

Distributed Norm Enforcement: Ostracism in Open Multi-Agent Systems

Adrian Perreau de Pinninck, Carles Sierra, and Marco Schorlemmer

IIIA – Artificial Intelligence Research Institute
CSIC – Spanish National Research Council
Bellaterra (Barcelona), Catalonia, Spain
`adrianp,sierra,marco@iia.csic.es`

Abstract. Enforcement in normative agent societies is a complex issue, which becomes more problematic as these societies become more decentralized and open. A new distributed mechanism is presented to enforce norms by ostracizing agents that do not abide by them in their interactions with other agents in the society. Simulations are run to check the mechanism’s impact in different types of societies. The simulations have shown that complete ostracism is not always possible, but the mechanism substantially reduces the number of norm violations.

1 Introduction

In an open multi-agent system (MAS), there is no easy control over who is allowed to join the system. Open MAS are composed of autonomous agents of all types, without a pre-defined structure. In a normative MAS a set of norms are added to restrict the set of available actions in order to improve the coordination between agents. An autonomous agent has the choice whether or not to support a norm. It is up to the agent to decide if it benefits itself to abide by the norm. A utility maximizer agent will follow a norm if it is profitable for it, it is in the agent’s own interest to act as the norm establishes. But some norms make it worthwhile for an agent to not abide by it if all other agents abide by them. For this kind of norms, an agent that does not adhere (*i.e.*, a violator) will profit at the expense of the agents that adhere.

Gnutella¹ is a suitable real life application to show how a multi-agent system may behave when norms are added. The scenario we will use in this paper is based on a simplification of this application. Gnutella is a pure peer-to-peer (P2P) file sharing application without centralized servers. Peers can share files on their hard drives so that others can download them. Each peer knows other peers (*i.e.*, friends or neighbours) with which it can interact. A peer can carry out two actions: search for peers that have a file it wants, and download the file from any of them. Peers are found by asking neighbouring peers if they or any of their neighbours have the file. This process is recursive. Once a list of peers that share the file are returned to the querying peer, it can choose one of them from

¹ <http://www.gnutella.org>

which to download the file. Anyone with a Gnutella compliant software can join the society as long as it has a list of other peers with which to interact.

This system works even though users can use the network to download files without sharing any files themselves. Therefore, the burden of sharing is carried only by some of them, while the rest benefit from the service. In order to solve this problem the following norm could be added: “Users must share files in order to be allowed to download files themselves”. But since the norm is not built into the Gnutella protocol, Gnutella peers can choose whether or not to adhere to it. A mechanism is needed so that norms can be enforced in any P2P network such as Gnutella.

This paper presents a new distributed mechanism that ostracises norm violating agents in an open MAS, thus attaining norm compliance. The test scenario in this paper allows agents in the MAS to interact with each other. The agents are structured in a network, in which agents can interact with the agents they are linked to directly or indirectly through a path of links (*i.e.*, agents can interact with direct neighbours, with neighbours of neighbours, and with their neighbours and so on...). The interaction initiator will search for a path in the society structure that leads to an interaction partner. All the agents in the path that are not the initiator or the partner agent are called mediator agents (*i.e.*, agents mediating the interaction).

A game-theoretic approach to interactions has been implemented. Interactions are modelled as a two-player game with two possible strategies; abide and violate. The utility function will be that of a prisoner’s dilemma (see Figure 1), since the total utility gained by both players is maximized if both players abide by the norm, and the maximum utility to be gained by a single agent is maximized if it violates the norm while the other abides by it.

PD	Abide	Violate
Abide	3,3	0,5
Violate	5,0	1,1

Fig. 1. Prisoner’s Dilemma Payoff Matrix

Violators are better off if they interact with norm-abiding agents, since they gain more utility. In order to attain norm enforcement the violators are not allowed to interact. Some agents in the society can enforce the norm through the ability to stop interacting with violators, and to stop them from interacting with the enforcer’s own neighbours. When all the neighbours of a violator are enforcers, and they use this ability against it, it is ostracised.

The motivation for this technique comes from the study of enforcement in primitive societies [12]. A member of a community that repeatedly ignores its customs is forced to leave upon general consent. No one in the community will interact with the ostracized member from then on. Therefore, the harsh natural conditions surrounding those communities mean death for the ostracised

member. Ostracism is achieved in human societies through force and physical constraint. If the ostracised member tried to return he may be killed. Achieving ostracism of electronic entities is a bit trickier, since they don't suffer pain. Inspiration has been sought from the network security area, where the most commonly used component is a firewall. Firewalls block those communications which appear to be harmful. The problem with firewalls is that they are usually set up by humans through complex rules, which must be updated manually. The enforcer agents in this paper will use gossip as a way to inform each other about the maliciousness of other agents. Thus building a distributed reputation measure.

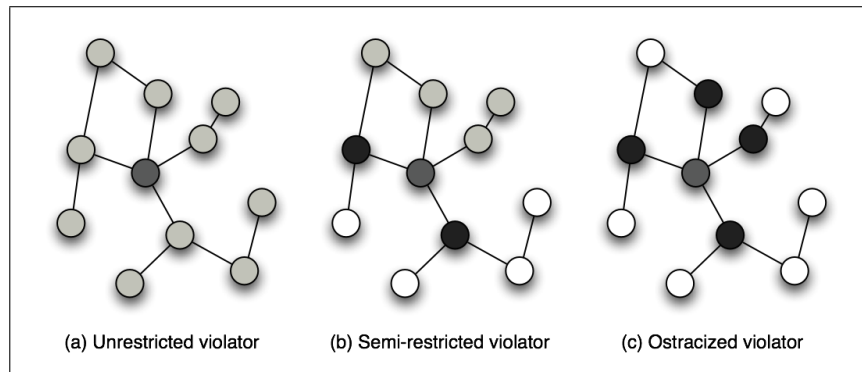


Fig. 2. Ostracizing a violator

The ostracism process is shown in Figure 2. Before a violator interacts, it is undetected (the dark gray node), and can interact with all the other agents (light gray nodes are liable to interact with the violator). When the violator interacts and violates the norm, if its partner is an enforcer, it will start blocking its interactions (black nodes are blocking agents, and white nodes are agents that the violator cannot interact with). When a violator is partially blocked, it is still able to reach part of the network. But when all the violator's neighbours block it, it is ostracised.

In order to find out information about other agents in a distributed environment, gossip between them can be used. The enforcement technique uses gossip as part of the enforcement strategy prior to ostracising agents. Since gossip should not take up too many resources, the outcome of interactions is only gossiped to the interaction mediators. If the violator agent interacts with an enforcer agent, the enforcer agent will spread this information to all mediator agents so they may block the violator in the future.

To study under which conditions the mechanism works, and give measures of its success (such as the violations received or the utility gained), a set of simulations have been run. The data extracted from them is used to support the following hypotheses:

- **H1** - Norm violations are reduced by applying a simple local blocking rule.
- **H2** - Network structure influences the enforcement capabilities.
- **H3** - The enforcement strategy used by enforcer agents can reduce the number of violations received by meek agents (*i.e.*, norm abiding agents which do not enforce the norm).
- **H4** - Enforcement makes abiding by the norm a rational strategy.

This paper is divided into five sections. Section 2 describes related work in the area of norm enforcement. Section 3 presents a detailed description of the scenario employed in the simulations. Section 4 describes the simulations and analyses the resulting data. Finally, Section 5 presents future work that will follow from this research.

2 Related Work

The problem of norm enforcement is not new. It has been dealt with in human society (also an open MAS) through the study of law, philosophy, and the social sciences. Recently studies in computer science deal with it, specially as a coordination mechanism for multi-agent systems. The application of norms from an evolutionary perspective was first studied by Axelrod in [1]. Where enforcement is seen as a meta-norm dictating that agents which do not punish violators should be punished themselves. The norm game is modelled as an N-Player Iterated Prisoner's Dilemma [1, 8]. Since the norm is specified to maximise the society's utility, agents are expected to cooperate. Enforcement techniques are sought to ensure that agents prefer cooperation. In [4, 7, 13, 16] norms are seen as a way to avoid aggression or theft. In these cases agents gain utility by collecting items that they find while they move around or by receiving them as gifts. But agents also have the ability to steal items from agents they find through aggression. A norm is added that dictates when a good is possessed by an agent. In which case it cannot be stolen by another. Therefore, a norm-abiding agent will not steal food possessed by another agent.

Two main lines of research in norm enforcement exist: sanctions and rewards to change the utilities of the game [2, 3, 8, 15], and the spread of gossip in order to avoid interaction with violators [4, 6, 7, 13, 16]. Both approaches are based on making norm adopters better off than norm violators. But there is a downside to this [4, 7], since all agents benefit from the norm while only normative agents bear the cost of enforcing it. Therefore, some agents are tempted to abide by the norm, but not to enforce it. Which makes it a recursive problem.

Norm enforcement models have been suggested in [2, 6]. They show that norm-violation becomes an irrational strategy when non-normative behaviour is punished. Nonetheless these models assume the following: (i) agents are able to monitor other agents' activities; and (ii) agents have the ability to influence the resulting utility of interactions. Assumption (i) can be brought about by having a central agent mediate all interactions as done in [2]. Another way in which agents have information of other agents is through direct interaction or gossip [4]. The first solution does not scale, since the mediator agent is the bottleneck in a large

system. The second scales, but it is less efficient since detection of all violations is not always possible, furthermore gossip is an extra cost. Assumption (ii) can be carried out through third-party enforcement [2], or self-enforcement [6]. Using a third party does not scale because the third party can easily be overwhelmed by enforcement petitions of agents. Also the third party must have access to a resource that all other agents need, and this is not always the case in real systems. Self-enforcement means that each agent takes care of those violators that affect it. Thus, all agents must have the ability to affect the outcome utility of interactions, by applying sanctions or spreading gossip.

Axelrod's mechanism for norm enforcement is based on self-enforcement and sanctions. He terms it the "shadow of the future" [1]. Defection by an agent is unlikely if it will interact often with the other agent. In which case the other agent will retaliate in future interactions. Nonetheless, this mechanism affects the utility of both agents because in the future they will both defect, and the utility will be less than if they had both cooperated. Furthermore, if the norm is to cooperate, then the enforcer is forced to violate the norm in order to retaliate, thereby becoming a violator.

Another mechanism for norm enforcement is the threat of ostracism. By avoiding interaction with violators, an agent can use the time to interact with a normative agent and achieve a higher payoff. Furthermore, violators eventually have no one with which to interact and may starve. Younger has studied [16] the possibility of avoiding interaction with norm-violators, but this is just one part of ostracism. An ostracised agent cannot interact with anyone in the society, which implies preventing it from interacting with anyone else. Human societies have used ostracism as a means to deal with norm violators [12]. In primitive societies the violator was expelled from the village and had to wander in no-man's land, or try to find another village that would take him. In modern societies, all land is owned by some state, thus violators are placed in a special area so that they cannot interact with the rest of society (e.g., prisons), but this measure has the associated cost of maintaining these areas. The electronic network in this article resembles a primitive society, an agent that has been ostracised wanders a sort of virtual no-man's land.

Emergence of norms in a structured multi-agent system has been studied in [9]. The first approach was to study regular graphs, hierarchies, and trees. This work was followed by another [5] that studied emergence in complex graphs with properties such as scale-free and small-world. Furthermore, the relationship between norm emergence and other graph parameters such as clustering factor and diameter are studied [10]. In recent work, the notion of role models has been studied and its effect in norm emergence in networks [11].

The scenario presented in this paper, is used to justify how agents can monitor other agents' activities, and how they can influence future interactions. A mix of techniques have been used to accomplish this; the spread of normative reputation through gossip, and sanctioning norm-violators by blocking their access to the network in order to achieve ostracism. Norm enforcement is studied using these techniques in societies with differing structures.

3 The Scenario

The multi-agent system in this paper is structured as a network. Thus, it is modelled as an undirected, irreflexive graph: $MAS = \langle Ag, Rel \rangle$, where Ag is the set of vertices and Rel the set of edges. Each vertex models an agent and each edge between two vertices denotes that the agents are linked. Agents can communicate through their links. Three kinds of graphs have been chosen for their significance: Tree, Random, and Small-World. A tree is a graph in which each node is linked to one parent and some number of children; only one node, the root node, has no parent, and the leaf nodes have no children. A tree has a large average distance between nodes, and no clustering. A random graph, on the other hand, does not have any regular structure. The nodes in this type of graph can be linked to any other one with a given probability. Random graphs have a small average distance between nodes, but the clustering factor is very low. Small-world graphs reside half way between regular, structured graphs, and random ones. The average distance between nodes is as small as in a random graph with the same number of nodes and edges, but its clustering factor is orders of magnitude higher [14]. The small-world graphs in the simulations have been created by starting with a regular graph², and rewiring enough random edges to make the average distance between any two vertices significantly smaller. The different graph structures have been generated to have a similar average number of links per node.

A game-theoretic approach is used to model interactions between agents. Interactions are two-player prisoner's dilemma games. Agents ought to choose the abide action given by the norm (*i.e.*, an agent disobeys the norm by choosing the violate action). An agent is capable of interacting with another if there must be a path in the graph between the two. An *initiator agent* searches for a path that leads to a *partner agent* with which to interact. The *mediator agents* are those agents in the path between the initiator and the partner. The partner finding process is explained below, but first some terms need to be formally described.

An agent's a_i neighbours are the agents it is linked to directly in the graph: $N(a_i) = \{a_j \in Ag \mid (a_i, a_j) \in Rel\}$. Each agent maintains a set of agents it blocks (an agent cannot block itself): $B(a_i) \subseteq Ag \setminus \{a_i\}$. An agent a_i can search through the network by querying other agents a_j for a list of their neighbours. Since agents are autonomous, when queried for their neighbours agent a_j can respond with any subset of its real neighbours. $RN(a_i, a_j) \subseteq N(a_j)$ are the reported neighbours a_j will return when queried by a_i . The set of reported neighbours depends on the blocking strategy of a_j . The strategies used in the simulations are explained below. A path is the route (without cycles) in the graph structure through which interaction messages are delivered. Paths are represented as finite sequences of agents $p = [a_1, a_2, \dots, a_n]$ such that for all i with $1 \leq i \leq n-1$ and $n \geq 2$ it follows that $a_{i+1} \in N(a_i)$, and for all i, j with $1 \leq i, j \leq n$ and $i \neq j$ it

² $C_{N,r}$ is a regular graph on N vertices such that vertex i is adjacent to vertices $(i+j) \bmod N$ and $(i-j) \bmod N$ for $1 \leq j \leq r$.

follows that $a_i \neq a_j$. The initiator agent will always be the first element in the path, the partner agent will be the last, while the remaining ones are mediators.

The process through which an initiator agent a_i finds a path to a partner agent a_n is as follows. First a_i creates the path $p = [a_i]$. Since an agent cannot interact with itself, the path with one agent is not valid. Then the initiator agent queries the last agent in the path (the first time it will be itself) to give it a list of its neighbours. It will choose one of the reported neighbours³ (a_j) and add it to the end of the path $p = [a_i, \dots, a_j]$. At this point the initiator can choose agent a_j as the partner, if a_j allows it. Otherwise, it can query agent a_j for its neighbours and continue searching for a partner. If the path's last element is an agent a_n that refuses to interact with the initiator agent, and a_n does not report any neighbours when queried, backtracking is applied. Agent a_n is removed and a different agent is chosen from the list of a_{n-1} 's reported neighbours and added to the end of the list.

A prisoner's dilemma game is played between the initiator and the partner, when the first has chosen the latter. Each interacting agent has complete knowledge of the game results and mediating path. Interacting agents may gossip the game results to all the mediators in the path. The information contained in *gossip* is a tuple with the agents' names and their strategy choices for the given game: $Gossip = \langle ag_i, choice_i, ag_j, choice_j \rangle$, where $choice_i$ and $choice_j$ are either *abide* or *violate*.

During the whole process agents can execute any of the following actions:

- Return a list of neighbouring agents when asked for its neighbours.
- Choose one of the agents of a list as a mediator.
- Request an agent to become the interaction partner.
- Accept or reject an invitation to interact.
- Choose a strategy to play in the PD game when interacting.
- Inform mediators of the outcome of the interaction.

The society of agents is composed of three types of agents, each one characterised by a different strategy for the actions it can execute. The *meek agent* will always abide by the norm, it will always report all its neighbours to any agent, and it will always accept an offer to interact from any agent. When searching for an interaction partner, a meek agent will request the last agent in the current path to become its partner with probability p , and with probability $1 - p$ it will ask for its neighbours⁴, and it will choose an agent randomly from the list of reported neighbours. Finally, a meek agent will not gossip the results of its interactions. A *violator agent* follows the strategy of a meek agent, except that it never abides by the norm, therefore it is not a norm-abiding agent.

Finally, *enforcer agents* have the ability to block violators, which is essential to achieve their ostracism. Enforcer agents have the same strategy of meek agents

³ To avoid loops, an agent that is already part of the path cannot be chosen again.

⁴ The value of p is set to 0.3 in all simulations. Since the path length follows a geometric distribution $Pr(L = n) = (1 - p)^{n-1}p$, the path's length expected value is $E(L) = 1/p = 3.33$ and its variance $var(L) = (1 - p)/p^2 = 7.77$. In future work we plan to relax the constraints on partner searching.

with the following exceptions: They will add agents that they know to have violated the norm to the set of blocked agents, and when they interact with a violator they gossip the interaction results to all mediators. Enforcer agents will never choose an agent in their blocked set as a partner, and will reject requests to interact from agents in their blocked set. Therefore, enforcers never interact with a violator more than once. When an agent a_i queries an enforcer agent a_m for its neighbours, if a_i is in the enforcer’s blocked set it will return an empty reply. On the other hand, if a_i is not on the blocked set, two different strategies are possible: The Uni-Directional Blockage (UDB) strategy, where all its neighbours will be returned ($RN(a_i, a_m) = N(a_m)$). And the Bi-Directional Blockage (BDB) strategy, where only those neighbours not in its blocked set are returned ($RN(a_i, a_m) = N(a_m) \setminus B(a_m)$).

Choosing one enforcement strategy over another entails a tradeoff. When the BDB strategy is chosen, violators will be more efficiently ostracized, the tradeoff is that initiator agents may also be blocked from reaching certain parts of the network, the cost is freedom. Intuitively, one can see that enforcer agents are better off with the UDB strategy. An enforcer will never interact with a violator, but it can use it as a mediator to reach other parts of the society. Meek agents, on the other hand, do not hold a memory of violating agents. Therefore, meek agents may choose violators unknowingly as their partner repeatedly. The BDB protects meek agents, by reducing the chances of them choosing violator agents.

In order to focus on the aspects such as network structure and simple blocking strategies, the following assumptions have been made to limit the number of variables:

- Agents cannot change their interaction strategy.
- Agents cannot lie when sending gossip.
- There are no corrupt enforcer agents.
- There is no noise (*i.e.*, an agent knows its opponent’s chosen strategy).

These assumptions imply that modelling an agents’ reputation is simple, and there is no redemption for violators. Since gossip is always truthful and there is no noise, the validity of information is permanent. Therefore, if there is any evidence that an agent has violated it must be a violator. Furthermore, since a violator will never change its strategy, sanctions must be indefinite. Relaxation of these assumptions will be studied in future work. Thus allowing for sophisticated violators which could trick enforcers into blocking other enforcers by giving them false information through gossip.

4 Simulations

This section shows the results of the simulations that have been run following the scenario specified in Section 3. In order to focus the experiments to see the effect of certain variables, the rest have been set with the same value for all simulations. Each simulation consists of a society of 100 agents, with 1000

rounds per simulation. In each round agents take turns to find a partner with which to interact. If an agent cannot find a partner its turn is skipped. As said before, interactions consist of the prisoner’s dilemma game specified in Figure 1.

The parameters that change in each simulation can take up the following values:

- Percentage of Violators (V) - from 10% to 90% in 10% increments.
- Percentage of Enforcers (E) - from 0% to 100% in 10% increments⁵.
- Type of Graph (G) - either tree, small world, or random.
- Enforcement Type (ET) - Uni-Directional Blockage (UDB), or Bi-Directional Blockage (BDB).

Exhaustive simulations have been run with all the possible combinations of parameter values. Each simulation is repeated 50 times in order to obtain an accurate average value. The following metrics have been extracted for each simulation: number of games played, violations received, and utility gained by an agent. The metrics have been calculated for the whole society and for each type of agent. Furthermore for each metric, both the mean and the variance have been calculated. The data gathered from the simulations support our hypotheses.

(H1) Norm violations are reduced by applying a simple local blockage rule. The different graphs in Figure 3 contain eight different lines, each one represents a different percentage of violating agents. The x -axis represent the enforcer to meek agent ratio, and the y -axis the average violations received by agents of each type. In all cases, the higher the percentage of violators, the higher the violations received by agents, which is intuitive. On the other hand, a higher ratio of enforcer to meek agents reduces the number of violations received by the society as a whole, and by norm abiding agents particularly. But this is not true for meek agents and violator agents, which receive more violations when the ratio of enforcers increases.

The improvement in the society as a whole is not significant, as seen in Figure 3(a). When just norm-abiding agent are taken into account, the reduction in violations received is much greater (see Figure 3(b)). This happens because when the enforcer ratio is high, most norm-abiding agents are enforcers. Enforcers will only interact with each violator at most one time, therefore violations received by norm-abiders are greatly reduced. Therefore, violators end up interacting with the few agents they have access to. This is the meek agents not being protected by enforcers and other violators. Both of which increase the number of violations received as the ratio of enforcers increases (see Figures 3(c) and 3(d)). Since meek agents are a small portion of the norm supporters, the fact that they receive more violations does not influence the total violations received by norm supporters as a whole. The number of games played by violator agents also supports this hypothesis. In average, violators play less games when the number of enforcer agents is high, because enforcers manage to ostracize some violators.

⁵ The percentage of meek agents is computed through the following formula: $M = 100\% - V - E$. Therefore, $V + E$ cannot be more than 100%.

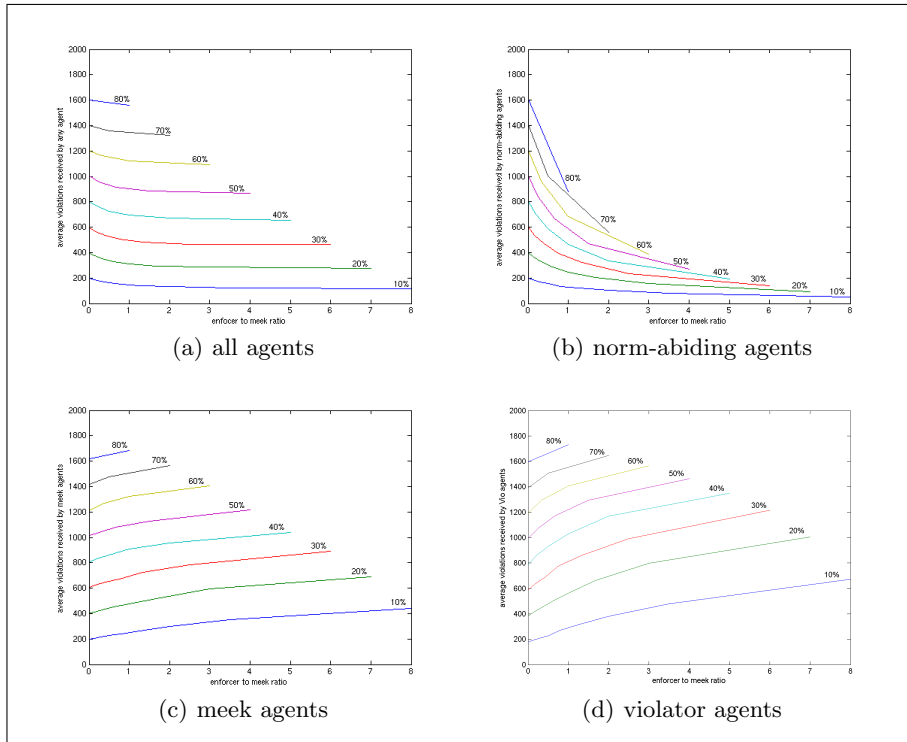


Fig. 3. Local blocking rule reduces violations

(H2) Network structure influences the enforcement capabilities.

The simulations show that different multi-agent system organisational structures have different effects on norm enforcement. Figures 4(a) and 4(b) show the average norm violations (y -axes) for each of the different structures tested: Random, Small World, and Tree. They represent the simulations where violator agents account for 10% and 20% of the population respectively. Therefore, at most there will be 90% or 80% of enforcers, respectively. The x -axes plots the percentage of enforcer agents. Both random and small world networks have an almost identical graph line. The tree structure has an altogether different graph line which greatly improves the enforcement capabilities. The fact that in a tree there is only one path between any two agents is the determining factor in making the society more secure to violations. In random and small world graphs, many paths can be usually found between any two agents. From the simulations it is deduced that the higher the number of paths that unite agents, the more vulnerable they are to non-normative attacks. On the other hand, the main difference between small world graphs and random graphs is their clustering coefficient. Since the two types of graphs have very similar results, the clustering coefficient can be ruled out from the variables that have an impact in norm-enforcement.

As an interesting side note, the tendency is that the more enforcer agents, the less violations. But in random and small world networks, when the percentage of enforcer agents reaches its maximum the percentage of violations received are increased (see Figure 4(b)). This happens because in both these networks violator agents manage to find paths that link them. Since at this point there are few meek agents for them to prey on, they are forced to interact with each other. Figures 3(c) and 3(d) show that the number of violations received by meek and violator agents increases with higher enforcer ratio. In an interaction between two violator agents, two violations are being accounted for and the average of violations is increased. A sub-society of violating agents is formed. This has been observed in all simulations with a ratio of violator agents of 20% and above, but it is not so acute in the simulations with 10% of violators (see Figure 4(a)). When the ratio of violator agents is low enough, enforcers manage to ostracise more of them, and they cannot interact with each other. When this happens, no sub-society of violating agents exists, they are completely blocked.

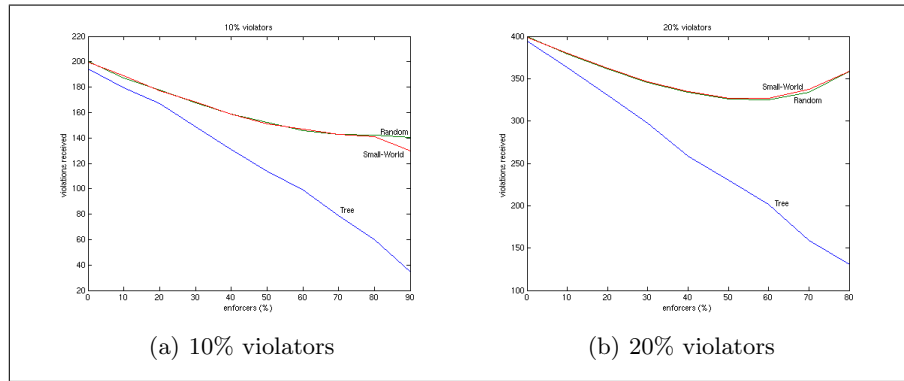


Fig. 4. Enforcement capabilities vary depending on structure

(H3) The enforcement strategy used by enforcer agents can reduce the number of violations received by meek agents. The x -axes in Figure 5(a) shows the enforcer to meek agent ratio. A higher ratio implies more enforcer agents. The y -axes measures the increment in efficiency at protecting meek agents from violations. Efficiency is calculated as the increment in percentage of the violations received by meek agents when enforcers use uni-directional blockage over bi-directional blockage (see Equation 1). A positive efficiency value means that BDB managed to stop more violations than UDB.

$$\Delta Efficiency = ((Violations_{UDB} / Violations_{BDB}) - 1) \times 100 \quad (1)$$

In Figure 5(a) it can be observed that in random and small world networks the efficiency is always positive for any enforcer to meek agent ratio. It is also observed that for low ratio values the efficiency is increasing. But after a rate

of 3 enforcers per meek agent the efficiency hits a ceiling. The results show that Bi-Directional Blockage has a higher efficiency at protecting meek agents from violator agents in these cases. The case of the tree network is different. The efficiency increment stays along the 0% line with some deviations. In networks organized as trees, the choice of enforcement strategy does not have a significant influence in the outcome. The reason being that the tree network ostracises violators quickly, independently of the blockage strategy used.

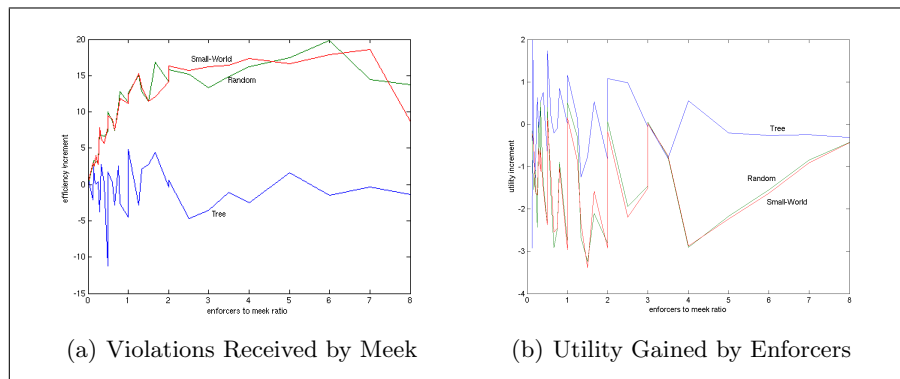


Fig. 5. Enforcement strategy influences outcomes

The gain in efficiency at guarding meek agents comes at a cost. When enforcers use the BDB strategy they can ostracise themselves. This is the case when the enforcer is completely surrounded by violators. If the enforcer uses the UDB strategy it will use its neighbours as mediators, independently of their type. But when using the BDB strategy an enforcer is not be able to do this and therefore it cannot interact with anyone. Thus ostracising itself. This is a rare case but it can reduce the average utility gained by enforcer agents by up to 3% (see Figure 5(b)). The metric used to calculate the difference in utility can be seen in equation 2. A negative number means that the agent gains more utility when using a UDB strategy.

$$\Delta Utility = ((Utility_{UDB} / Utility_{BDB}) - 1) \times 100 \quad (2)$$

(H4) Enforcement makes supporting the norm a rational strategy.

The simulation data that refers to the utility gained by agents has been used to support this hypothesis. In the context of this paper, a strategy is said to be rational if the the agent will maximize the utility gained in the game. What has been tested is whether following the norm maximizes the agent's utility, and in which conditions. The simulation data has shown that when the ratio of enforcers passes a certain threshold, norm-abiding agents will gain more utility than norm-violating ones. This threshold depends on the amount of violating agents in the system. In a society with 10% of violator agents, five enforcers are

needed for every four meek agents to make supporting the norm the rational strategy. For a society with 50% of violator agents, the ratio needs to be higher than 0.7 enforcers for each meek agent. The rest of simulations have inflection points between those two values. Strangely, societies with higher percentage of violators need a smaller ratio of enforcers to meek agents.

Finding a partner is a random walk through the network. Therefore, when a violator searches for a partner in a society where they are the majority, chances are that it will interact with another violator. When two violators interact they get a very low utility, thus a small number of enforcers interacting amongst themselves can easily win more utility than the violators, and even make up for the meek agents which are being preyed upon.

Figure 6(a) and 6(b) show the utility gained (y -axes) by norm supporting agents and violators respectively. Their x -axes show the enforcer to meek agent ratio. Each of the lines in the figures represent simulations with different percentage of agents in the society. As the number of enforcers increases norm supporters gain more utility. The opposite effect is observed for violator agents. When the enforcer to meek agent ratio is low, the utility gained by violator agents is much higher than the one gained by norm supporters. As the number of enforcer agents grows the values are reversed. The inflection point depends on the percentage of violator agents.

Interestingly, even though meek agents receive more violations as the rate of enforcer agents grows (see Figure 3(c)), the utility gained by them is not decreased. It surprisingly increases (see Figure 6(c)). This is due to the fact that meek agents are still able to interact with other norm supporters. Enforcer agents will never interact with violators more than once, so they are restricted to interacting with other enforcers or meek agents. This helpful interaction from enforcers makes their utility increase despite receiving more violations. The ratio of violations to normative actions is lowered and the utility is increased.

5 Future Work

This paper is part of ongoing research on norm enforcement. Many other variants of this model will be simulated. In future work the set of assumptions about agents will be relaxed, by giving them the ability to change their strategies, to lie, and to allow enforcer agents to violate the norm (*i.e.*, corrupt enforcers). The perfect information assumption will be relaxed by adding uncertainty and noise. In these cases enforcer agents will need elaborate gossip techniques and reputation management to allow them to pick the right targets for enforcement. Furthermore, the agent's reputation can also be modelled by having the interaction mediators overhear the conversations they mediate. If overhearing is possible, there is no need to wait for interacting agents to report the interaction outcome. More so, other conservative blocking strategies can be studied; such as blocking off agents that mediate norm violators, or blocking agents until they are shown to be norm-abiders by interacting with the mediators.

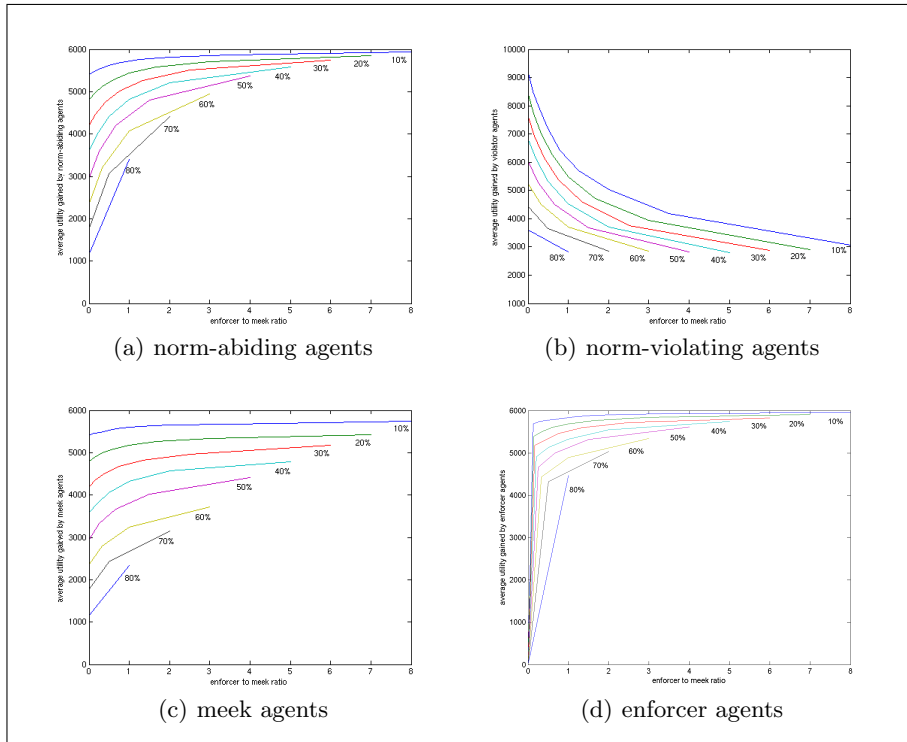


Fig. 6. Utility gained by agents

The process an agent uses to search for a partner can influence the utility it gains. All agents in the simulations searched for partners using the same procedure, which was random. The process could be modified in many ways. For instance, the probability with which an agent is chosen as the interaction partner can be modified to make the average path length longer or shorter, this modification could be for all agents, or they could each have different average path length. Also, agents could choose an agent as the partner first and then try to find a path to it. These changes are not mutually exclusive, many combinations could be studied.

The impact of other network parameters (*e.g.*, clustering factor, diameter, number of links per agent, number of paths between agents) on norm enforcement should also be studied in future work. It has been shown that a tree network is better from an enforcement perspective. Further studies could also relax the assumption of fixed networks. In order to find realistic models to be used in real networks, dynamic links must be allowed. Links could be added between agents dynamically and test how it affects norm enforcement. New enforcement techniques should be used that take advantage of the dynamic nature of the network.

Finally, related work has shown that when enforcement conveys a cost to the agent, the efficiency of enforcement diminishes [1, 8]. The scenario in this paper does not consider such a cost associated to blocking violators. A cost could be associated to blocking interactions in order to test the enforcing efficacy in such a scenario. Enforcers would bear the cost of enforcement if they were not able to reject interactions. In such a case they would spare other agents from receiving norm-violations by receiving them themselves.

All these scenario modifications can also be used to research into the necessary conditions for norm emergence. Our goal is to find ways to apply this work to more realistic scenarios, such as security from malicious agents in open multi-agent systems over the internet.

Acknowledgements

This research has been partially supported by the Generalitat de Catalunya under the grant 2005-SGR-00093, and the OpenKnowledge⁶ Specific Targeted Research Project (STREP), which is funded by the European Commission under contract number FP6-027253. The OpenKnowledge STREP comprises the Universities of Edinburgh, Southampton, and Trento, the Open University, the Free University of Amsterdam, and the Spanish National Research Council (CSIC).

A. Perreau de Pinninck is supported by a CSIC predoctoral fellowship under the I3P program, which is partially funded by the European Social Fund. M. Schorlemmer is supported by a *Ramón y Cajal* research fellowship from Spain's Ministry of Education and Science, which is also partially funded by the European Social Fund.

References

1. Robert Axelrod. An evolutionary approach to norms. *The American Political Science Review*, 80:1095–1111, 1986.
2. Guido Boella and Leendert van der Torre. Enforceable social laws. In *AAMAS '05: Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 682–689, 2005.
3. Jeffrey Carpenter, Peter Matthews, and Okomboli Ong'ong'a. Why punish: Social reciprocity and the enforcement of prosocial norms. *Journal of Evolutionary Economics*, 14(4):407–429, 2004.
4. Cristiano Castelfranchi, Rosaria Conte, and Mario Paolucci. Normative reputation and the costs of compliance. *Journal of Artificial Societies and Social Simulation*, 1(3), 1998.
5. Jordi Delgado. Emergence of social conventions in complex networks. *Artificial Intelligence*, 141(1):171–185, 2002.
6. Amandine Grizard, Laurent Vercouter, Tiberiu Stratulat, and Guillaume Muller. A peer-to-peer normative system to achieve social order. In *AAMAS '06 Workshop on Coordination, Organization, Institutions and Norms in agent systems (COIN)*, 2006.

⁶ <http://www.openk.org>

7. David Hales. Group reputation supports beneficent norms. *Journal of Artificial Societies and Social Simulation*, 5(4), 2002.
8. Douglas D. Heckathorn. Collective sanctions and compliance norms: a formal theory of group-mediated social control. *American Sociological Review*, 55(3):366–384, 1990.
9. James E. Kittock. The impact of locality and authority on emergent conventions: initial observations. In *AAAI '94: Proceedings of the Twelfth National Conference on Artificial Intelligence*, volume 1, pages 420–425, Menlo Park, CA, USA, 1994. American Association for Artificial Intelligence.
10. Josep M. Pujol, Jordi Delgado, Ramon Sangüesa, and Andreas Flache. The role of clustering on the emergence of efficient social conventions. In *IJCAI '05: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, pages 965–970, 2005.
11. Bastin T.R. Savarimuthu, Maryam Purvis, Stephen Cranefield, and Martin Purvis. Role model based mechanism for norm emergence in artificial agent societies. In *Proceedings of the International Workshop on Coordination, Organization, Institutions, and Norms (COIN), Honolulu, Hawai'i, USA, 2007*.
12. Michael Taylor. *Community, Anarchy & Liberty*. Cambridge University Press, 1982.
13. Adam Walker and Michael Wooldridge. Understanding the emergence of conventions in multi-agent systems. In Victor Lesser, editor, *Proceedings of the First International Conference on Multi-Agent Systems*, pages 384–389, San Francisco, CA, 1995. MIT Press.
14. Duncan J. Watts and Steven H. Strogatz. Collective dynamics of small-world networks. *Nature*, (393):440–442, 1998.
15. Fabiola López y López, Michael Luck, and Mark d'Inverno. Constraining autonomy through norms. In *AAMAS '02: Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 674–681, New York, NY, USA, 2002. ACM Press.
16. Stephen Younger. Reciprocity, sanctions, and the development of mutual obligation in egalitarian societies. *Journal of Artificial Societies and Social Simulation*, 8(2), 2005.