limited to the situation when they have $perfect$ odometry noise, and are hence able to navigate without relying on external information too much.

Considering individually C Behavior shows how its generally lower, but steady, value of wealth metrics reflect of the foraging performances, with a lower, but steady, number of collected items and acceptance rates. It is not robust against Byzantines with $perfect$ odometry noise, but very robust against their increasing number, since it is able to penalize each group consistently, either honest or Byzantines. Anyway with this performances, it is not very useful in this scenario.

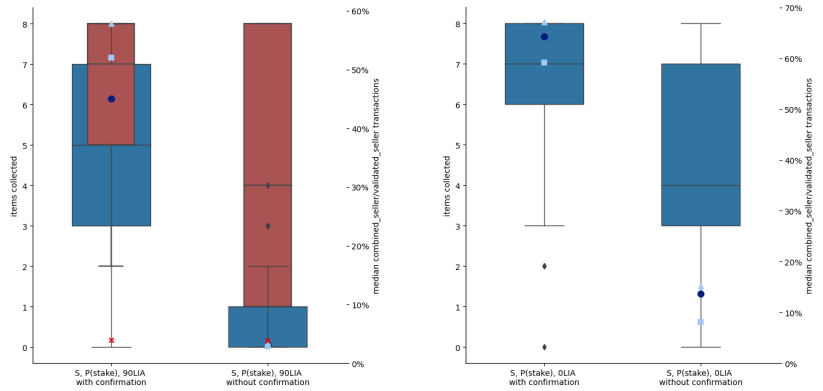
Instead T and R Behaviors are able to forage better than the Naive Strategy almost in all the conditions. The former has basically the same acceptance rate for all groups, resulting in catastrophic effects even for a single Byzantine. T and R instead, are able to maintain a low acceptance rate for the Byzantines when their attack is not so strong, but they both fail against the Sceptical behavior, which is able to maintain a huge separation between the acceptance of honest and Byzantines, even in the case where their number raises or the quality of their odometry noise is $perfect$, and even if the acceptance rate of the

dishonest is comparable to the one of T and R Behaviors.

Its strength comes from its dual individual protection mechanism, that also considers the confirmation of messages rejected at first, but later confirmed by subsequent ones. Figure 3.15 shows how confirmation mechanism improves the foraging performances of Sceptical robots, both improving the honest and undermining the Byzantines: since the Byzantines are not confirmed, they cannot collude.



(a) S 8 *perfect* Byzantines, 25 lie. (b) S 3 *perfect* Byzantines, 25 lie.



(c) S 3 *perfect* Byzantines, 90 lie. (d) S 25 lie.

Figure 3.15: Foraging FIM confirmation s

The "cost of protection" for the Sceptical Behavior is shown in Figure 3.16, comparing the collected items of the former with the Naive Strategy, in absence of Byzantines. Since the *bad* robots, which in this have the worst information, compared to the one of the *good* group, are confirmed by other members of the same group, they are able to collude, causing a minor, but noticeable decrees in performances. T and R Behaviors are able to forage at the same efficiency of the Naive robots, because they do not need the local confirmation of the information, but rely on a global infrastructure to decide which sources

of information they must reject. Analysis of the acceptance rates of the honest show that the strategies that are more robust against the collusion phenomenon are the ones that have a higher relative difference between the one of the *good* the one of the *bad* groups, despite the former being smaller, in absolute value, with respect to the same group in the Sceptical Behavior. In Figure 3.16 this is clearly visible; the *good* group is represented by a triangle of a lighter shade with respect to the relative behavior, while the *bad* one is represented by a square of the same shade. The darker circle is the mean between the whole population.

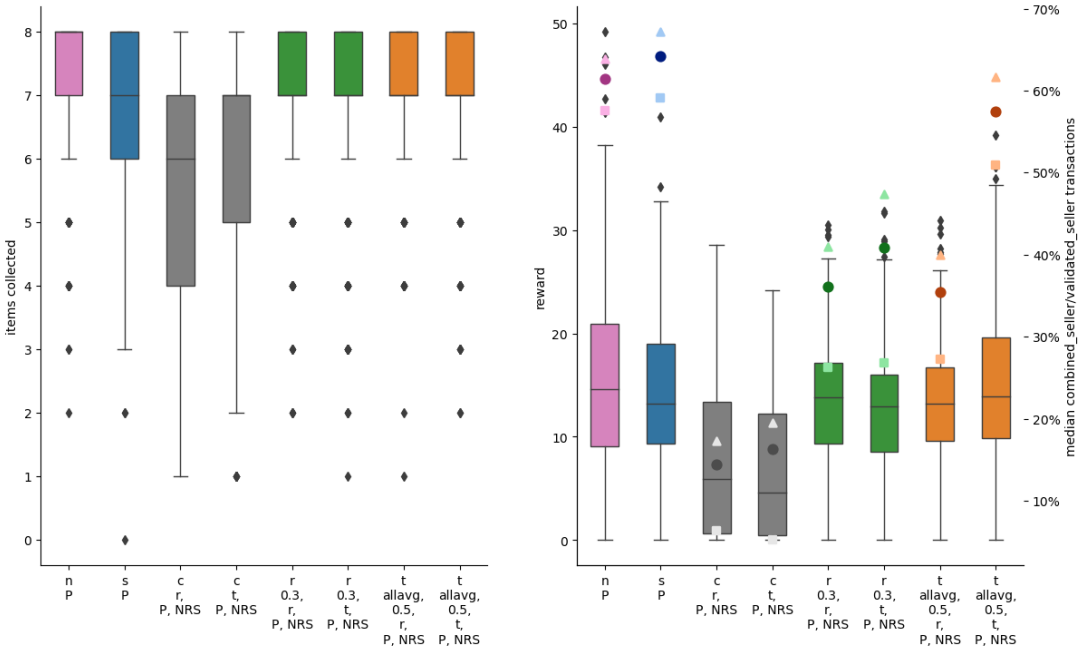


Figure 3.16: Foraging FIM 25

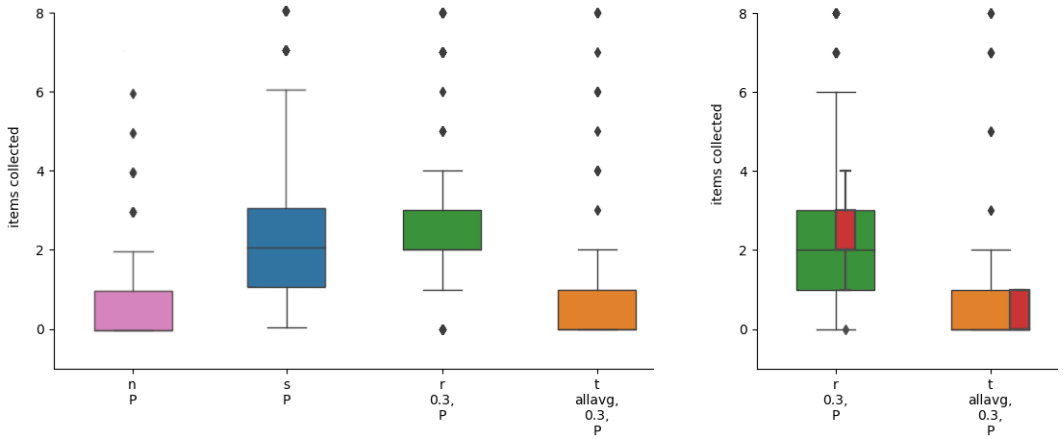
To summarize, wealth proves its ability to function as a metric for a global reputation management system, but suffers in presence of Byzantine attacks.

3.3. Information Market

My interest is to understand if it is possible to improve the performances of the behaviors using reputation by changing the market conditions, instead of focusing only on the tuning of their parameters. In this section, I discuss the results obtained using a market that does not reward foraging, and the introductions of the possibility to create debit. The conditions of the experiments are identical to the ones reported in Section 3.2

I show the effect of preventing the robots to obtain the information for free, in the case

that they do not have enough wealth to pay the creditors when a contract closes. The test is conducted at the moment of the creation of the contract, and does not account for the wealth the buyer could obtain between that moment and the finalisation of the contract.

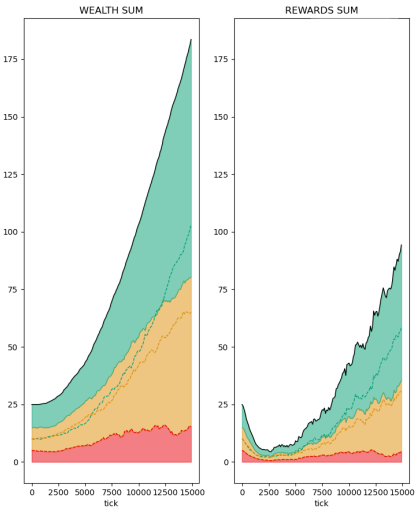


(a) No Byzantines. (b) Three *average* Byzantines.

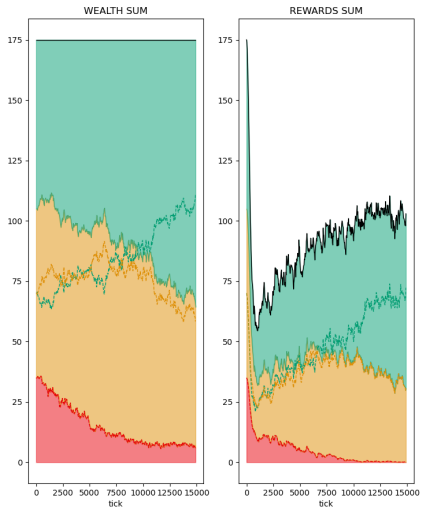
Figure 3.17: Foraging performance with a test on buyer's wealth before giving to it information for 25 robots and 32 experiments. In (a) naive, sceptical, R and T behaviors, represented by magenta, blue, green and orange color respectively; in (b), only R and T.

In Figure 3.17 it is possible to see that the result of this test is a strong reduction of performances for all robots, honest or Byzantines; preventing the spreading of the honest information free of charge imposes a great cost to the collectivity, even in absence of Byzantines. The cost payed by the robot selling information is completely justified.

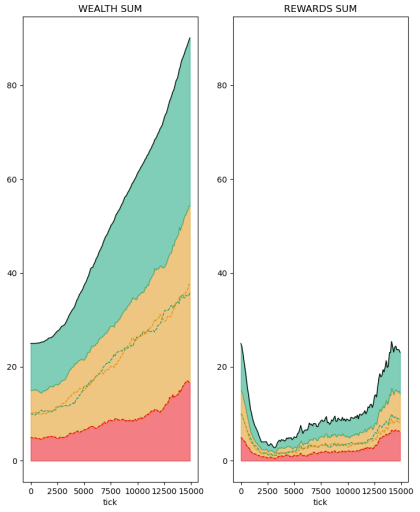
Now, I focus on the results using a simple information market, with staking penalization, but without the foraging reward or the introduction of debit. Byzantines are able to exploit the market discussed in Section 3.2 and recover wealth to satisfy staking requisites by foraging. In this different market type, the reward share of the creditors is payed entirely with debtor's wealth. The Figure 3.18 shows the different wealth evolution in the two market cases, for Byzantines and honest robots.



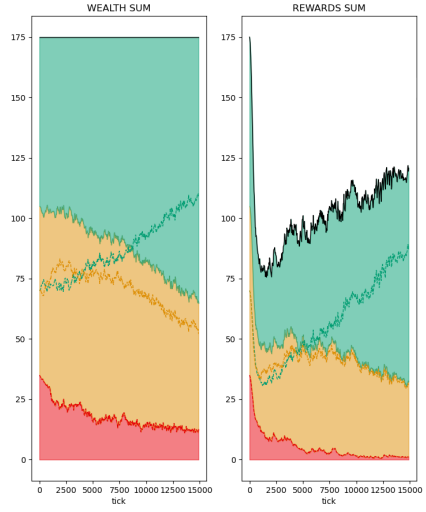
(a) Foraging market with five *average* Byzantines; C behavior using reward as reputation metric.



(b) Information market with five *average* Byzantines; C behavior using reward as reputation metric.



(c) Foraging market with five $perfect$ Byzantines; R behavior using total wealth as reputation metric.



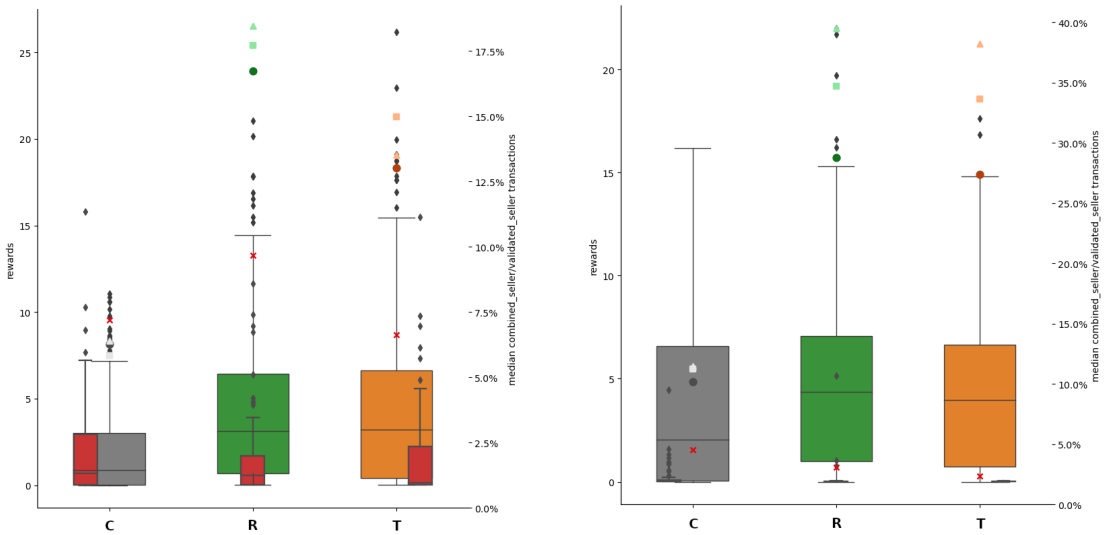
(d) Information market with five $perfect$ Byzantines; R behavior using total wealth as reputation metric.

Figure 3.18: Open foraging market in (a) and (c), closed information market in (b) and (d). Share of wealth of the *good* and the *bad* groups are green and orange respectively, Byzantines is red. Both wealth metrics for the latter are increasing in the open market case, while decreasing in the closed one.

Figure 3.18 clearly shows that the Byzantines are unable to increase their reward, when prevented from exploiting the foraging payment. Therefore, they never recover the initial investment conducted when the chain is not formed yet, and their wealth will be eroded

with time, until the will be prevented to use staking. Regarding the formation of the chain, it can be observed that the moment where the reward is minimum is localized around 1000 simulations steps, considerably before than the the moment this same event happens in the market with foraging, indicating that the dynamic chain is built at a faster speed.

Honest robots are not affected by the removed additional income, because they are able to exploit the Byzantines. This could also be observed considering the mean reward and the acceptance rates of the groups, as shown in as shown in Figure 3.19

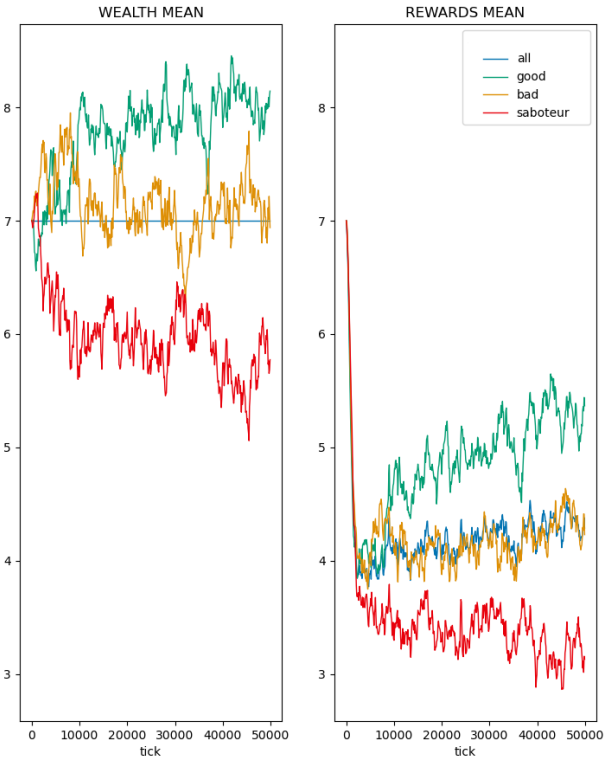


(a) Foraging market with three *perfect* Byzantines; behaviors using reward as reputation metric.

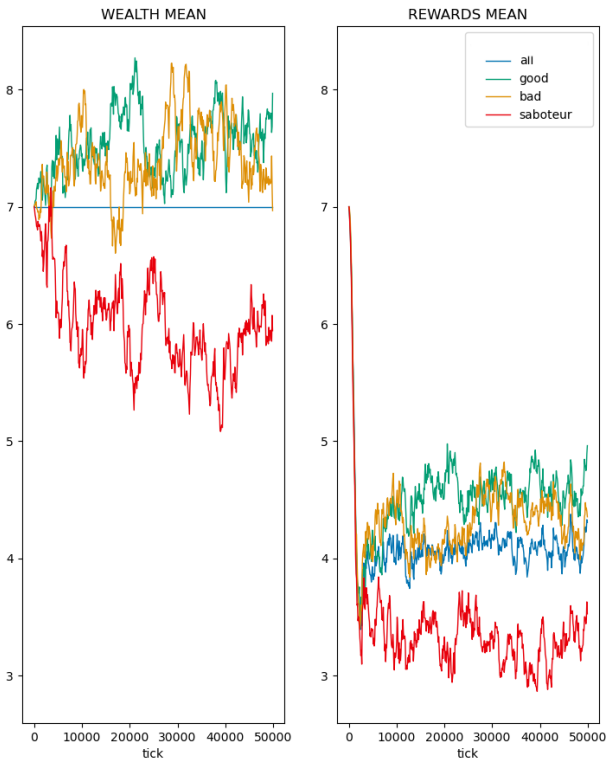
(b) Information market with three *perfect* Byzantines; behaviors using reward as reputation metric.

Figure 3.19: Honest do not suffer from the lack of foraging compensation: median reward, shown in grey, green, and orange color for C, R and T behavior respectively on the left vertical axis, is similar; they also benefit from increased acceptance rates for all the honest groups, with *good* represented by a triangle, *bad* by a square, in lighter colors. Byzantines, on the contrary, almost see their reward vanishing: the red bar is barely visible in the information market. Moreover, their acceptance rates, represented by the red cross, plummets. Darker circle represents the population's median.

Now, I analyze the results of the best combinations for the behaviors and the reputation metrics. In this series of experiments I also added a second value for the angle that the Byzantine add to the counterfeit messages, 25°, because the Sceptical robots using outlier penalization with staking are particularly susceptible to it, for the particular values selected for the design parameters.



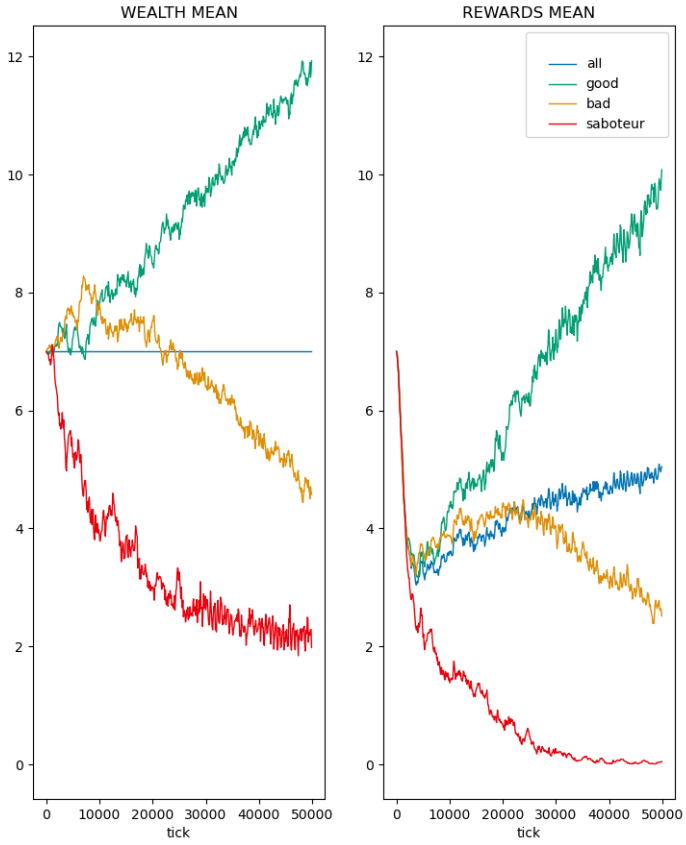
(a) R Behavior.



(b) T Behavior.

Figure 3.20: 8 *average* Byzantines, t deals better with 25 degrees lie

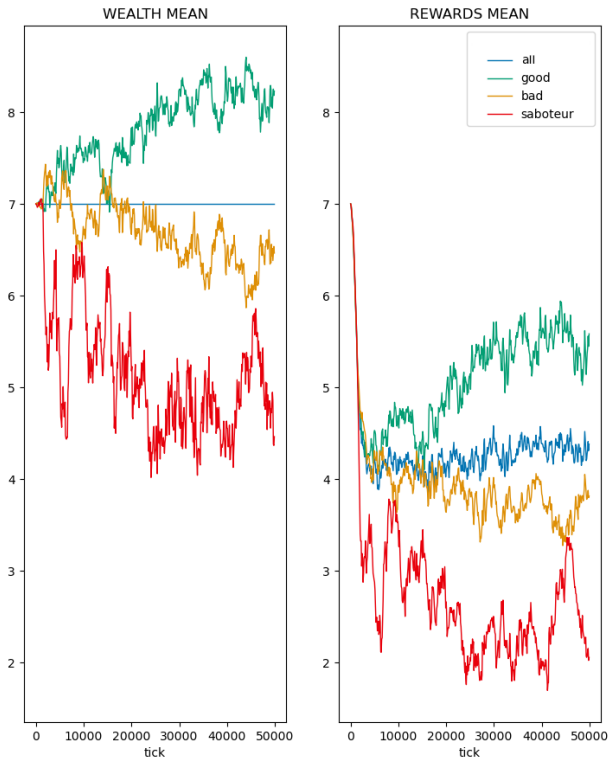
Figure 3.20 shows that R Behavior using W available wealth metric obtains a clear separation of the groups, even in the case of 8 *average* Byzantines, lying for 25°; as I initially hypothesized, using this strategy helps where the differentiation of groups is not so easy, such as in the context of the market that does not introduce new wealth as foraging payment. Figure 3.21 shows that R behavior is still robust when the Byzantines are $perfect$. However, group inversion phenomenon appears, even tho it is not harming the performances.



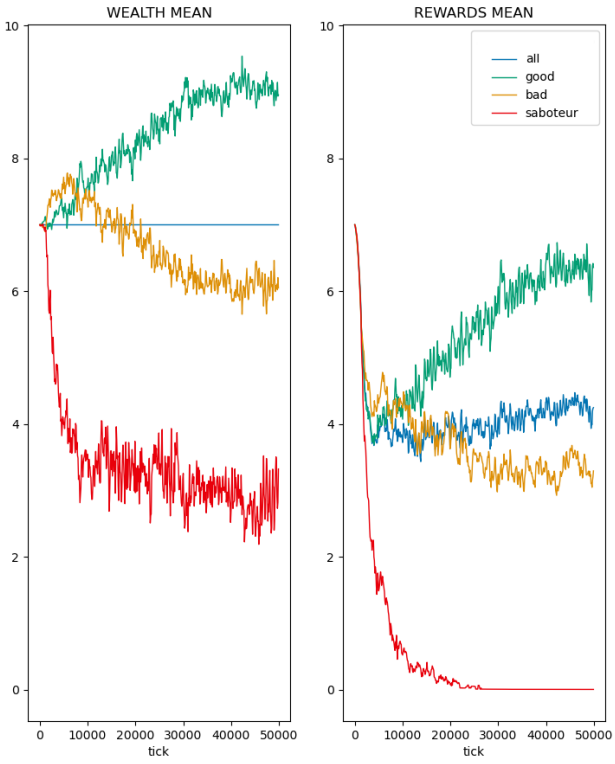
(a) 5 $perfect$ Byzantines.

Figure 3.21: very good separation of groups, for different perfect byzantines, using reward reputation method, total wealth method also obtains good results

T Behavior improves in this setup using W reputation metric, but its performance are weak compared are not comparable to the ones of R. The threshold mechanism is not very suited to operate in a situation where separation of groups is not easy. Moreover, Figure 3.22 also shows the effect of a 25° degrees angle.



(a) 25 degrees.

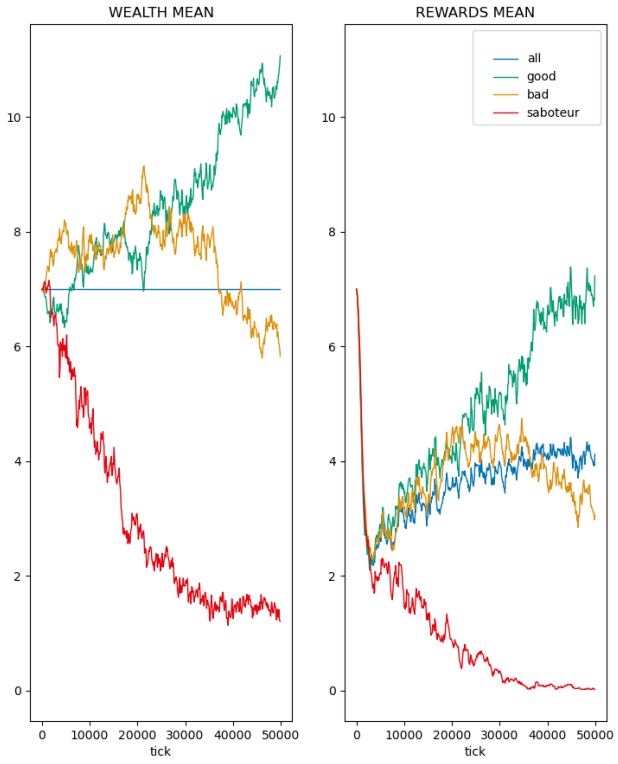


(b) 90 degrees.

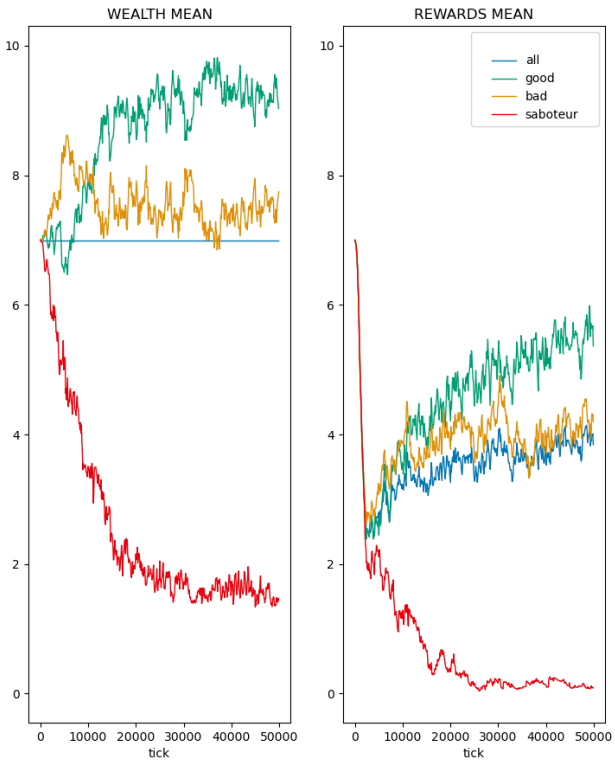
Figure 3.22: T does not suffer the different lie angle

T behavior does not use the same parameters as Sceptical one, but all strategies use the same method o compute the weights to update the information, which shares some parameters with the latter, pointing that that one as well also suffers from this particular value of Byzantine attack. Group inversion is also visible.

C Behavior, presents robustness against strong Byzantine attacks, as shown in Figure 3.23 in the case for 5 Byzantines. Reward reputation method is slower, but is able to separate the groups better, while total reputation is more consistent in performances. The fact that is robust also with an large number of Byzantines shows the intrinsic robustness of this strategy. Group inversion in also present here, indicating that it could be an issue widespread in this particular market. Despite that, as shown in the Figures 3.24, 3.24, 3.26, and 3.27, that does not harms performances.



(a) C, reward wealth.



(b) C, total wealth.

Figure 3.23: effect of repuation method on 90 degrees lie angle

I now compare how the differences between behaviors traduce in foraging performances. The following plots are identical to the one shown at the end of Section 3.2. Figure 3.24 shows the swarm in absence of Byzantines. All behaviors appear to forage a little worse to respect to the case of Figure 3.16, in this case where the foraging payment has been removed. The acceptance rate of the *bad* groups diminished and has a large gap with the *good* one, that arrives at peaks above 75%. This could indicate that the *bad* group could sometimes provide useful information, and a trade off between collusion and absence of information should be done. Moreover, since the resources of the market are limited, and staking requires a big part of them, the *bad* group could be partially victim of the same stacking mechanism.

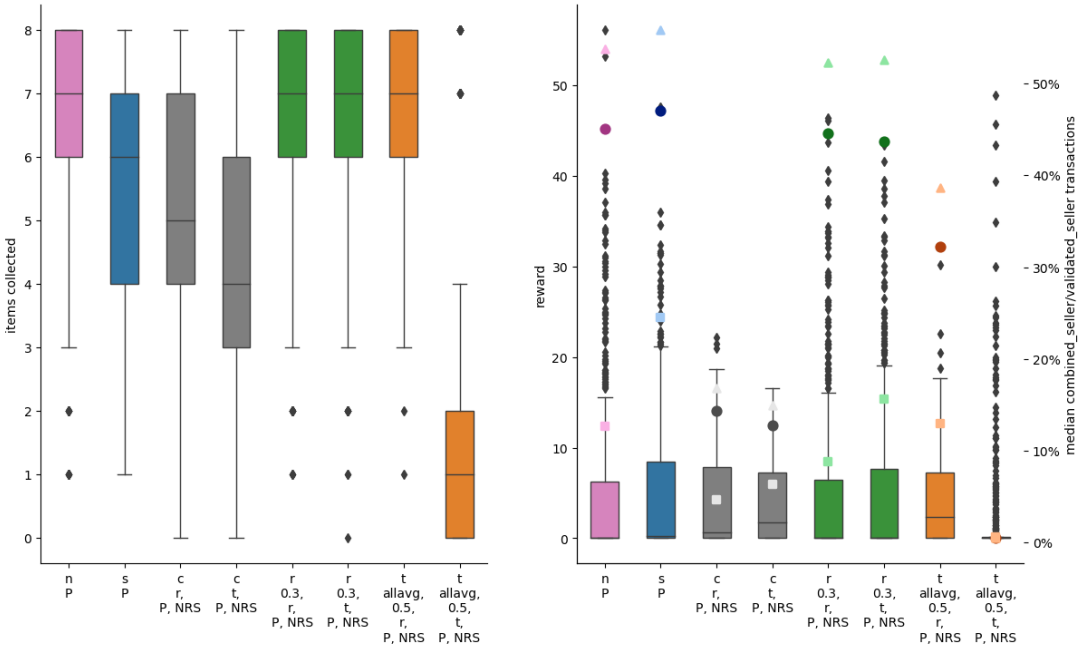


Figure 3.24: Foraging IM 25

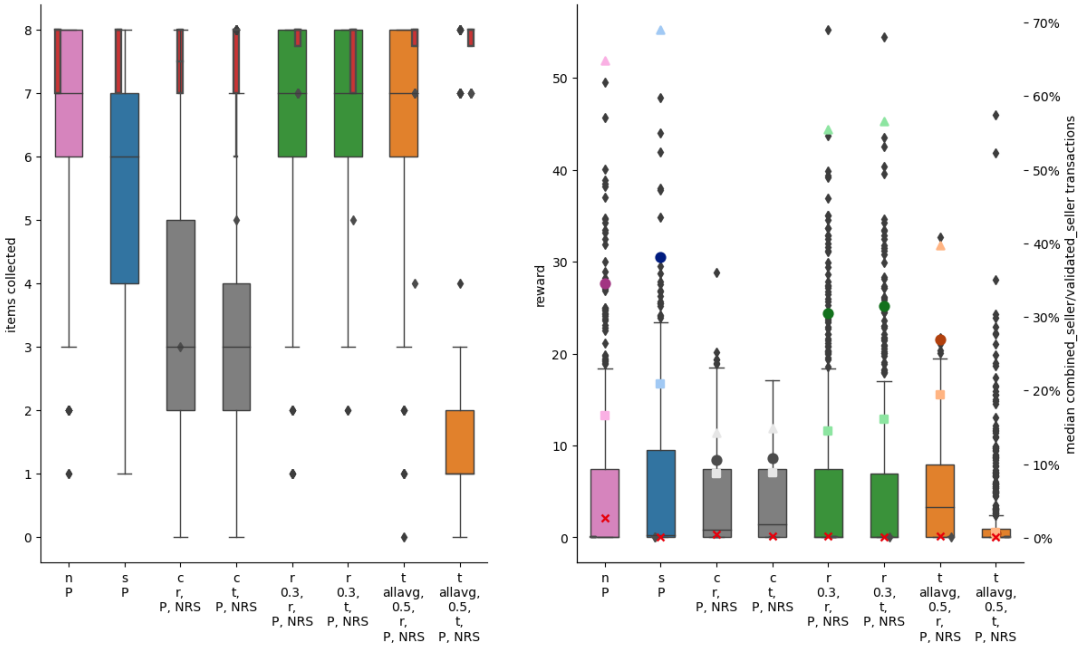


Figure 3.25: Foraging IM 24 1perf

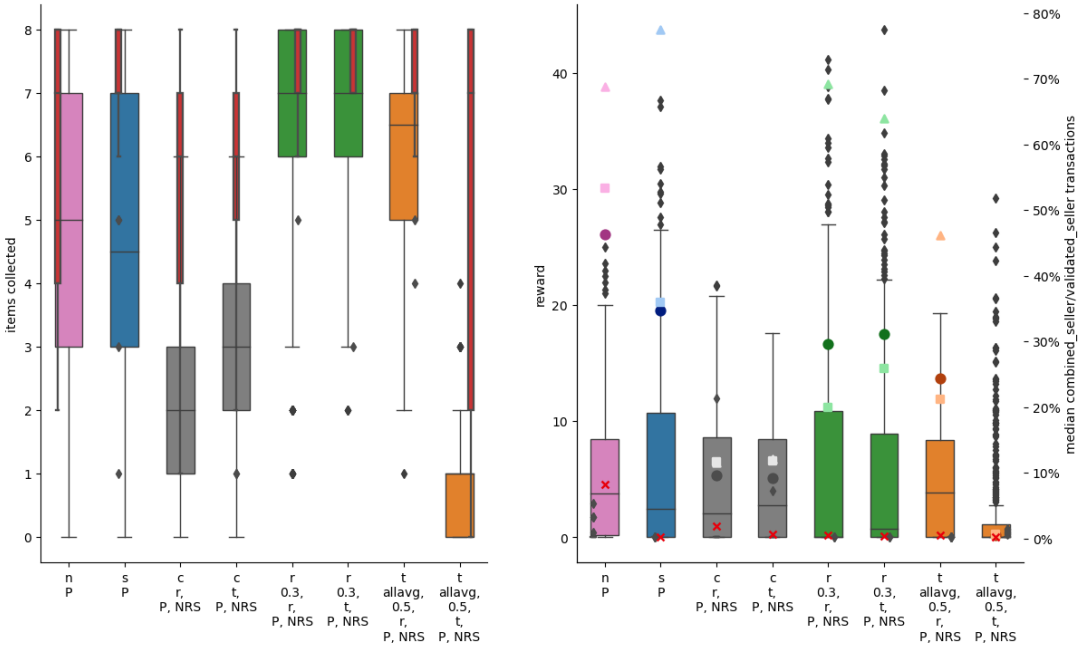


Figure 3.26: Foraging IM 22 3perf

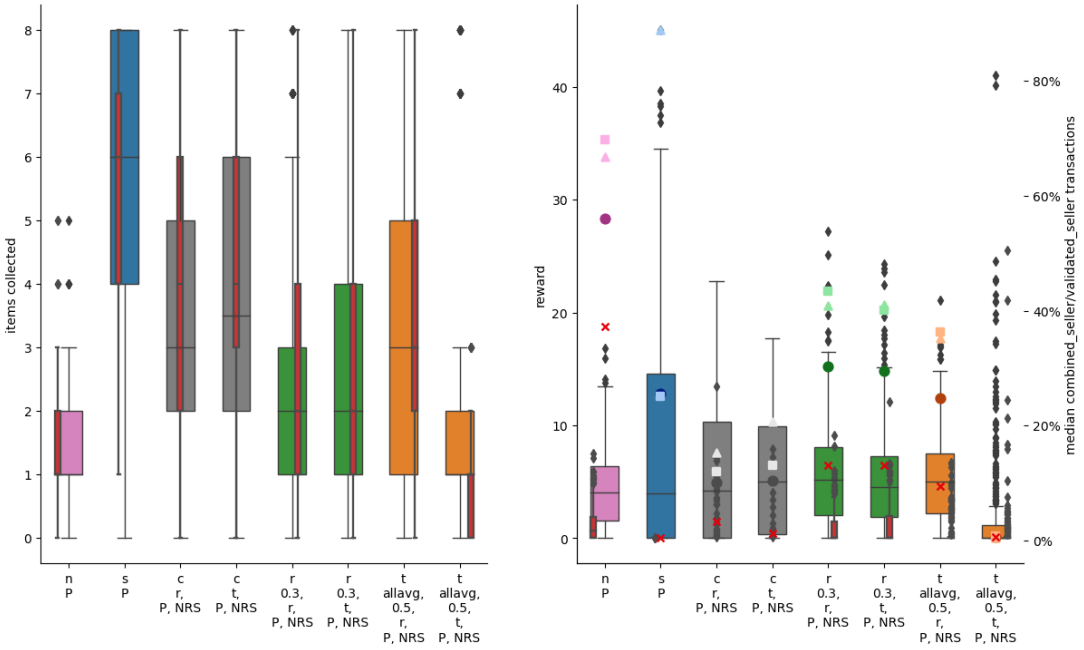


Figure 3.27: Foraging IM 17 8avg

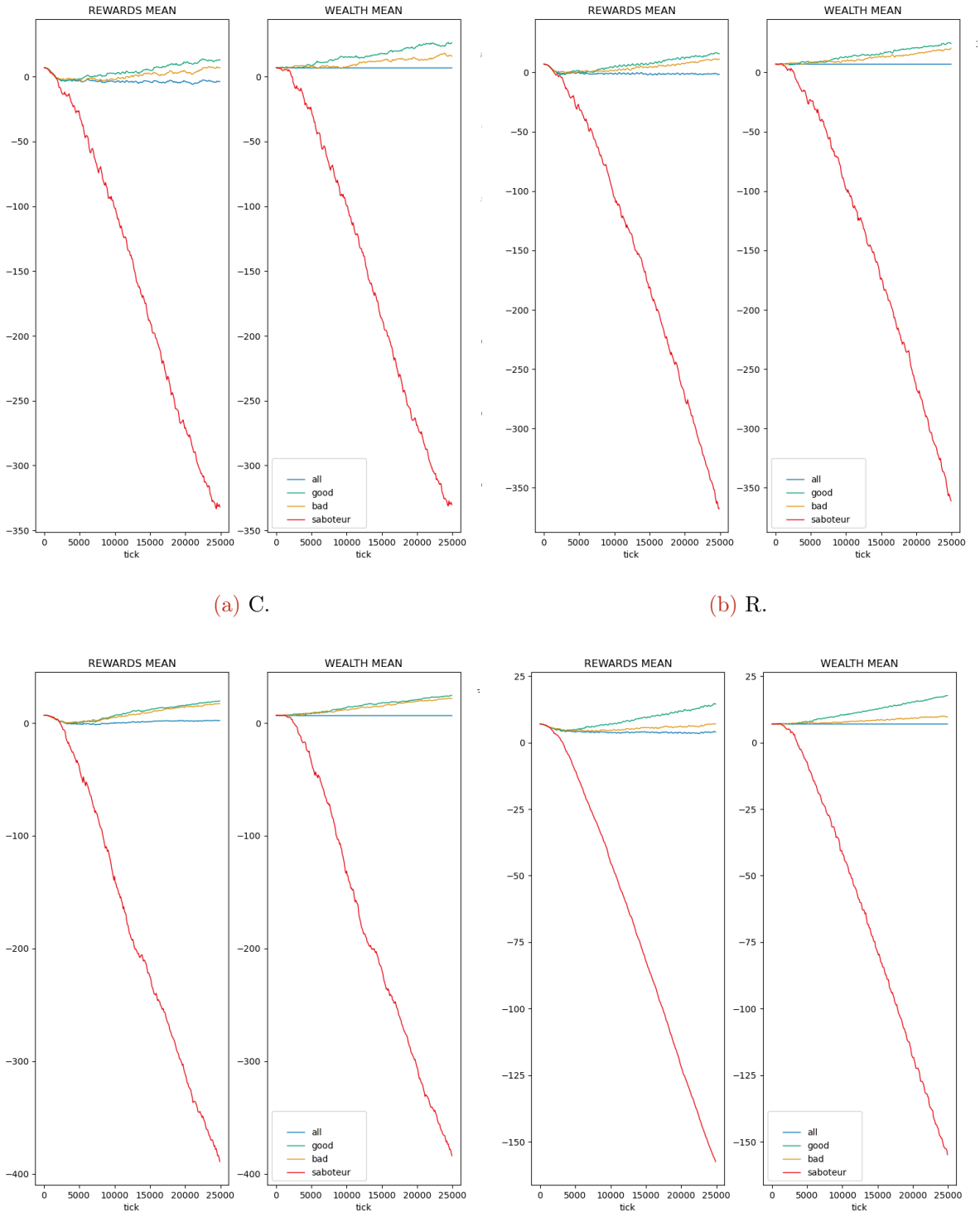
Figures 3.24, 3.26, and 3.27 show that the the absence of the reward for foraging harms the byzantine ability of accessing the market and spread bad information. The strategies using reputation are resistant up to a strong Byzantine attack, except for the T Behavior using total wealth as reputation, that, as already said, does not perform well with a small margin for separate the groups. C behavior instead confirms its steady performance, meaning that the strategy that it uses is decent. Sceptical Behavior is able to regain its advantage at high number of Byzantines.

To summarize, a market that does not reward the foraging is able to harm the Byzantines by removing wealth from their groups, to redistribute it to the honest ones. This comes to a small price to pay in the case of no Byzantine attack.

3.3.1. Information Market with Debit

Here, I show the results of the same setup described at Section 3.3, with the additional possibility for robots to make debit by borrowing wealth from the market. The debit is recorded as negative wealth. This comes from the fact that, due to the staking mechanism, the Byzantines have the innate tendency to squander their resources; hence, if a market continues to provide them with new wealth, they tend to separate themselves more visibly from the honest groups, while not being able make enough debit to improve their performances enough to counteract this effect.

Figures 3.29 and 3.28 show that this effect is clearly visible for *perfect* Byzantines, for 1 and 5 robots respectively, and for C, R, T, and Sceptical Behaviors.



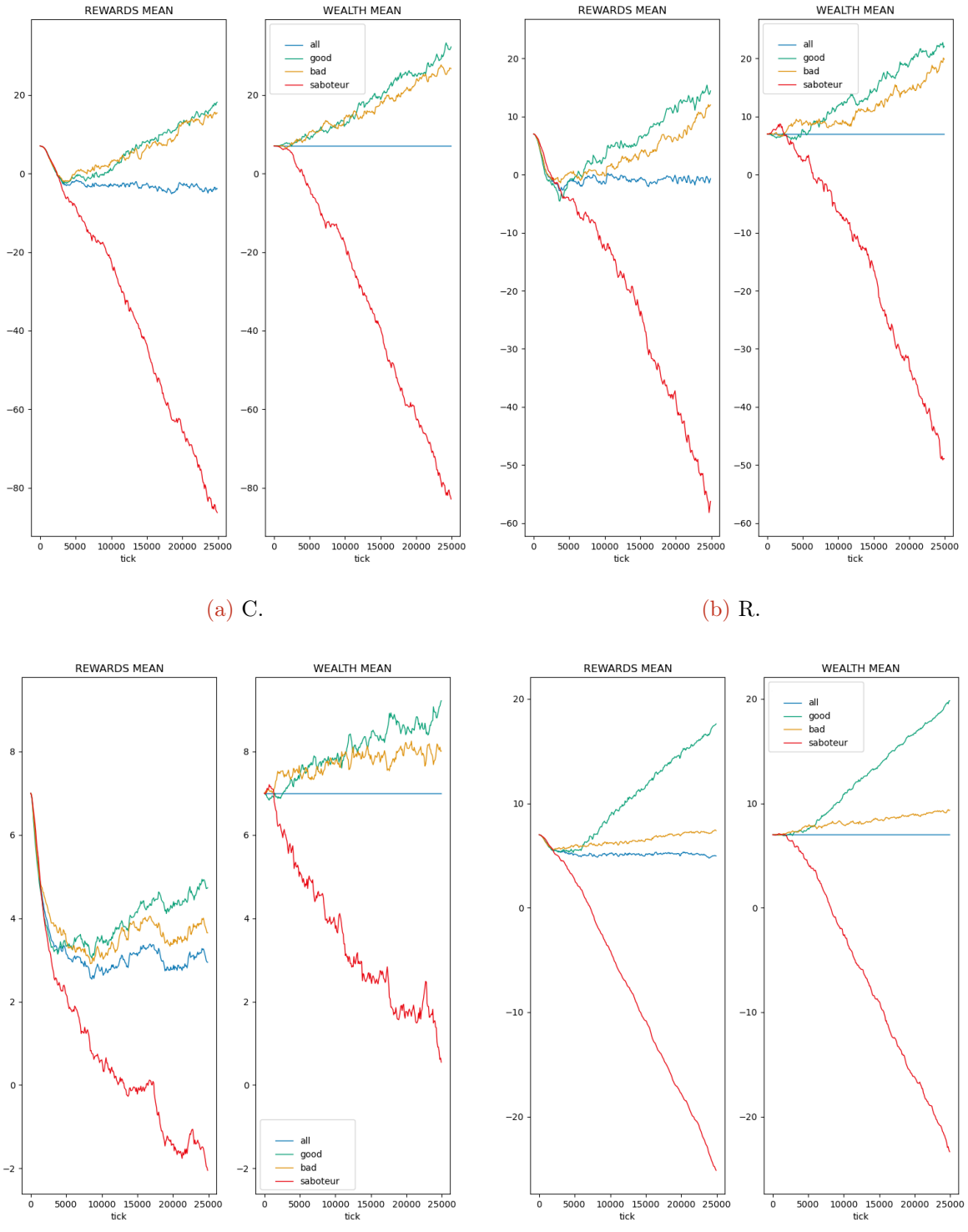
(a) C.

(b) R.

(c) T.

(d) S.

Figure 3.28: Group Separation IMD 24perf



(a) C.

(b) R.

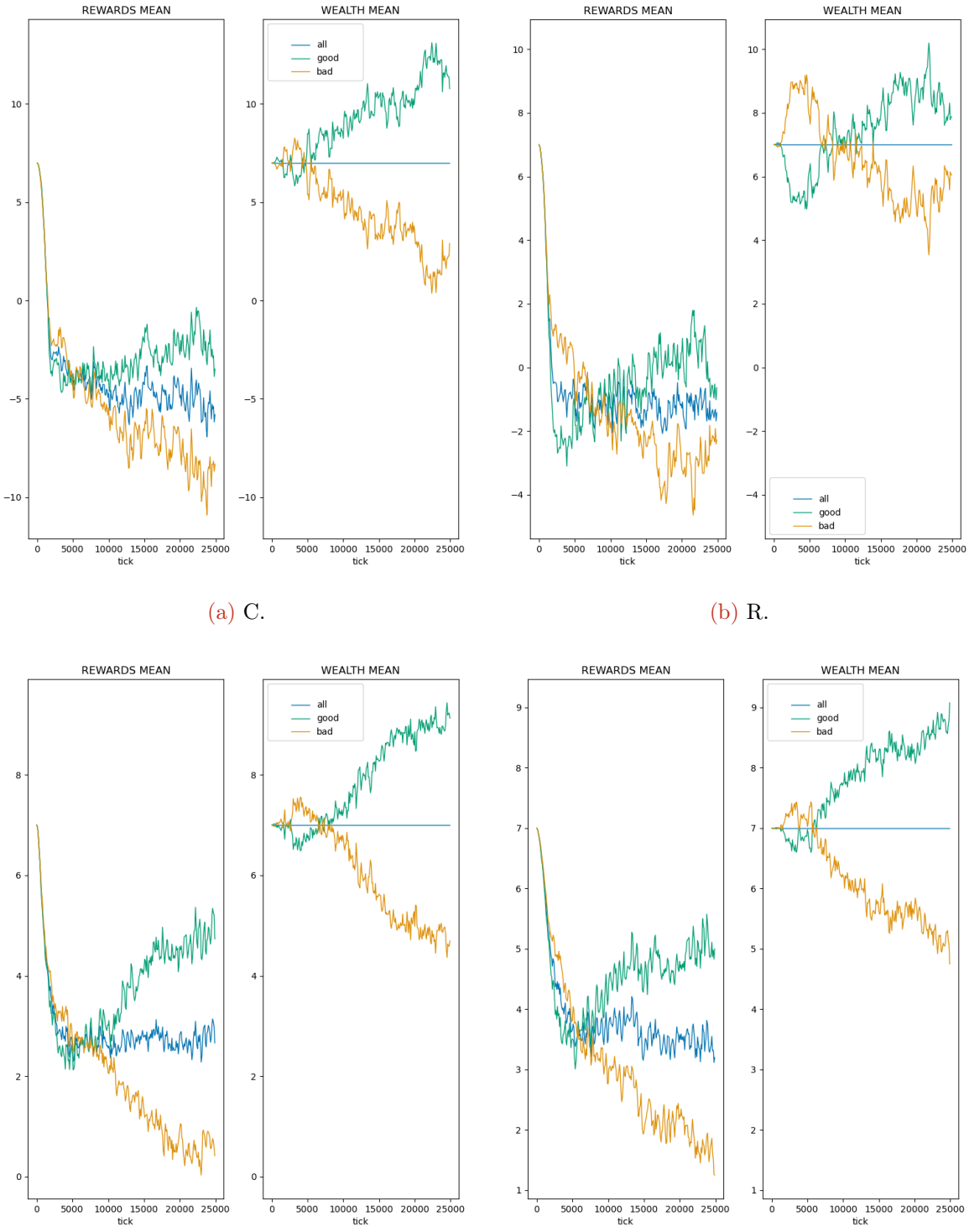
(c) T.

(d) S.

Figure 3.29: Group Separation IMD 20avg

T behavior, which I hypothesized is better suited for conditions where the separation

between groups is larger, does suffer from a bigger Byzantine presence, despite having good performances with just 1 of them.



(a) C.

(b) R.

(c) T.

(d) S.

Figure 3.30: Group Separation IMD 25

Figure 3.30 shows that C and R Behaviors are prone to make debt even in the absence of Byzantines, and have less separated groups for what it concerns the reward; considering the total wealth, the separation is more visible. To different degrees, all strategies suffer from group inversion, also the Sceptical Behavior, which proven to be resistant against this phenomenon in the other market conditions.

Considering the foraging performances, surprisingly T behavior suffers from the debit mechanism when no Byzantine is present. Considering the wealth evolution plots of Figure 3.30, the T Behavior is does not present a bad group inversion and does not make as much debt as R, but still performs worse. The low acceptance rates show that too many messages are rejected, especially for the *good* group.

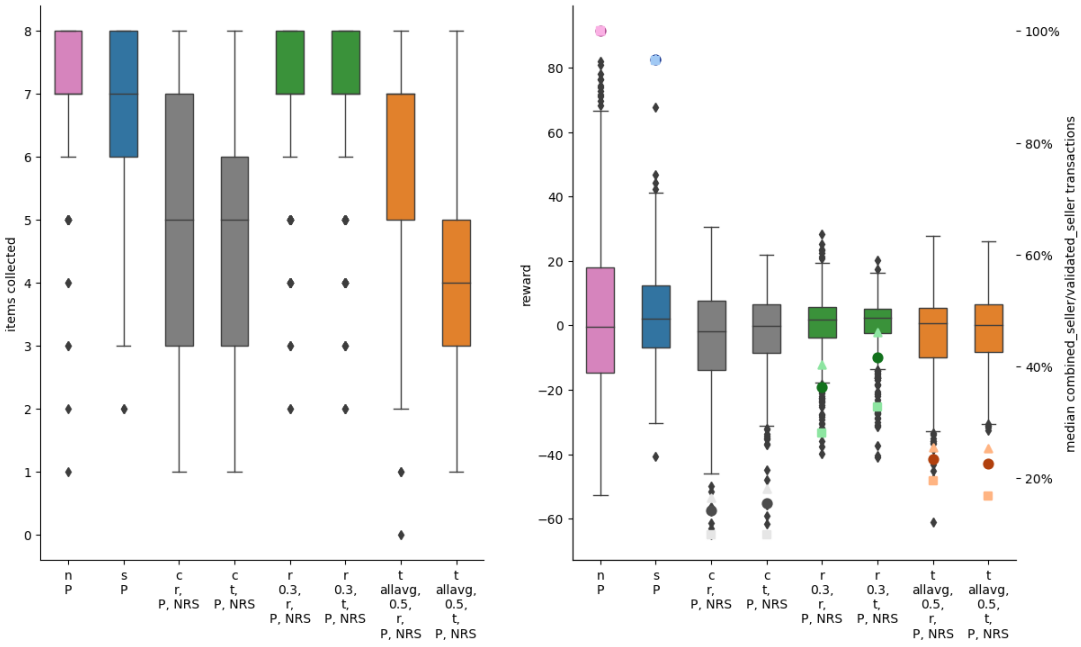


Figure 3.31: Foraging IMD 25

Figures 3.31 and 3.33 show that, when Byzantines are present, T behavior is able to perform as good as R behavior, and considerably better than Sceptical behavior, which is unable to protect the robots in the case of 3 *perfect* Byzantines. T behavior obtains an increment of performances with the respect to the case without debit of Figure 3.26. In fact, is all the figures it is possible to see the amount of debit that the Byzantines make on average, represented by the red box-and-whisker candles in the plots on right.

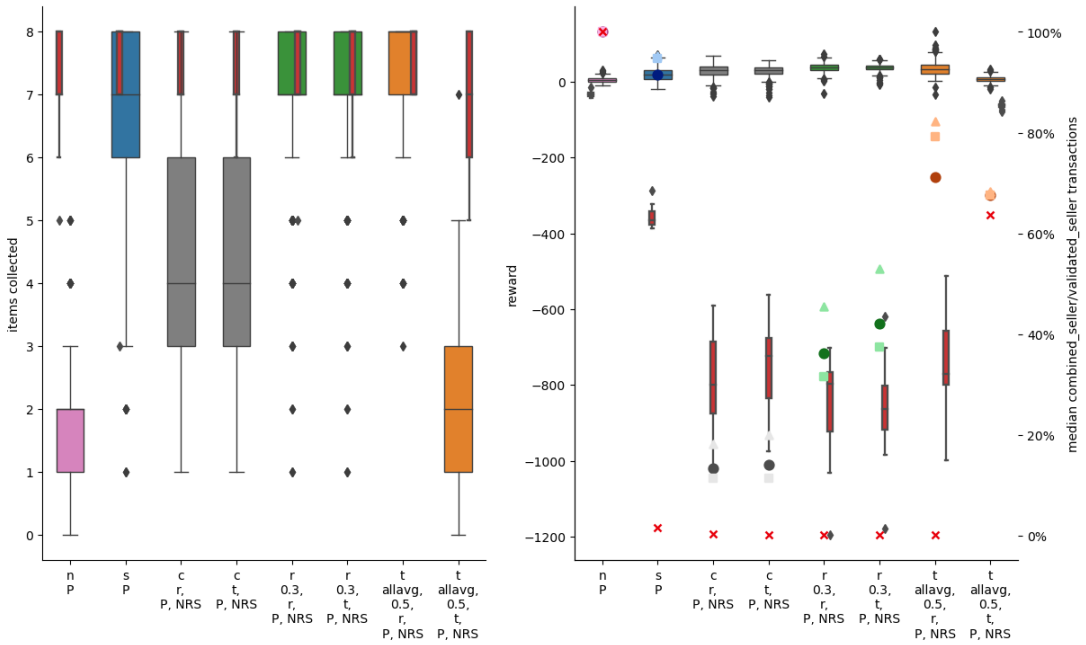


Figure 3.32: Foraging IMD 24 1perf

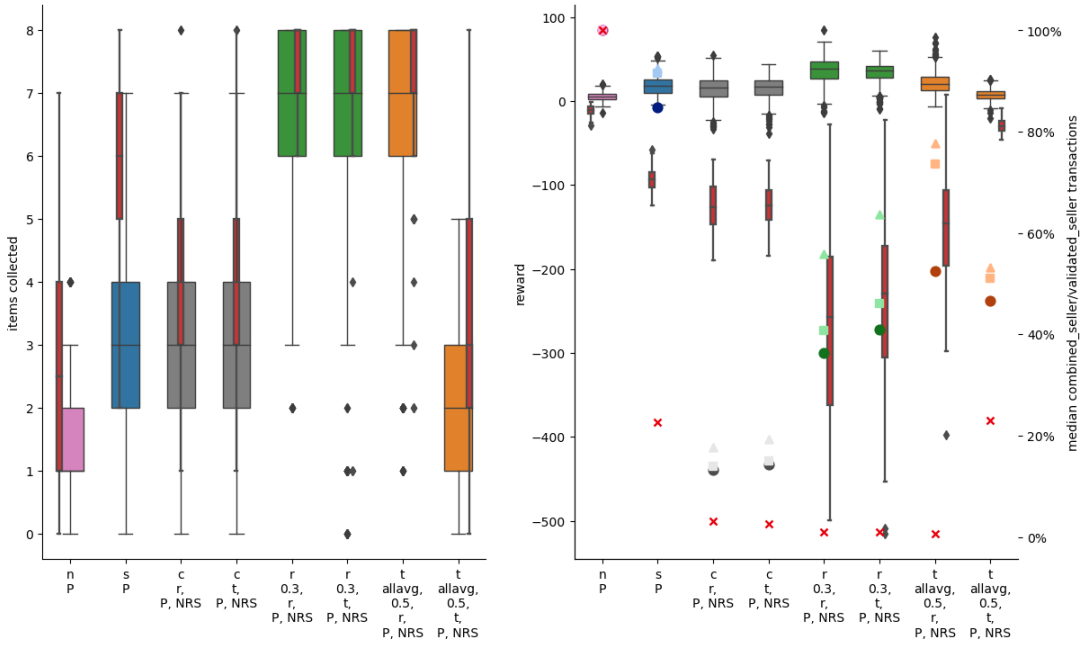


Figure 3.33: Foraging IMD 22 3perf

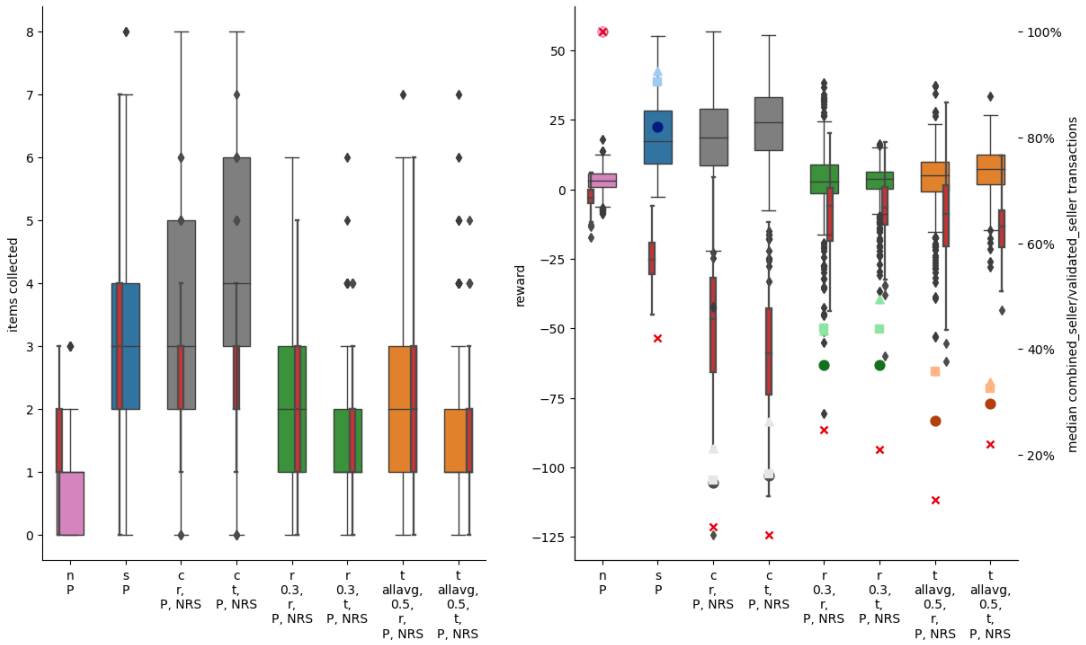


Figure 3.34: Foraging IMD 17 8avg

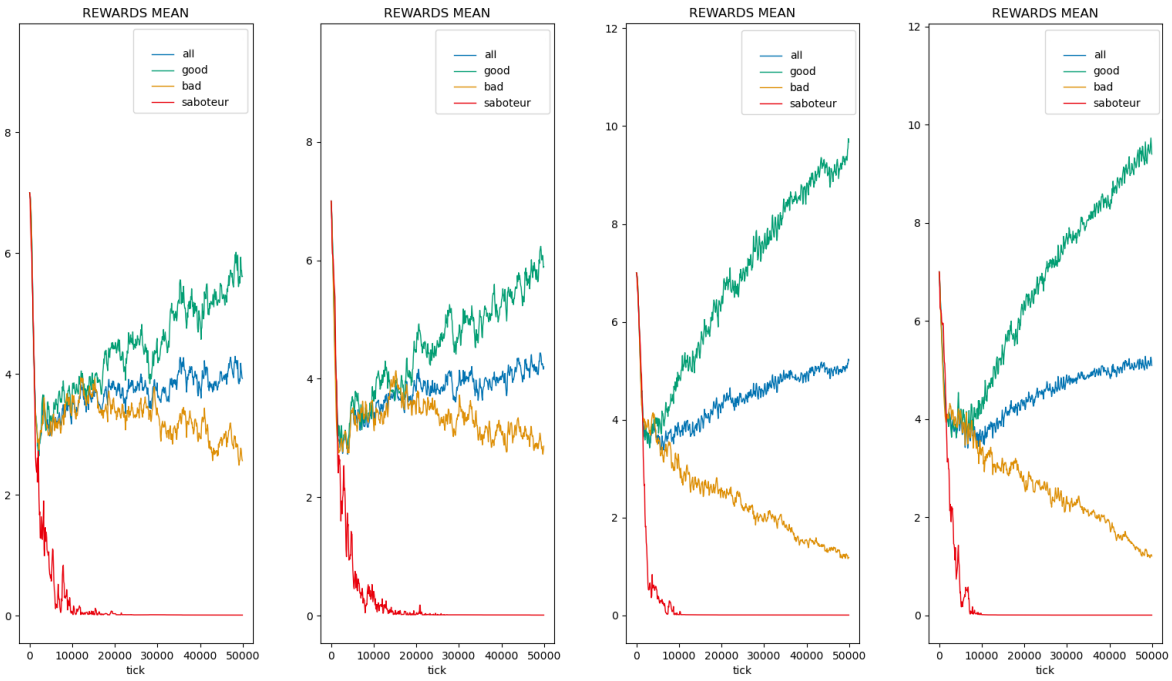
When considering Figure 3.34, the intrinsic robustness of Sceptical Behavior emerges, but C behavior shows that its simple strategy improves considerably when the number of robots able to make large amounts of debts is large.

To summarize, forcing the Byzantines to show a more clear separation with the other groups is able to improve their identification. When their number is too large, their ability to collude becomes stronger, and the possibility of a sceptical approach helps to avoid that. Wealth can still offer a promising solution; sensitivity analysis and fine parameter tuning are compulsory to obtain the best from this methods.

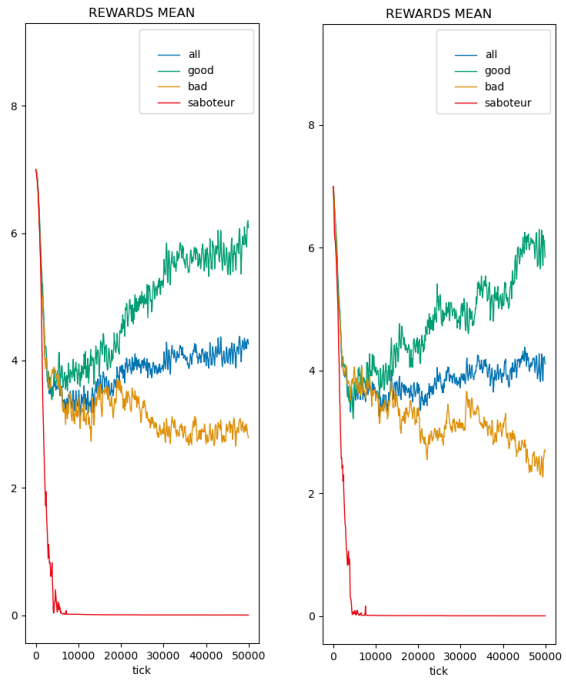
3.4. Stability using Reputation-based Staking

In this section, I show results related to a variation of the staking mechanism, when the amount that is required to initiate a transaction depends on the reputation of the seller. Since staking is one of the main mechanisms that drive the evolution of the market, my intent is to increase its effect and force a faster differentiation of groups

Figures 3.35 and 3.36 show the introduction of variable staking in a market without foraging reward and debit. In the two cases, both with no Byzantines or a single *perfect* one, the effect of variable staking is negligible. Group inversion appears to some extent, but there is no direct correlation between the staking type used.

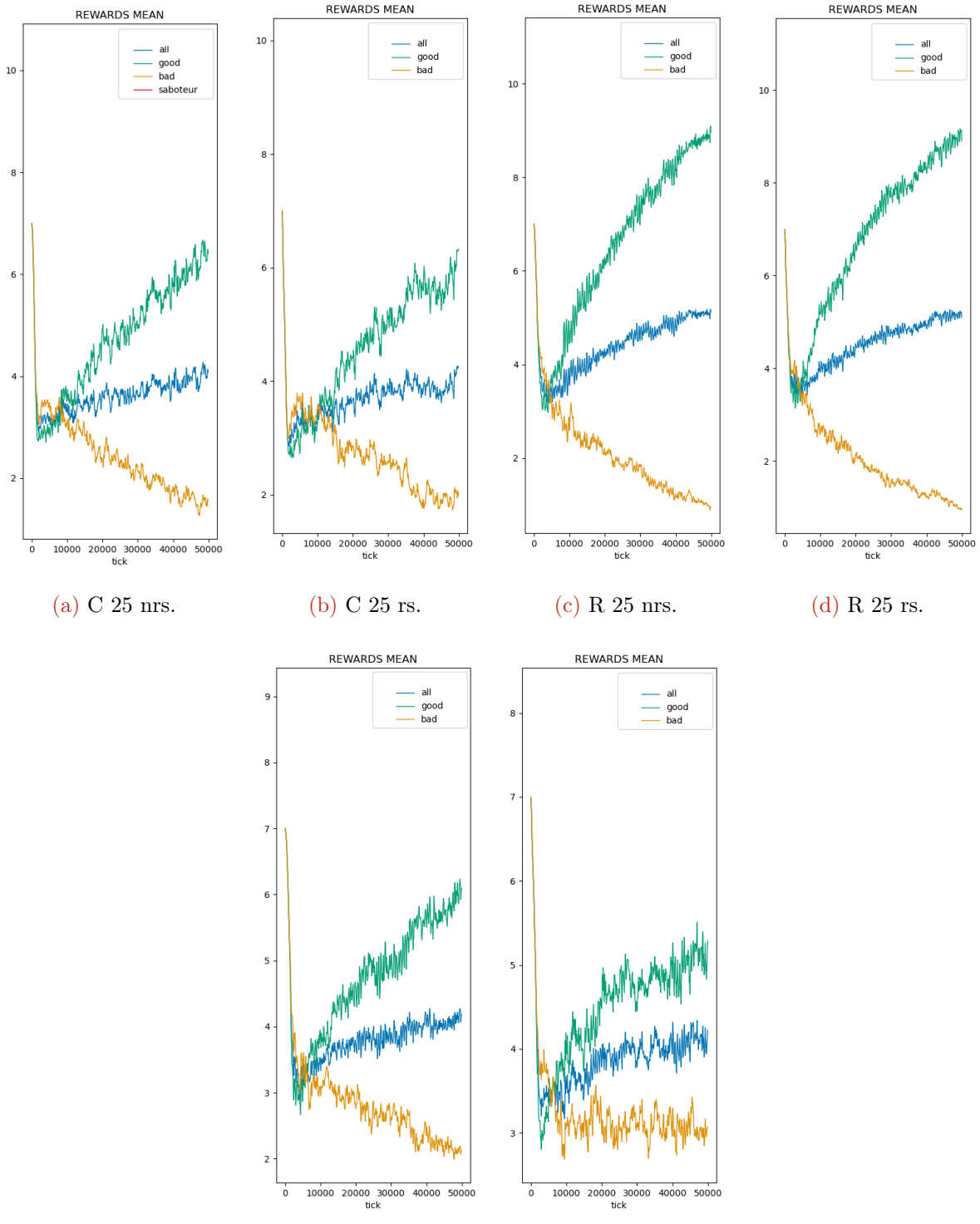


(a) C 24 perf nrs. (b) C 24 perf rs. (c) R 24 perf nrs. (d) R 24 perf rs.



(e) T 24 perf nrs. (f) T 24 perf rs.

Figure 3.35: Group Separation IM 24 perf



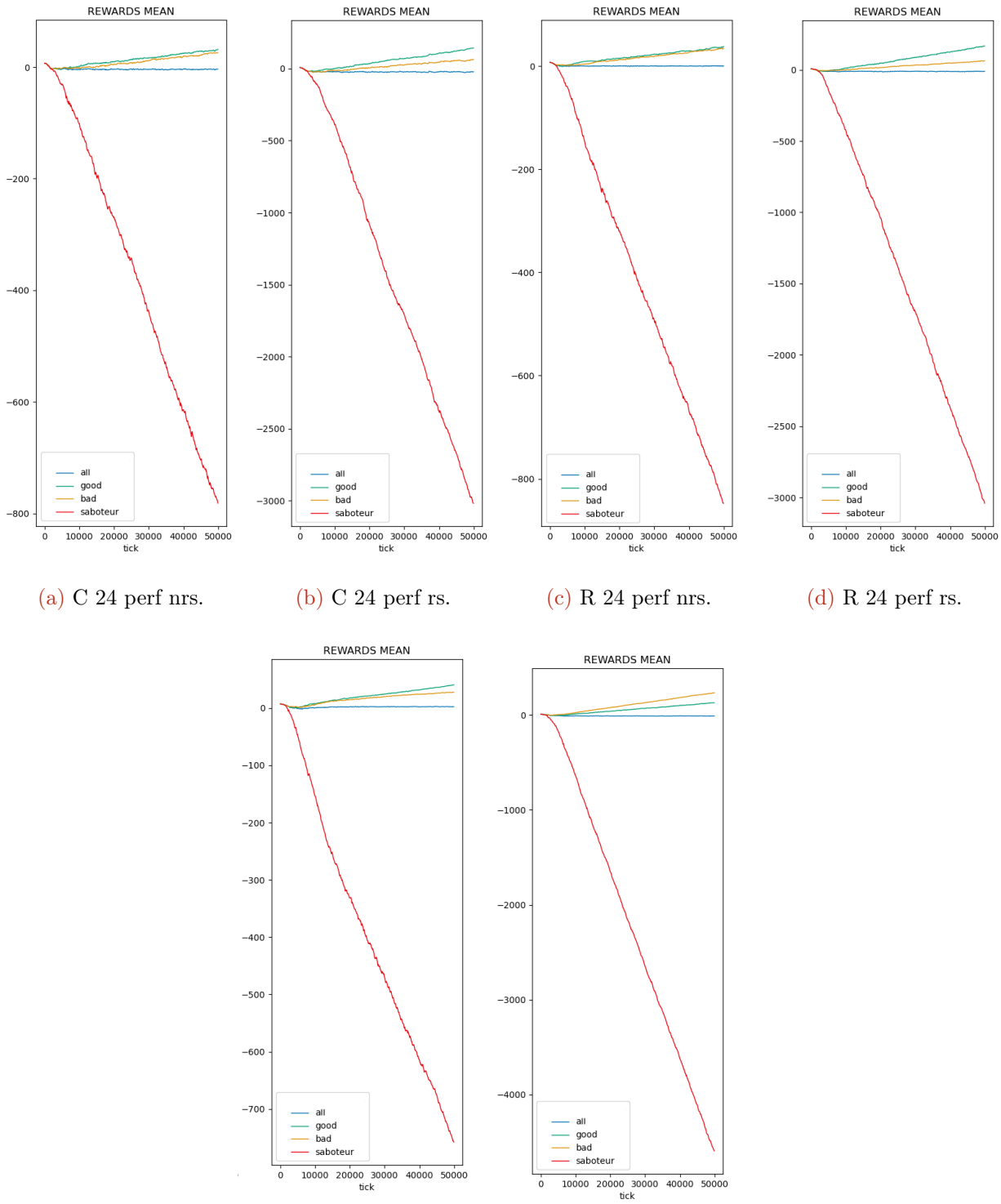
(a) C 25 nrs. (b) C 25 rs. (c) R 25 nrs. (d) R 25 rs.

(e) T 25 nrs. (f) T 25 rs.

Figure 3.36: Group Separation IM 25

Figures 3.37 and 3.38 show how the variable staking mechanism operates in the market

with debit. The effect on the group divergence speed, as expected, is clearly visible, reaching one order of magnitude higher in the last case.

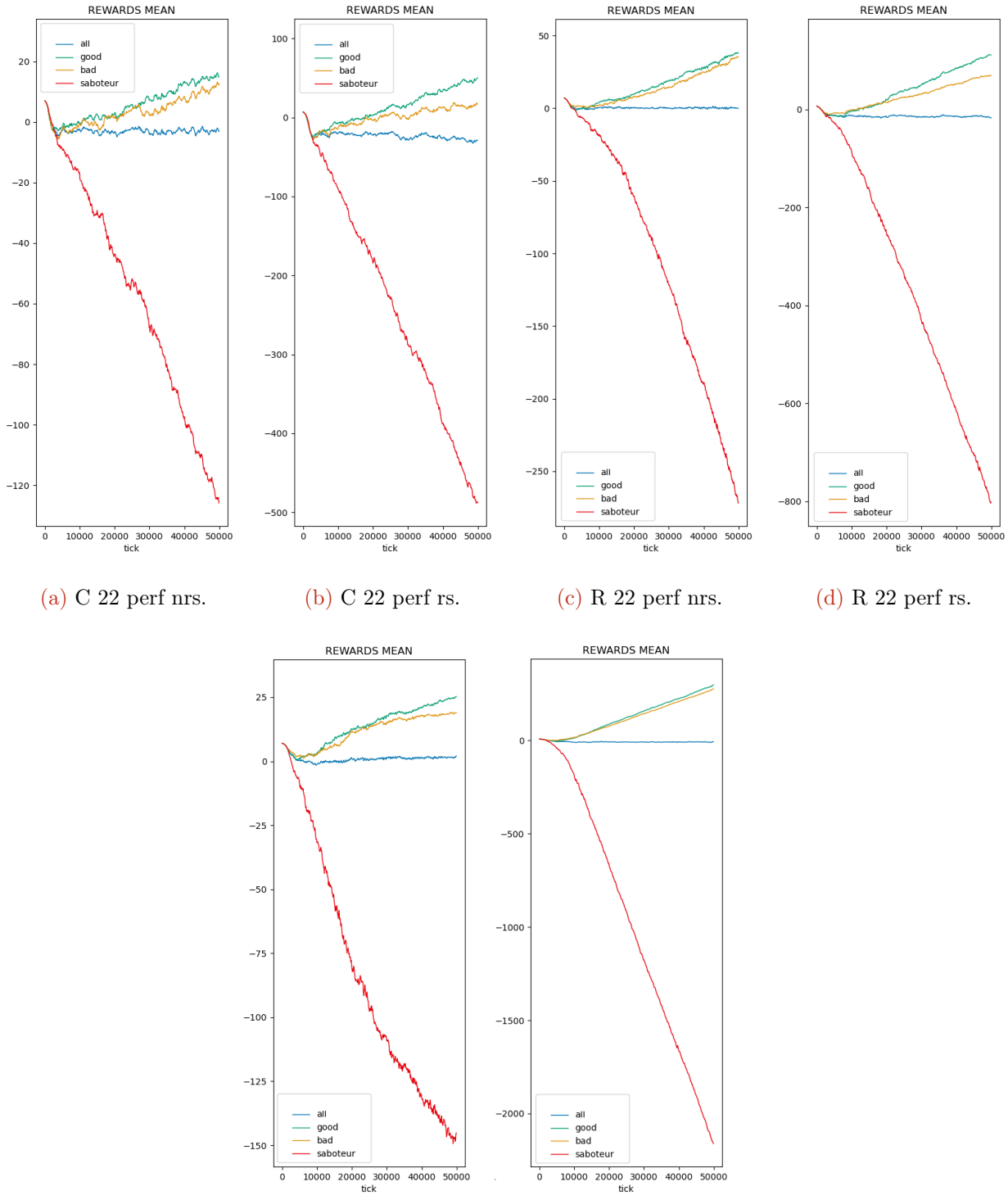


(a) C 24 perf nrs. (b) C 24 perf rs. (c) R 24 perf nrs. (d) R 24 perf rs.

(e) T 24 perf nrs. (f) T 24 perf rs.

Figure 3.37: Group Separation IMD 24 perf

Separation between groups improves, but the inversion is still possible, especially at the start of the experiment.



(a) C 22 perf nrs. (b) C 22 perf rs. (c) R 22 perf nrs. (d) R 22 perf rs.

(e) T 22 perf nrs. (f) T 22 perf rs.

Figure 3.38: Group Separation IMD 22 perf

Figures 3.39 and 3.40 show that the similitude in the wealth evolution reflect also in the foraging performance, with no real difference in the case of the market without foraging reward.

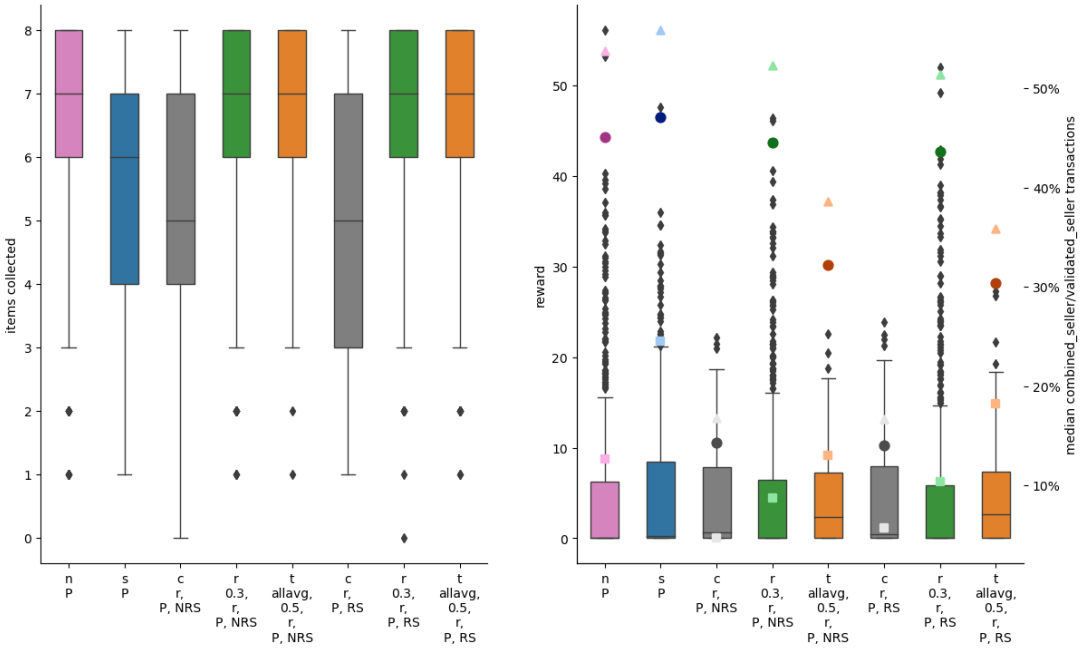


Figure 3.39: Foraging IM 25 Variable Stacking

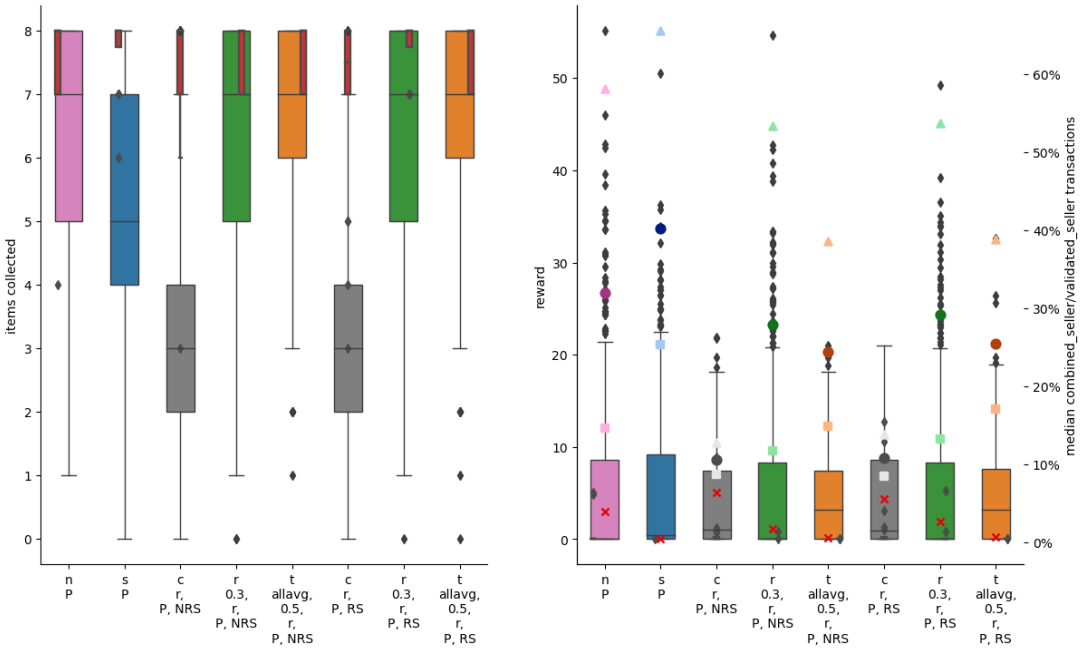


Figure 3.40: Foraging IM 24 1perf Variable Stacking

Figures 3.41 and 3.42 instead offer an insight on the fact that the Byzantines are now

able to use debit for staking insane amount of wealth, are able to counteract some the benefits previously introduced in this setup. But since they cannot use this debit to contact communicate with more robot, it is possible that the *bad* group is using the debit as well, as shown in their mean wealth distribution.

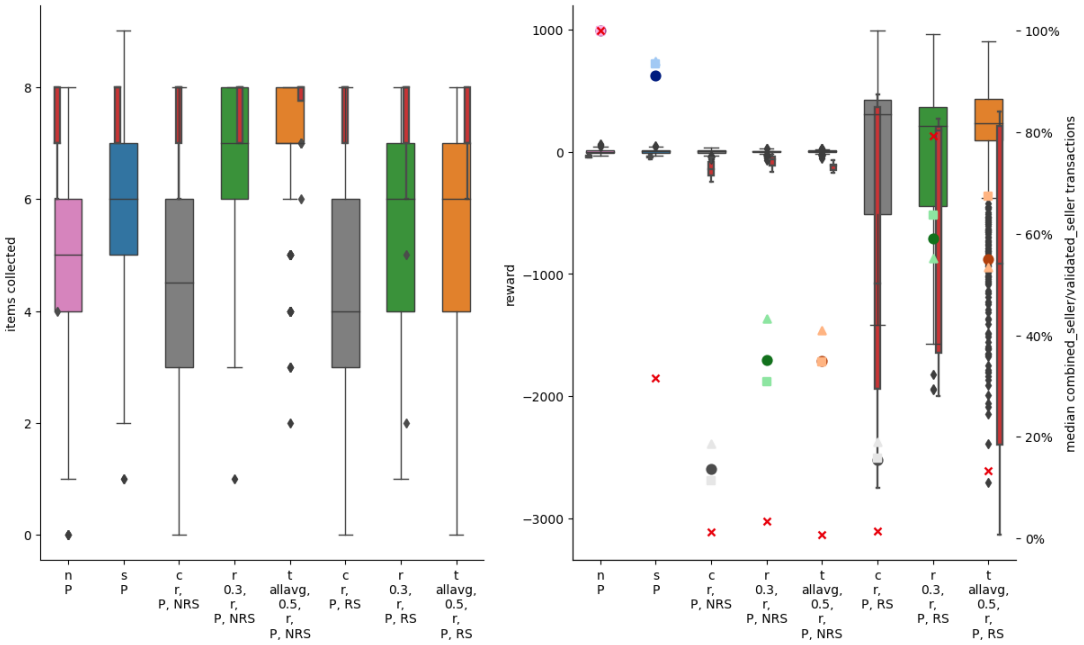


Figure 3.41: Foraging IMD 24 1perf Variable Stacking

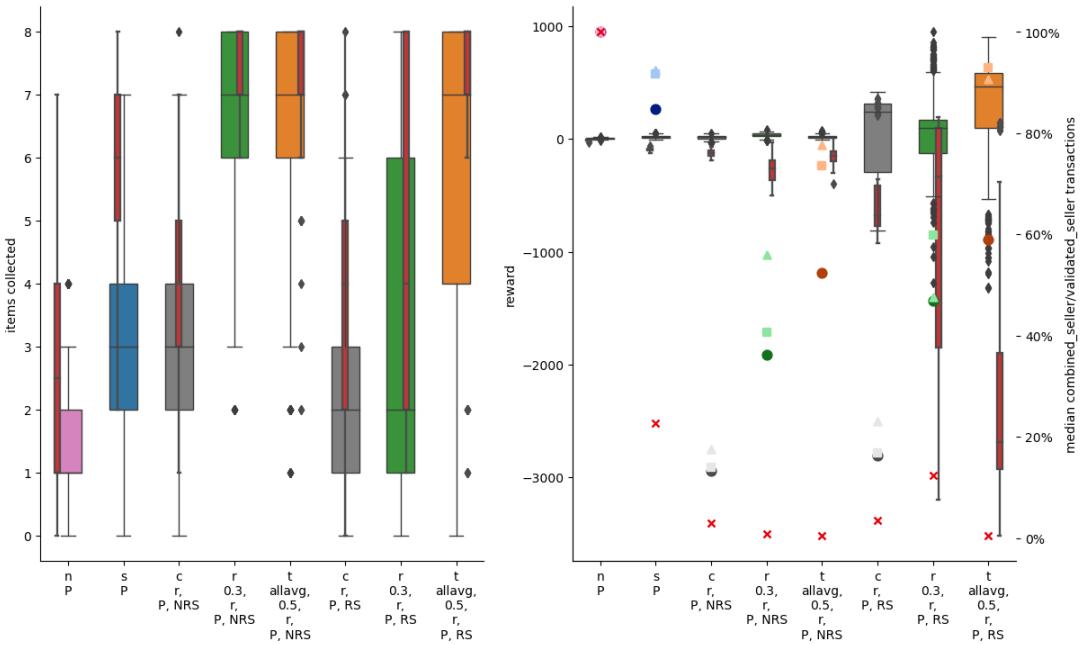


Figure 3.42: Foraging IMD 22 1perf Variable Stacking

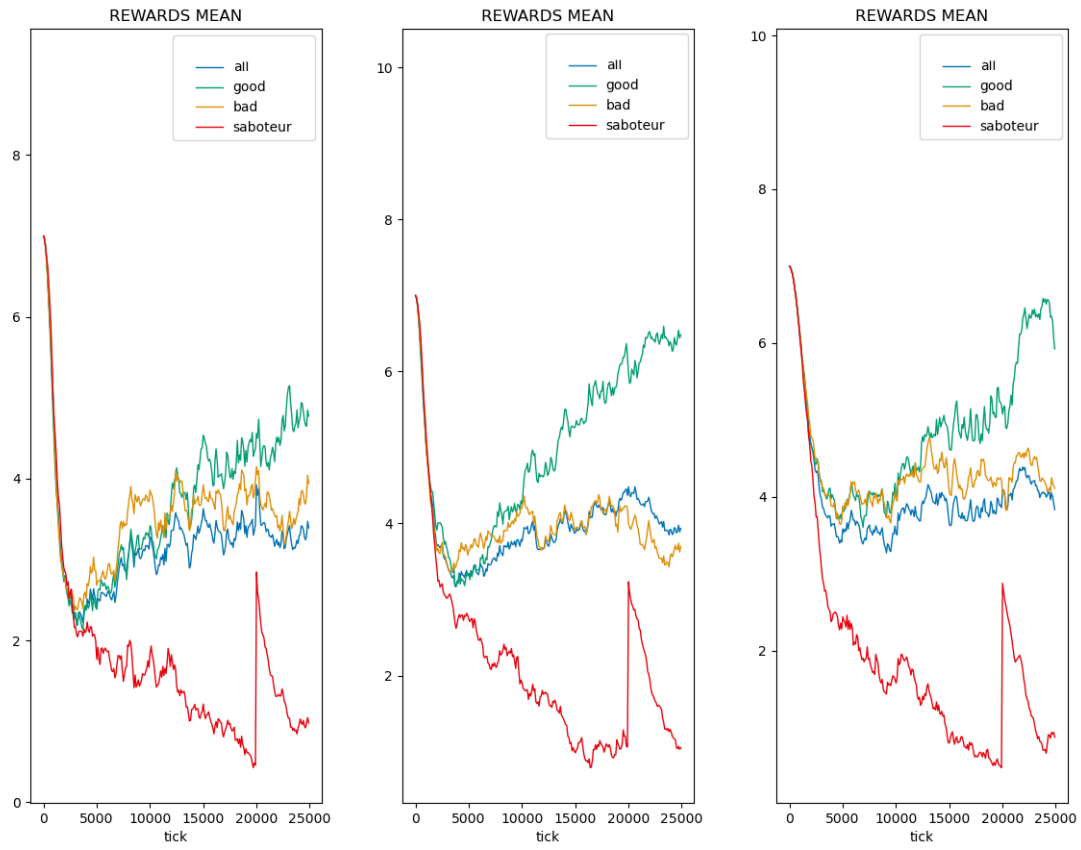
A marginal improvement in the T Behavior is visible, motivated by the most clear separation between the wealth of Byzantine and honest in all the experiments. Instead C Behavior suffers from the biggest reduction in performances because, since the values of reputation tend to be more extreme, especially the negative ones, this will reduce the number of sources of information.

3.5. Stability in Open Swarm Scenario

To conclude my work, I test how the introduction of a new wave of attackers influences the stability and robustness of the market.

Originally, Byzantines were present from the beginning of the experiments, while now a second group of 3 dishonest robots is introduced after 20000 steps of the simulation, for a duration of 5000 additional steps. This robots enter the environment with a neutral reputation and the same wealth that all the other robots received at the start of the experiment. This case is the first one because it could represent a Sybil attack, where a robot forges a new identity to delete all the bad reputation it had until that moment. In this case, I don't remove a robot to effectively simulate the attack, but I add a new one, leaving the former Byzantine in the experiment, able to continue its attack.

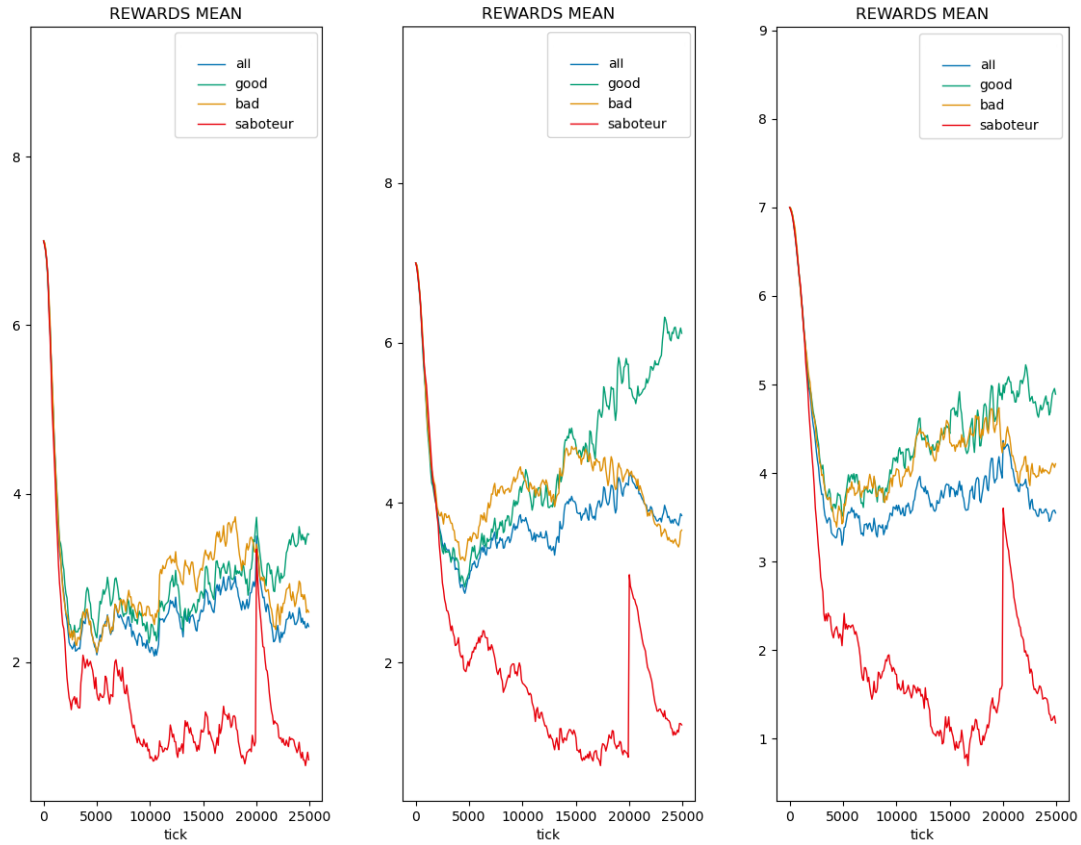
The following figures present an attack conducted against a swarm that is already hosting 5 Byzantines in it: the addition of 3 of them brings their number above the 33% of population of the swarm, which is a very disadvantageous situation.



(a) C 20avg nrs. (b) R 20avg nrs. (c) T 20avg nrs.

Figure 3.43: wealth evolution newcomers IM 20avg

The introduction of the new-coming Byzantines is evident by the abrupt increase of wealth of their group, seen in Figure 3.43. However, this resources cannot be used because the system is readily able to deplete the attackers of wealth. R Behavior appears the less disturbed by this market oscillation, while T Behavior presents a sudden increase of wealth, followed by a collapse. In the case of *average* Byzantines, all behaviors show robustness in the case of market without foraging reward.

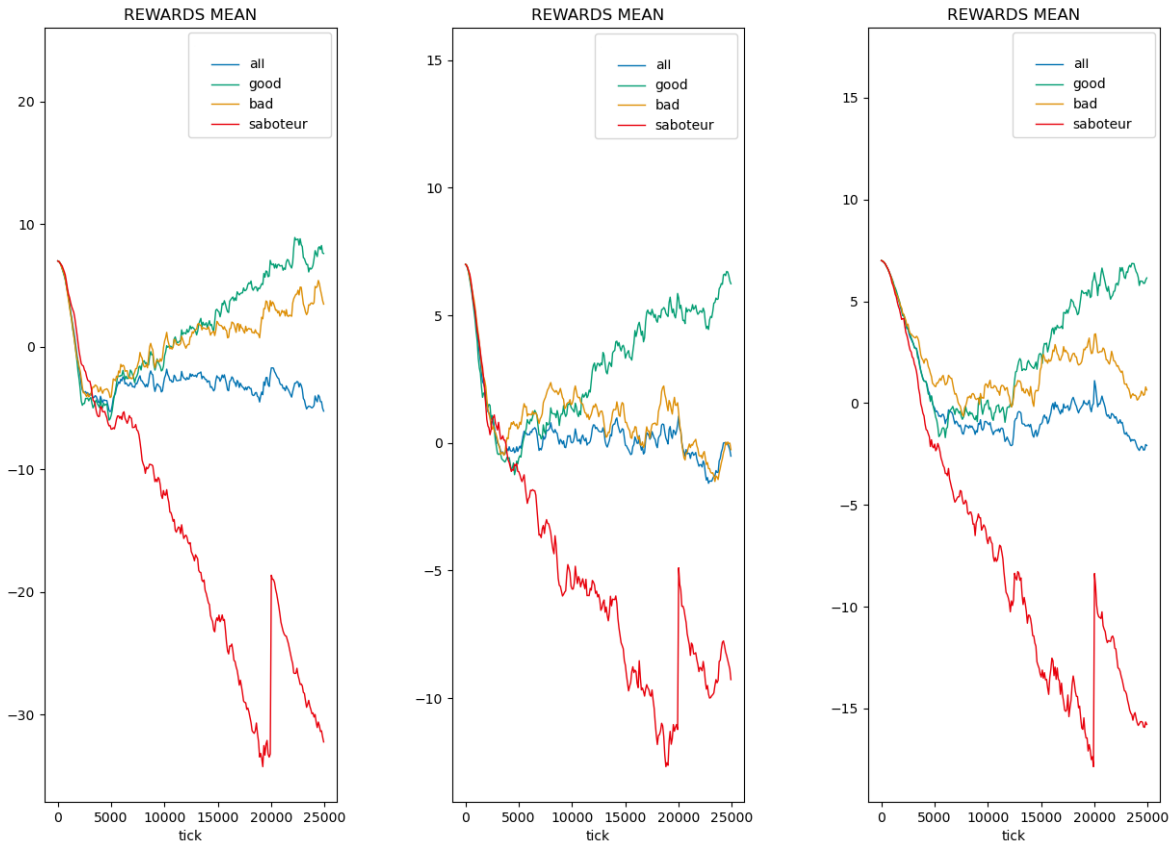


(a) C 20perf nrs. (b) R 20perf nrs. (c) T 20perf nrs.

Figure 3.44: wealth evolution newcomers IM 20perf

Figure 3.44 shows that this same market is also robust in the case of *perfect* Byzantines. Only R behavior is completely unhindered by the attack, while C and T have a noticeable change

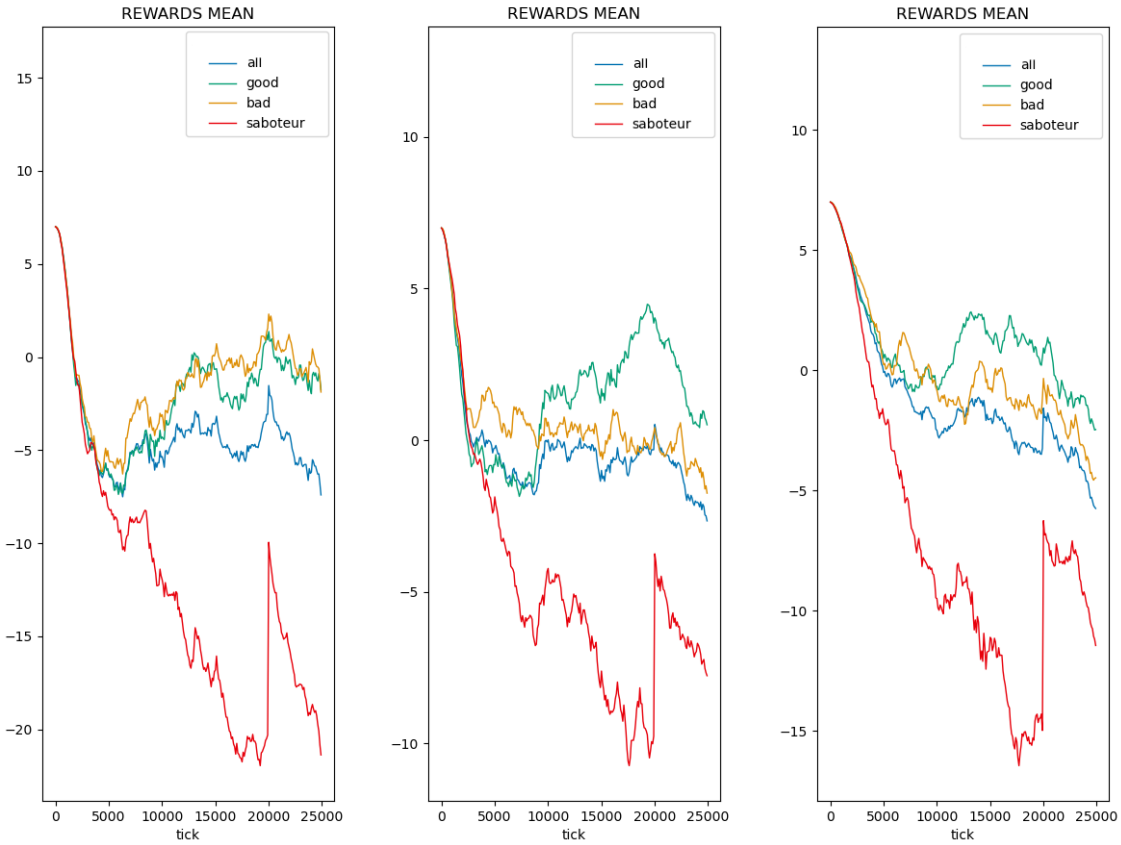
To summarize, market without foraging reward displays strong robustness against the Sybil attacks, even in the case where the Byzantines are above 33% of the population.



(a) C 20avg nrs. (b) R 20avg nrs. (c) T 20avg nrs.

Figure 3.45: wealth evolution newcomers imd 20avg

Considering a market with debit, Figures 3.45 shows the attack of *average* Byzantines. The *bad* group is the one more coinvolted by the perturbation, while the effect of the *good* one appears to be transitory.



(a) C 20perf nrs. (b) R 20perf nrs. (c) T 20perf nrs.

Figure 3.46: wealth evolution newcomers imd 20perf

Figures 3.46 instead shows that the attack of the additional *perfect* Byzantines on the debit market is able to disrupt the protection mechanism. Since the plot shows the evolution of reward, a decrease for all the groups indicates that the time needed to regain the stake is increased, due to the fact that the attack was successful in destroying the chain between the sites.

To summarize, the market with debit cannot resist a Sybil attack; since this type of attack can happen at any time and at any frequency, this market cannot be considered robust.

4 | Conclusion and Future Work

4.1. Discussion

My work proves that a reputation management system,based on wealth is a possible form of distributed, yet at the same time centralized, controller for an open swarm of robots, where cooperation robots act in a selfish way and cooperation is but one of the possible strategies they can adopt. This duality is possible thanks to the use of the a distributed ledger residing on the blockchain, and is enforced by market rules and tamper proof contracts. The market is able to stimulate cooperation by rewarding the diffusion of useful information.

I test three different types of market, and three different different behaviors. The market that rewards foraging work shows that reputation is able to show that the assay on the quality of the information source is not anymore limited by of locality, thanks to the use of the blockchain. The market that removes that reward shows that it was able to help the Byzantines; this new market show that the wealth-based reputation could be able to defend the swarm in a better way than scepticism. A market that introduces debt, finally, shows that reputation protects the swarm better when the inequality between the attackers and the honests is enhanced.

The R Behavior, based on the ordered the relative, ordered, wealth-ranks proves to be the more robust in all the situations. A finer tuning and a deeper analysis must be performed to understand the margin of improvement of the former and the other behaviors.

A sceptical approach is however useful when the Byzantines population approaches the 33% limit of the total one.

Nonetheless, I prove that the reputation protection method is robust against two types of Byzantine attack: a simple zealot attack where a robot simply lies, by a fixed amount, about the information they try to share, and a Sybil attack, where a dishonest robot can create a new identity to clear its past non-cooperative behavior.

The metrics that I designed are useful to analyse the performances of the systems, and

to possibly predict how it could perform.

In my work I considered another metric and another method to predict the performances, but the analysis is far from being completed. However, I present two aspects that for me are interesting to consider.

4.2. Wealth Distribution and Inequality

I prove robots amass reward proportionally to their information quality, the latter inversely proportional to the odometry noise defined in Equation (2.1). The result will be an unequal distribution of wealth, advisable to be used in the reputation system to identify the Byzantine. However, I am interested in understanding if the distribution of wealth inside the honest group, namely between *good* and *bad* groups can impact the performances of the swarm. It must be noted that the robots are able to sell the information investing their available wealth W^t, hence having too little of it may have a disrupting effect in the spreading of information: due to the limitation of the spreading of information, caused by the locality of the transactions, in some situations it may be useful to buy from robots belonging to the *bad* group, accepting a higher error and relying on the averaging protocol, instead of rejecting the information. In the first case, the distribution of total wealth inside the honest group would be more equal, while in the latter the wealth will be more condensed in the *good* group.

To characterise inequality, I consider level of wealth distribution as aggregated per group, to show the link with the noise and byzantinity, and, after normalizing in the range of extrema for each experiment, in quantiles, in a number equal to the robot population, in order to show the probability of belonging to a given wealth bracket. Moreover, I identify three wealth classes, with the intent to map them onto the noise level of groups: *poor* class should be filled by Byzantines, depleted of resources and unable to interact with the market; *middle* and *rich* classes, by *bad* and *good* noise groups, with the latter being rewarded for their quality. Despite this rationale, it's pretty easy so see in wealth distribution that, although *poor* class numerosity is close to the number of Byzantine, there is a net separation between the wealth of *middle* and *rich* class in many cases. Often it appears in the form of *superrich* entities, that condensate a lot of wealth, and sometimes even in the form of an oligarchy, where a single robot possess 10-30% of the total wealth. The reason for this could be a snowballing effect, where the *rich* are able to exploit the system and become even richer. To compute the classes, I use the heuristics

shown in Equations (4.1) and (4.2)

$$robot_i \; is \; rich \; if \begin{cases} rank(\mathscr{W}_i^t) \in highest \; 13\% \; of \; wealth \\ \mathscr{W}_i^t > \underset{j}{mean}(\mathscr{W}_j^t) + 0.8 \, \underset{j}{var}(\mathscr{W}_j^t) \end{cases} \tag{4.1}$$

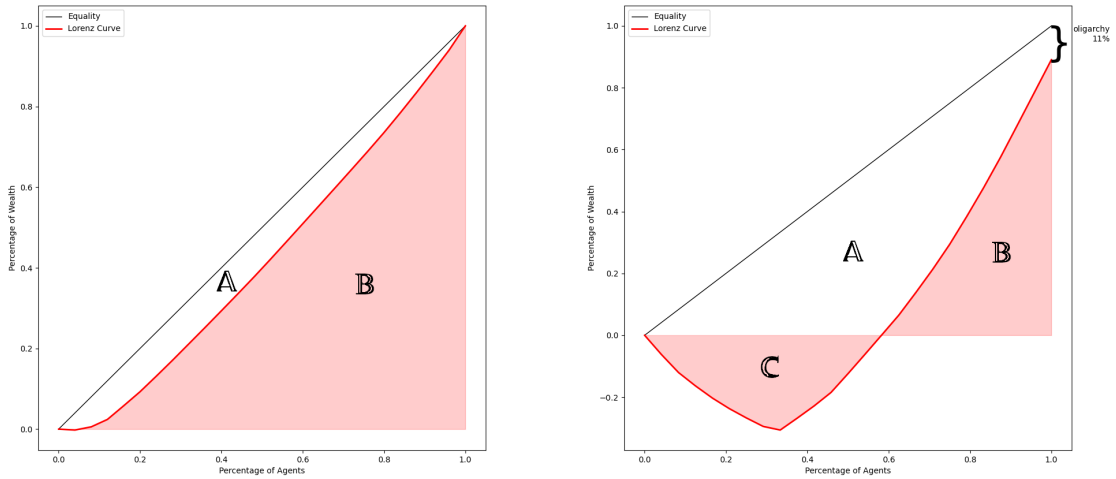
$$robot_i \; is \; poor \; if \; rank(W_i) \in lowest \; 10\% \; of \; wealth \tag{4.2}$$

A more practical way to study wealth distribution is to use the Lorenz curve [33]: it shows the amount of wealth possessed by the fraction of robots, ordered by increasing wealth, in the normalized [1,1] square; I compute the curve as in [32]. The flatter is the curve is, the more equally distributed the wealth is in the system, with the bisector line corresponding to perfect equality. A flat part close the origin corresponds to that the poor class have very little wealth; the steeper the curve is to the right (1,1) limit, the wealthier the rich are. On to boundaries of the curve it holds that $L(0) = 0$ and $L(1) = 1$. Ideally, the curve should be flat for the Byzantine population, belonging to the *poor* class, and slightly above linear for the honest robots. It is also convenient to identify an oligarchy when a robot has a very high amount of wealth, using the rule of Equation (4.3)

$$robot_i \; is \; oligarch \; if \; \frac{\mathscr{W}_i^t}{\mathscr{W}_P^t} \geq 0.1 \tag{4.3}$$

where $\mathscr{W}_P^t = \sum_{j \in P} \mathscr{W}_j^t$ and P is the robot population.
This will be removed from the plot, and in this case $L(1) = 1 - \frac{\mathscr{W}_{oligarchy}}{\mathscr{W}_P^t} < 1$.

From this curve it is possible to compute the Gini inequality coefficient [10], the ratio between the area between the Lorenz and the ideal wealth distribution curves and the one subtended by the the ideal wealth distribution curve, respectively named $\mathbb{A}$ and $\mathbb{B}$ in Figure 4.1; value of 0 indicates perfect equality, while a value of 1 indicated perfect inequality, with wealth condensed in only one robot. The advantage of this metric is to aggregate all the measure of inequality in a scalar value. I compute the Gini coefficient using the affine wealth model [32], as in Equation (4.4); even in the case of the debit market, where negative wealth can be present, this wealth model still offers a way to compute the coefficient using ratios between areas, also considering the area $\mathbb{C}$ below the x-axis.



(a) In a market without debit.

(b) In a market with debit.

Figure 4.1: Lorenz curve in different markets, with equality curve and $\mathbb{A}$,$\mathbb{B}$ and $\mathbb{C}$ areas indicated. In (b) it is possible to see the effect of the oligarchy.

$$Gini = \begin{cases} \dfrac{\mathbb{A}}{\mathbb{B}} = 2\mathbb{A}, & \text{if } \textit{market without debit} \\ \dfrac{\mathbb{A}-\mathbb{C}}{\mathbb{A}-\mathbb{C}+\mathbb{B}} = \dfrac{2(\mathbb{A}-\mathbb{C})}{2(\mathbb{A}-\mathbb{C})+1}, & \text{if } \textit{market with debit} \end{cases} \tag{4.4}$$

In my experiments, I observed different levels of inequality and wealth distribution. It would be interesting to understand how they influence the performances, considering that, for example, the wealth-inversion of the *good* and *bad* groups not always resulted in poor performances.

Understanding the relation between wealth distribution and performances, could shed the light also on the effect of $super-rich$ robots, the *oligarch* that I previously mentioned.

Possibly, once a target wealth distribution is known, it would be possible to impose a more powerful redistribution of wealth, based on taxation, charity and fairness between the honest in general.

In the following section I show a possible tool to compute the sought-after wealth distribution.

$\mathcal{B}$ Behavior

By considering the trading partnership mechanisms, as seen from the outside, it seems to be completely described only by the acceptance rate of transactions that each robot

experiences; my interest is then to understand if this assumption is true and, therefore, if it is possible to describe each and all mechanisms taking place at the same time in the system using an equivalent acceptance rate for each robot. I test a stochastic behavior where it is possible to configure the acceptance rate for each robot, differentiating them by their ID and the noise quality of the Byzantines to order them with respect to the honest, as shown in Figure 2.2. To simplify it, each robot in a given noise group has the same acceptance rate and I will impose it to the mean acceptance rate at the end of the experiment I want to mimic.

I'm also interested in finding an approximated correlation between acceptance rates of the noise groups and the foraging performances. The goal here is to understand how the acceptance of messages from each group is able, on average to influence the performance and the wealth distribution in a given market, eventually finding a correlation between all of them and an effective performance predictor.

In this case a random number generated by a $random()$ function, with codomain= $[0, 1]$, is compared with the selected acceptance rate, for the noise group $robot_i$ belongs to, as shown in Equations (4.5) and (4.6). A_{good}, A_{bad} and $A_{byzantine}$ represents respectively the acceptance rates for the *good*, *bad* and *byzantine* groups.

$$combine\_info() \; if \; random() \leq A(robot_i) \tag{4.5}$$

$$A(robot_i) = \begin{cases} A_{good}, & if \; robot_i \in good \; group \\ A_{bad}, & if \; robot_i \in bad \; group \\ A_{byzantine}, & otherwise \end{cases} \tag{4.6}$$

In Table 4.1, A_{good}, A_{bad} and $A_{byzantine}$ are as defined in the previous paragraph, $|P|$ and $|B|$ are the population and Byzantine amount, η is the Byzantine noise odometry performance, defined in Section 2.2.1

	A_{good}	A_{bad}	$A_{byzantine}$	$\|P\|$	$\|B\|$	η
$\mathcal{B}$	[0.5,0.95]	[0.35,0.75]	[0.05,0.45]	25	[0,8]	average, perfect

Table 4.1: Parameters and used value ranges for $\mathcal{B}$ behavior

When a particular wealth distribution is known, related to a particular acceptance rates set, forced wealth redistribution could be used to enforce it, and hence obtain the wanted performances of the varios groups.

Bibliography

[1] D. Albani, D. Nardi, and V. Trianni. Field coverage and weed mapping by UAV swarms. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Sept. 2017. doi: 10.1109/iros.2017.8206296. URL https://doi.org/10.1109/iros.2017.8206296.

[2] A. L. Alfeo, E. C. Ferrer, Y. L. Carrillo, A. Grignard, L. A. Pastor, D. T. Sleeper, M. G. C. A. Cimino, B. Lepri, G. Vaglini, K. Larson, M. Dorigo, and A. lSandy' Pentland. Urban swarms: A new approach for autonomous waste management. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, May 2019. doi: 10.1109/icra.2019.8794020. URL https://doi.org/10.1109/icra.2019.8794020.

[3] J. C. Biesmeijer and H. de Vries. Exploration and exploitation of food sources by social insect colonies: a revision of the scout-recruit concept. *Behavioral Ecology and Sociobiology*, 49(2-3):89–99, Jan. 2001. doi: 10.1007/s002650000289. URL https://doi.org/10.1007/s002650000289.

[4] M. D. Breed and J. Moore. Foraging. In *Animal Behavior*, pages 309–341. Elsevier, 2022. doi: 10.1016/b978-0-12-819558-1.00009-9. URL https://doi.org/10.1016/b978-0-12-819558-1.00009-9.

[5] V. Buterin. A next-generation smart contract and decentralized application platform. https://ethereum.org/en/whitepaper, 2014.

[6] A. Campo, Á. Gutiérrez, S. Nouyan, C. Pinciroli, V. Longchamp, S. Garnier, and M. Dorigo. Artificial pheromone for path selection by a foraging swarm of robots. *Biological Cybernetics*, 103(5):339–352, July 2010. doi: 10.1007/s00422-010-0402-x. URL https://doi.org/10.1007/s00422-010-0402-x.

[7] J. Carbo, J. M. Molina, and J. Davila. Trust management through fuzzy reputation. *International Journal of Cooperative Information Systems*, 12(01):135–155, Mar. 2003. doi: 10.1142/s0218843003000681. URL https://doi.org/10.1142/s0218843003000681.

[8] C. Castelfranchi and R. Falcone. Towards a theory of delegation for agent-based

systems. *Robotics and Autonomous Systems*, 24(3-4):141–157, Sept. 1998. doi: 10.1016/s0921-8890(98)00028-1. URL https://doi.org/10.1016/s0921-8890(98)00028-1.

[9] C. Castellano, S. Fortunato, and V. Loreto. Statistical physics of social dynamics. *Reviews of Modern Physics*, 81(2):591–646, May 2009. doi: 10.1103/revmodphys.81.591. URL https://doi.org/10.1103/revmodphys.81.591.

[10] L. Ceriani and P. Verme. The origins of the gini index: extracts from variabilità e mutabilità (1912) by corrado gini. *The Journal of Economic Inequality*, 10(3):421–443, June 2011. ISSN 1573-8701. doi: 10.1007/s10888-011-9188-x. URL http://dx.doi.org/10.1007/s10888-011-9188-x. (Original work published 1912 titled "Variabilità e mutabilità",by C. Gini).

[11] M. Dorigo and E. Şahin. Guest editorial. *Autonomous Robots*, 17(2/3):111–113, Sept. 2004. doi: 10.1023/b:auro.0000034008.48988.2b. URL https://doi.org/10.1023/b:auro.0000034008.48988.2b.

[12] M. Dorigo, D. Floreano, L. M. Gambardella, F. Mondada, S. Nolfi, T. Baaboura, M. Birattari, M. Bonani, M. Brambilla, A. Brutschy, D. Burnier, A. Campo, A. L. Christensen, A. Decugniere, G. Di Caro, F. Ducatelle, E. Ferrante, A. Forster, J. M. Gonzales, J. Guzzi, V. Longchamp, S. Magnenat, N. Mathews, M. Montes de Oca, R. O'Grady, C. Pinciroli, G. Pini, P. Retornaz, J. Roberts, V. Sperati, T. Stirling, A. Stranieri, T. Stutzle, V. Trianni, E. Tuci, A. E. Turgut, and F. Vaussard. Swarmanoid: A novel concept for the study of heterogeneous robotic swarms. *IEEE Robotics & Automation Magazine*, 20(4):60–71, Dec. 2013. ISSN 1070-9932. doi: 10.1109/mra.2013.2252996. URL http://dx.doi.org/10.1109/MRA.2013.2252996.

[13] M. Dorigo, G. Theraulaz, and V. Trianni. Reflections on the future of swarm robotics. *Science Robotics*, 5(49), Dec. 2020. doi: 10.1126/scirobotics.abe4385. URL https://doi.org/10.1126/scirobotics.abe4385.

[14] M. Dorigo, G. Theraulaz, and V. Trianni. Swarm robotics: Past, present, and future [point of view]. *Proceedings of the IEEE*, 109(7):1152–1165, July 2021. doi: 10.1109/jproc.2021.3072740. URL https://doi.org/10.1109/jproc.2021.3072740.

[15] F. Ducatelle, G. A. D. Caro, C. Pinciroli, and L. M. Gambardella. Self-organized cooperation between robotic swarms. *Swarm Intelligence*, 5(2):73–96, Mar. 2011. doi: 10.1007/s11721-011-0053-0. URL https://doi.org/10.1007/s11721-011-0053-0.

[16] F. Ducatelle, G. A. D. Caro, C. Pinciroli, F. Mondada, and L. M. Gambardella. Communication assisted navigation in robotic swarms: Self-organization and coop-

eration. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Sept. 2011. doi: 10.1109/iros.2011.6094454. URL https://doi.org/10.1109/iros.2011.6094454.

[17] A. M. Edwards, R. A. Phillips, N. W. Watkins, M. P. Freeman, E. J. Murphy, V. Afanasyev, S. V. Buldyrev, M. G. E. da Luz, E. P. Raposo, H. E. Stanley, and G. M. Viswanathan. Revisiting lévy flight search patterns of wandering albatrosses, bumblebees and deer. *Nature*, 449(7165):1044–1048, Oct. 2007. doi: 10.1038/nature06199. URL https://doi.org/10.1038/nature06199.

[18] R. Falcone and C. Castelfranchi. Social trust: A cognitive approach. In *Trust and Deception in Virtual Societies*, pages 55–90. Springer Netherlands, 2001. doi: 10.1007/978-94-017-3614-5_3. URL https://doi.org/10.1007/978-94-017-3614-5_3.

[19] R. Fujisawa, S. Dobata, K. Sugawara, and F. Matsuno. Designing pheromone communication in swarm robotics: Group foraging behavior mediated by chemical substance. *Swarm Intelligence*, 8(3):227–246, Aug. 2014. doi: 10.1007/s11721-014-0097-z. URL https://doi.org/10.1007/s11721-014-0097-z.

[20] S. Goss, S. Aron, J. L. Deneubourg, and J. M. Pasteels. Self-organized shortcuts in the argentine ant. *Naturwissenschaften*, 76(12):579–581, Dec. 1989. doi: 10.1007/bf00462870. URL https://doi.org/10.1007/bf00462870.

[21] Á. Gutiérrez, A. Campo, F. C. Santos, F. Monasterio-Huelin, and M. Dorigo. Social odometry: Imitation based odometry in collective robotics. *International Journal of Advanced Robotic Systems*, 6(2):11, June 2009. doi: 10.5772/6794. URL https://doi.org/10.5772/6794.

[22] H. Hamann. *Swarm Robotics: A Formal Approach*. Springer International Publishing, 2018. doi: 10.1007/978-3-319-74528-2. URL https://doi.org/10.1007/978-3-319-74528-2.

[23] N. Hoff, R. Wood, and R. Nagpal. Distributed colony-level algorithm switching for robot swarm foraging. In *Springer Tracts in Advanced Robotics*, pages 417–430. Springer Berlin Heidelberg, 2013. doi: 10.1007/978-3-642-32723-0_30. URL https://doi.org/10.1007/978-3-642-32723-0_30.

[24] A. I. Houston and J. M. McNamara. A general theory of central place foraging for single-prey loaders. *Theoretical Population Biology*, 28(3):233–262, Dec. 1985. doi: 10.1016/0040-5809(85)90029-2. URL https://doi.org/10.1016/0040-5809(85)90029-2.

[25] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, Mar. 2006. doi: 10.1007/s10458-005-6825-4. URL https://doi.org/10.1007/s10458-005-6825-4.

[26] B. R. Johnson. Division of labor in honeybees: form, function, and proximate mechanisms. *Behavioral Ecology and Sociobiology*, 64(3):305–316, Nov. 2009. doi: 10.1007/s00265-009-0874-7. URL https://doi.org/10.1007/s00265-009-0874-7.

[27] D. L. Johnson, N. Ntlatlapa, and C. Aichele. Simple pragmatic approach to mesh routing using batman. In *2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries*, 2008. URL https://api.semanticscholar.org/CorpusID:15392544.

[28] T. Kazama, K. Sugawara, and T. Watanabe. Collecting behavior of interacting robots with virtual pheromone. In *Distributed Autonomous Robotic Systems 6*, pages 347–356. Springer Japan, 2004. doi: 10.1007/978-4-431-35873-2_34. URL https://doi.org/10.1007/978-4-431-35873-2_34.

[29] A. A. Khaliq, M. D. Rocco, and A. Saffiotti. Stigmergic algorithms for multiple minimalistic robots on an RFID floor. *Swarm Intelligence*, 8(3):199–225, July 2014. doi: 10.1007/s11721-014-0096-0. URL https://doi.org/10.1007/s11721-014-0096-0.

[30] J. Krause and G. D. Ruxton. *Living in Groups*. Oxford University PressOxford, Oct. 2002. doi: 10.1093/oso/9780198508175.001.0001. URL https://doi.org/10.1093/oso/9780198508175.001.0001.

[31] L. Lamport, R. Shostak, and M. Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, July 1982. doi: 10.1145/357172.357176. URL https://doi.org/10.1145/357172.357176.

[32] J. Li, B. M. Boghosian, and C. Li. The affine wealth model: An agent-based model of asset exchange that allows for negative-wealth agents and its empirical validation. *Physica A: Statistical Mechanics and its Applications*, 516:423–442, Feb. 2019. doi: 10.1016/j.physa.2018.10.042. URL https://doi.org/10.1016/j.physa.2018.10.042.

[33] M. O. Lorenz. Methods of measuring the concentration of wealth. *Publications of the American Statistical Association*, 9(70):209, June 1905. ISSN 1522-5437. doi: 10.2307/2276207. URL http://dx.doi.org/10.2307/2276207.

[34] R. Mayet, J. Roberz, T. Schmickl, and K. Crailsheim. Antbots: A feasible visual em-

ulation of pheromone trails for swarm robots. In *Lecture Notes in Computer Science*, pages 84–94. Springer Berlin Heidelberg, 2010. doi: 10.1007/978-3-642-15461-4_8. URL https://doi.org/10.1007/978-3-642-15461-4_8.

[35] K. N. McGuire, C. D. Wagter, K. Tuyls, H. J. Kappen, and G. C. H. E. de Croon. Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment. *Science Robotics*, 4(35), Oct. 2019. doi: 10.1126/scirobotics.aaw9710. URL https://doi.org/10.1126/scirobotics.aaw9710.

[36] O. Olsson, J. S. Brown, and K. L. Helf. A guide to central place effects in foraging. *Theoretical Population Biology*, 74(1):22–33, Aug. 2008. doi: 10.1016/j.tpb.2008.04.005. URL https://doi.org/10.1016/j.tpb.2008.04.005.

[37] A. Pacheco, V. Strobel, and M. Dorigo. A blockchain-controlled physical robot swarm communicating via an ad-hoc network. In *Lecture Notes in Computer Science*, pages 3–15. Springer International Publishing, 2020. doi: 10.1007/978-3-030-60376-2_1. URL https://doi.org/10.1007/978-3-030-60376-2_1.

[38] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. D. Caro, F. Ducatelle, M. Birattari, L. M. Gambardella, and M. Dorigo. ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems. *Swarm Intelligence*, 6(4):271–295, Nov. 2012. doi: 10.1007/s11721-012-0072-5. URL https://doi.org/10.1007/s11721-012-0072-5.

[39] I. Pinyol and J. Sabater-Mir. Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review*, 40(1):1–25, July 2011. doi: 10.1007/s10462-011-9277-z. URL https://doi.org/10.1007/s10462-011-9277-z.

[40] E. P. Raposo, S. V. Buldyrev, M. G. E. da Luz, G. M. Viswanathan, and H. E. Stanley. Lévy flights and random searches. *Journal of Physics A: Mathematical and Theoretical*, 42(43):434003, Oct. 2009. doi: 10.1088/1751-8113/42/43/434003. URL https://doi.org/10.1088/1751-8113/42/43/434003.

[41] L. Rasmusson and S. Jansson. Simulated social control for secure internet commerce. In *Proceedings of the 1996 workshop on New security paradigms - NSPW '96*. ACM Press, 1996. doi: 10.1145/304851.304857. URL https://doi.org/10.1145/304851.304857.

[42] L. Rasmusson, A. Rasmusson, and S. Janson. Using agents to secure the internet marketplace reactive security and social control. In I. B. Crabtree, editor, *Proceedings of the Second International Conference on the Practical Application of Intelligent*

Agents and Multi-Agent Technology, PAAM 1997, Westminster Central Hall, London, UK, April 21-23, 1997, pages 193–206. Practical Application Company Ltd., 1997.

[43] A. Reina, A. J. Cope, E. Nikolaidis, J. A. R. Marshall, and C. Sabo. ARK: Augmented reality for kilobots. *IEEE Robotics and Automation Letters*, 2(3):1755–1761, July 2017. doi: 10.1109/lra.2017.2700059. URL https://doi.org/10.1109/lra.2017.2700059.

[44] T. D. Seeley. Division of labor between scouts and recruits in honeybee foraging. *Behavioral Ecology and Sociobiology*, 12(3):253–259, June 1983. doi: 10.1007/bf00290778. URL https://doi.org/10.1007/bf00290778.

[45] S. Sen and N. Sajja. Robustness of reputation-based trust. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems part 1 - AAMAS '02*. ACM Press, 2002. doi: 10.1145/544741.544808. URL https://doi.org/10.1145/544741.544808.

[46] V. Sperati, V. Trianni, and S. Nolfi. Self-organised path formation in a swarm of robots. *Swarm Intelligence*, 5(2):97–119, Apr. 2011. doi: 10.1007/s11721-011-0055-y. URL https://doi.org/10.1007/s11721-011-0055-y.

[47] V. Strobel, E. C. Ferrer, and M. Dorigo. Blockchain technology secures robot swarms: A comparison of consensus protocols and their resilience to byzantine robots. *Frontiers in Robotics and AI*, 7, May 2020. doi: 10.3389/frobt.2020.00054. URL https://doi.org/10.3389/frobt.2020.00054.

[48] V. Strobel, A. Pacheco, and M. Dorigo. Robot swarms neutralize harmful byzantine robots using a blockchain-based token economy. *Science Robotics*, 8(79), June 2023. doi: 10.1126/scirobotics.abm4636. URL https://doi.org/10.1126/scirobotics.abm4636.

[49] D. Sumpter and S. Pratt. A modelling framework for understanding social insect foraging. *Behavioral Ecology and Sociobiology*, 53(3):131–144, Feb. 2003. doi: 10.1007/s00265-002-0549-0. URL https://doi.org/10.1007/s00265-002-0549-0.

[50] P. Szilágyi. Eip-225: Clique proof-of-authority consensus protocol. https://github.com/ethereum/EIPs/issues/225, 2017.

[51] M. S. Talamali, T. Bose, M. Haire, X. Xu, J. A. R. Marshall, and A. Reina. Sophisticated collective foraging with minimalist agents: a swarm robotics test.

Swarm Intelligence, 14(1):25–56, Oct. 2019. doi: 10.1007/s11721-019-00176-9. URL https://doi.org/10.1007/s11721-019-00176-9.

[52] D. Tarapore, J. Timmis, and A. L. Christensen. Fault detection in a swarm of physical robots based on behavioral outlier detection. *IEEE Transactions on Robotics*, 35(6): 1516–1522, 2019. doi: 10.1109/TRO.2019.2929015.

[53] G. Valentini, A. Antoun, M. Trabattoni, B. Wiandt, Y. Tamura, E. Hocquard, V. Trianni, and M. Dorigo. Kilogrid: a novel experimental environment for the kilobot robot. *Swarm Intelligence*, 12(3):245–266, Jan. 2018. doi: 10.1007/s11721-018-0155-z. URL https://doi.org/10.1007/s11721-018-0155-z.

[54] L. Van Calck, A. Pacheco, V. Strobel, M. Dorigo, and A. Reina. A blockchain-based information market to incentivise cooperation in swarms of self-interested robots. *Scientific Reports*, 13(1), Nov. 2023. ISSN 2045-2322. doi: 10.1038/s41598-023-46238-1. URL http://dx.doi.org/10.1038/s41598-023-46238-1.

List of Figures

List of Tables

List of Symbols

Variable	Description	Unit of measure
$\mathscr{W}$	total wealth	tokens
W	available wealth or reward	tokens
$\mathcal{C}$	$\mathcal{C}$ behavior	-
$\mathcal{R}$	$\mathcal{R}$ behavior	-
$\mathcal{T}$	$\mathcal{T}$ behavior	-
$\mathcal{B}$	$\mathcal{B}$ behavior	-

Acknowledgements

Obviously I was also late for the acknowledgements too. There are a lot of important people that helped me to get to this point.

You know who you are.

To all the others I wanna say

Thanks

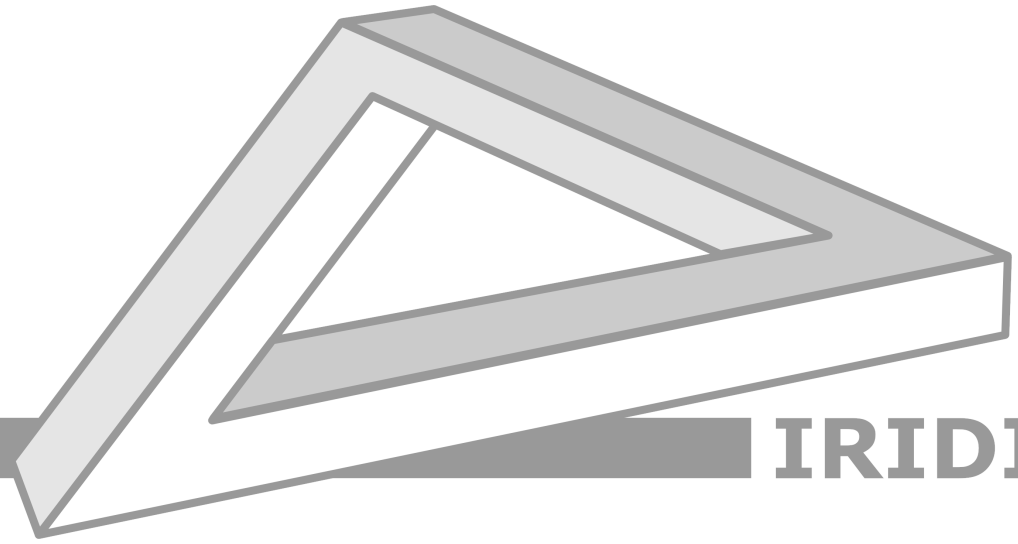
Merci

Danke

Gracias

Meio

Grazie



IRIDIA