Overcoming Omniscience for Norm Emergence in Axelrod's Metanorm Model

Samhar Mahmoud[1], Nathan Griffiths[2], Jeroen Keppens[1], and Michael Luck[1]

[1] Department of Informatics
King's College London
London WC2R 2LS, UK.
samhar.mahmoud@kcl.ac.uk
[2] Department of Computer Science
University of Warwick
Coventry CV4 7AL, UK.

Abstract. Norms are a valuable mechanism for establishing coherent cooperative behaviour in decentralised systems in which no central authority exists. In this context, Axelrod's seminal model of norm establishment in populations of self-interested individuals [2] is important in providing insight into the mechanisms needed to support this. However, Axelrod's model suffers from significant limitations: it adopts an evolutionary approach, and assumes that information is available to all agents in the system. In particular, the model assumes that the private strategies of individuals are available to others, and that agents are omniscient in being aware of all norm violations and punishments. Because this is an unreasonable expectation, the approach does not lend itself to modelling real-world systems such as peer-to-peer networks. In response, this paper proposes alternatives to Axelrod's model, by replacing the evolutionary approach, enabling agents to learn, and by restricting the metapunishment of agents to only those where the original defection is perceived, in order to be able to apply the model to real-world domains.

1 Introduction

In many application domains, engineers of distributed systems may choose, or be required, to adopt an architecture in which there is no central authority and the overall system consists solely of self-interested autonomous agents. The rationale for doing so can range from efficiency reasons to privacy requirements. In order for such systems to achieve their objectives, it may nevertheless be necessary for the behaviour of the constituent agents to adhere to certain constraints, or *norms*. In peer-to-peer file sharing networks, for example, we require (at least a proportion of) peers to provide files in response to the requests of others, while in wireless sensor networks nodes must share information with others for the system to determine global properties of the environment. However, there is typically a temptation in such settings for individuals to deviate from the desired behaviour. For example, to save bandwidth peers may not provide files, and to

conserve energy the nodes in a sensor network may not share information. It is therefore desirable to minimise the temptation for agents to deviate from the desired behaviour, and encourage the emergence of cooperative norms.

Norms have been studied by very many different researchers, over several different areas (for example, [6–8, 16, 18–20, 23]. Most notably, Axelrod's seminal investigation of norm establishment in populations of self-interested individuals [2] provides an analysis of the conditions in which norms can be established. In his experiments, a population of agents repeatedly play a simple game, in which agents make decisions about whether to comply with a desired norm of cooperation and whether to punish those who are seen to violate this norm. These decisions may result in certain penalties or rewards, with the strategies of agents being determined through an evolutionary process, in which the more successful strategies are reproduced. In this setting, Axelrod explored how the emergence of norm compliant strategies can be encouraged.

Although Axelrod's investigation is successful in establishing cooperative norms, the model makes several assumptions that are unrealistic in real-world settings. In particular, in many domains it is not possible to remove unsuccessful agents and replicate those that are more successful, and there is no centralised control that could oversee this process. Instead, we need a mechanism through which individuals can learn to improve their strategies over time. If we enable individuals to compare themselves to others, and adopt more successful strategies, then we can take a *learning interpretation* of the evolutionary mechanism [13], without needing to remove and replicate individuals. However, this learning interpretation requires that the private strategies of individuals are available for observation by other agents, which is again an unreasonable assumption. Furthermore, as has been shown elsewhere, Axelrod's model is unable to sustain cooperation over a large number of generations [10]. Axelrod's approach, as discussed below, relies on agents being able to punish both those that defect and those that fail to punish defection, yet this is unrealistic since it assumes *omniscience* through agents being aware of all norm violations and punishments.

In this paper we investigate alternatives that allow us to make use of the mechanisms resulting from Axelrod's investigations, in more realistic settings. Specifically, we first take a learning interpretation of evolution and describe an alternative technique, strategy copying, which prevents norm collapse in the long term. Second, we remove the assumption of omniscience and constrain the ability of agents to punish according to the defections they have observed. Finally, to obviate the need for information on the private strategies of others, we propose a learning algorithm through which individuals improve their strategies based on their experience.

The paper begins by reviewing Axelrod's original norms game and metanorms game, in which our work is situated. Then, in Section 3, we present our strategy copying technique, and show how it performs in the original context and in situations in which observation of defection is not guaranteed. In Section 4, we describe a reinforcement learning algorithm designed to avoid the need for access to the private strategies of others. Section 5, considers related work, before

presenting our conclusions in Section 6, with a discussion of the significance of our results.

2 Axelrod's Model

2.1 The Norms Game

Axelrod's *norms game* adopts an evolutionary approach in which successful strategies multiply over generations, potentially leading to convergence on cooperative norms [2]. Each agent in the population has a number of opportunities (o) in which it can choose to *defect* by violating a norm, and such behaviour has a particular known probability of being observed, or *seen* (S_o). An agent i has two decisions, or strategy dimensions, as follows. First, it must decide whether to defect, determined by its *boldness* (B_i); and second, if it sees another agent defect in a particular opportunity (with probability S_o) it must decide whether to punish this defecting agent, determined by its *vengefulness* (V_i), which is the probability of doing so. If $S_o < B_i$ then i defects, receiving a *temptation payoff*, $T = 3$, while *hurting* all other agents with payoff $H = -1$. If a defector is *punished* (P), it receives an additional punishment payoff of $P = -9$, while the punishing agent pays an *enforcement cost*, $E = -2$. The initial values of B_i and V_i are chosen at random from a uniform distribution of a range of 8 values between $\frac{0}{7}$ and $\frac{7}{7}$.

Axelrod's simulation had 20 agents, with each having four opportunities to defect, and the chance of being seen for each drawn from a uniform distribution between 0 and 1. After playing a full round (all four opportunities), scores for each agent are calculated to produce a new generation, as follows. Agents that score better or equal to the average population score plus one standard deviation are reproduced twice in the new generation. Agents that score one standard deviation or more under the average score are not reproduced, and all others are reproduced once. Finally, a mutation operator is used to enable new strategies to arise. Since B_i and V_i (which determine agent behaviour) take eight possible values they can be represented by three bits, to which mutation is applied (by flipping a bit) when an agent is reproduced, with a 1% chance.

In this model, cooperative norms are established when V_i is high and B_i is low for all members of the population, so that defection is unlikely, and observed defections are likely to be punished. In 100 generations, Axelrod found only partial establishment of a norm against defection, so introduced an additional mechanism to support norms in his *metanorm* model.

2.2 The Metanorms Game

The key idea underlying Axelrod's metanorm mechanism is that some further encouragement for enforcing a norm is needed. In the *metanorms* game, if an agent sees a defection but does not punish it, this is itself considered as a form of defection, and others in turn may observe this defection (with probability

S_o) and apply a punishment to the non-enforcing agent. As before, the decision to punish is based on vengefulness, and brings the defector (namely, the non-punisher) a punishment cost of $P' = -9$ and the punisher an enforcement cost of $E' = -2$. Applying the simulation to the metanorms game gives runs with high vengefulness and low boldness, which is exactly the kind of behaviour needed to support the establishment of a norm against defection.

However, Axelrod's analysis of results was limited. As has been shown subsequently, allowing Axelrod's *metanorms game* to run for an extended period (1,000,000 generations) ultimately results in norm collapse [9]. As Mahmoud et al. have shown [10], this norm collapse arises as a consequence of two aspects. First, a sufficiently long run (compared to Axelrod's limited run of 100 generations) provides the opportunity for a sequence of mutations to cause norm collapse even after a norm has been established in the population. Second, such mutation is magnified by the evolutionary manner of replication, generating a new population of agents.

3 Strategy Copying

As indicated above, the evolutionary approach causes some problems in extended runs, leading to norm collapse. In addition, for use in domains such as peer-to-peer or wireless sensor networks, the agents themselves cannot be deleted or replicated, but instead must modify their own behaviour. In this section, therefore, we examine a simple alternative to Axelrod's model in which an agent that performs poorly in comparison to others in the population can *learn* new strategies (in terms of vengefulness and boldness attributes) by adopting the strategy of other, better performing agents, replacing the existing strategy with a new one. Agents can achieve this in different ways: they can copy the strategy of the agent with the highest score or they can copy the strategy of one of the group of agents that perform best in the population.

3.1 Strategy Copying from a Single Agent

Intuitively, copying the strategy of the agent with the highest score appears to be a promising approach. However, it leads to poor results in the long term because it draws strategies from only one agent rather than a population of agents. This makes the approach vulnerable to strategies that are only successful in a small number of possible settings. Moreover, by failing to draw strategies from a variety of agents, the strategies tend to converge prematurely. To illustrate, consider a group of students taking an examination, with one of the students having cheated. If the cheating student has not been seen, they may achieve the best exam performance. However, if all other students copy this behaviour and cheat in the next exam, there is a high possibility that they will be caught, and will thus suffer from much worse results than if they had not cheated. This is supported by the results shown in Figures 1 and 2, illustrating experiments with runs of 100 and 1,000,000 timesteps (where a timestep represents one *round* of agents

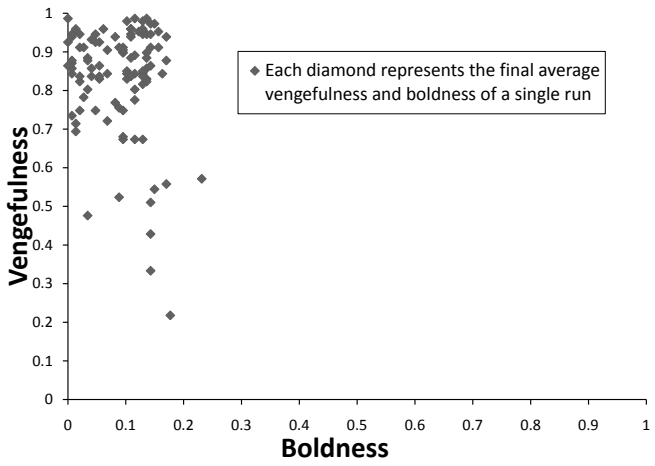

Fig. 1. Strategy copying from the best agent; 100 timesteps

having opportunities to defect and learning from the results, and is equivalent to a *generation* in the evolutionary approach). Each point on the graph (shown as a diamond to increase visibility) represents the average boldness and vengefulness of the population at the end of a single simulation run.

In the short term, as can be seen from Figure 1, copying from the best agent leads to norm establishment. However, in the long term the norm collapses, as shown in Figure 2. This can be explained by the fact that an agent with low vengefulness that does not punish a defector (and thus does not pay an enforcement cost) but is also not metapunished, scores better than any other agent with high vengefulness that does punish (and thus pays the enforcement cost). As a result, other agents copy the low vengefulness of this agent so that low vengefulness becomes prevalent in the population. In the same way, when low vengefulness prevails in the population, an agent with high boldness defects, gaining a *temptation payoff*, and hurting others without receiving punishment. As a result, other agents copy the high boldness of this agent so that low vengefulness and high boldness are propagated through the population, leading to norm collapse. This transition from high vengefulness to low vengefulness and from low boldness to high boldness requires time to manifest, but the duration of the period of time is not fixed.

3.2 Strategy Copying from a Group of Agents

Alternatively, and as we have suggested, we might seek to copy the strategy of one in a group of high-performing agents. In this view, agents choose one agent, at random, from the group of agents with scores above the average, and copy its strategy. As previously, experiments of different durations (between 100 and 1,000,000 timesteps) were carried out; the results in Figure 3, for 1,000,000

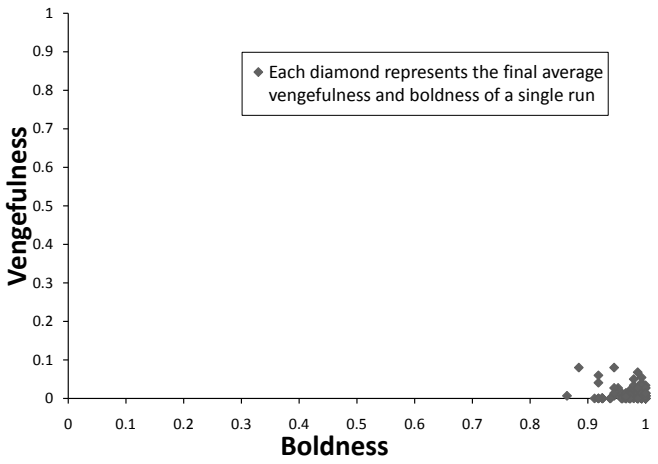

Fig. 2. Strategy copying from the best agent; 1,000,000 timesteps

timesteps, show that all runs ended with norm establishment in the long term, indicating that this approach is effective in eliminating the problematic effect of the replication method. This approach avoids norm collapse since it does not limit itself to the best performing agent, and thus does not run the risk of only adopting a strategy that performs well in a small number of settings.

3.3 Observation of Defection

As stated in Section 2, in Axelrod's model, an agent Z is able to punish another agent Y that does not punish a defector X, even though agent Z did not see the defection of agent X. However, such metapunishment is not possible if the original defection is not observed: guaranteed observation of the original defection is an unreasonable expectation in real-world settings. In consequence, our model needs adjustment so that metapunishment is only permitted if an agent observes the original defection. However, because this observation constraint limits the circumstances in which metapunishment is possible, its introduction corresponds to removing the metapunishment component from part of the game. In Axelrod's original experiments, metapunishment was introduced as a means to stabilise an established norm. In his setting, norms tend to collapse shortly after they are established without metapunishment. In fact, this remains the case in our model and our results confirm this.

More precisely, the observation constraint causes all runs to end in norm collapse when simulations are run for 1,000,000 timesteps, as shown in Figure 4. This is due to the fact that, as in the original model, runs initially stabilise on high vengefulness and low boldness, and then mutation causes vengefulness to reduce. If an agent Y with high vengefulness and low boldness changes through mutation to give lower vengefulness, while boldness for all remains low, there

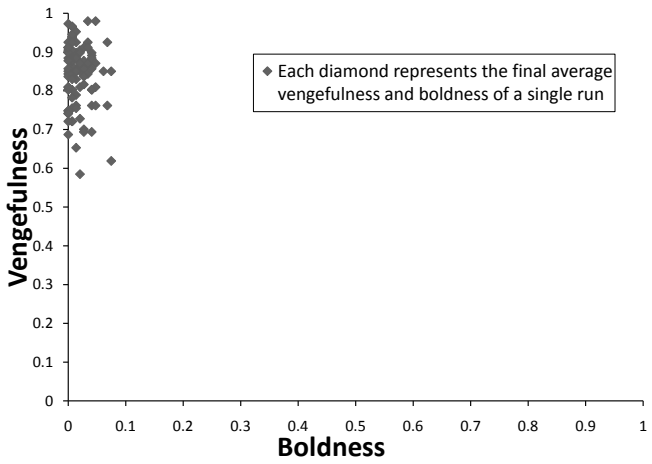

Fig. 3. Strategy copying from a group of agents

is no defection and the mutated agent survives. In addition, if boldness then mutates to become just a little higher for a different agent X, with average vengefulness remaining high, X will still rarely defect because of relatively low boldness.

If it *does* defect, however, and *is* seen by others, it receives a low score, unless it is *not* punished, in which case the non-punishing agents may themselves be punished because of the high vengefulness in the general population. Here, agent Y may not punish X because of the low probability of being seen (which must be below the low boldness level to have caused a defection) or because it has mutated to have lower vengefulness. In the former case, Y will not be metapunished for non-punishment (since there is a low probability of some other agent Z having seen it), but in the latter case, Y might be metapunished if it is seen by others. The likelihood of agent Y's non-punishment being seen requires first X's defection being seen by Y, and then Y's non-punishment being seen by others. Importantly, in this new model, agents that metapunish Y must themselves see X's defection. Since this combination of requirements is rare, such mutants survive for a longer duration, enabling their strategy to propagate through the population, and causing vengefulness to decrease. In addition, if another such event occurs, it will cause vengefulness to drop further until it reaches a very low level. When the model runs over an extended period, such a sequence of events is much more likely, and low vengefulness allows a mutant of higher boldness to survive and spread among the whole population, which is the cause of the norm collapse.

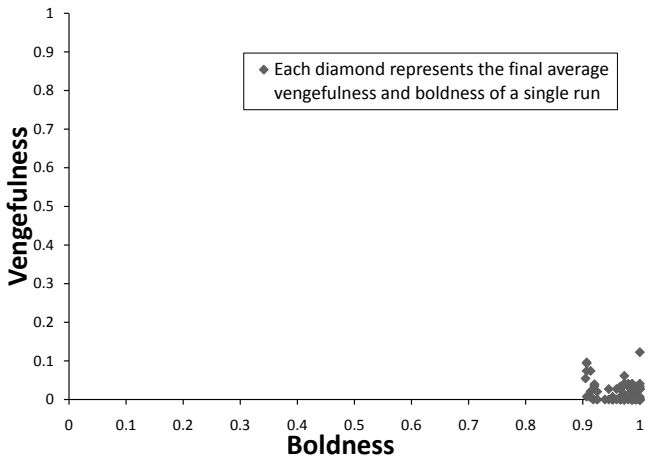

Fig. 4. Strategy copying with observation of defection

4 Strategy Improvement

Once the observation constraint is introduced, strategy copying becomes inadequate. Furthermore, it requires that agents have access to the strategies and decision outcomes of others in order to enable the copying mechanism. As we have argued, in real-world settings such observations tend to be unrealistic. *Reinforcement learning* offers an alternative to Axelrod's evolutionary approach to improving performance of the society while keeping agent strategies and decision outcomes private. There are many reinforcement techniques in the literature, such as Q-learning [21], PHC and WOLF-PHC [4], which we use as inspiration in developing a learning algorithm for strategy improvement in the metanorms game.

4.1 Q-learning

Q-learning is a reinforcement learning technique that allows the learner to use the (positive or negative) reward, gained from taking a certain action in a certain state, in deciding which action to take in the future in the same state. Here, the learner keeps track of a table of Q-values that record an action's quality in a particular state, and updates the corresponding Q-value for that state after each action. The new value is a function of the old Q-value, the reward received, and a learning rate, δ, and the action with the highest updated Q-value for the current state is chosen. However, for us, Q-learning suffers from two drawbacks. First, it considers an agent's past decisions and corresponding rewards, which are not relevant here; doing so would inhibit an agent's ability to adapt to new circumstances. Second, actions are precisely determined by the Q-value; there is no probability of action, unlike Axelrod's model.

Bowling and Veloso [4] proposed policy hill climbing (PHC), an extension of Q-learning that addresses this latter limitation. In PHC, each action has a probability of execution in a certain state, determining whether to take the action. Here, the probability of the action with the highest Q-value is increased according to a learning rate δ, while the probabilities of all other actions are decreased in a way that maintains the probability distribution, with each probability update occurring immediately after the action. In enhancing the algorithm, a *variable* learning rate is introduced, which changes according to whether the learner is winning or losing, inspired by the WOLF technique (win or learn fast). This suggests two possible values for δ: a low one to be used while an agent is performing well and a high one to be used while the agent is performing poorly.

However, in one round of Axelrod's game, an agent can perform multiple punishments (potentially one per defection and non-punishment observed), while only having a small number of opportunities to defect (four in Axelrod's configuration). Therefore, punishment and metapunishment actions would be considered much more frequently than defection, leading to disproportionate update of probabilities of actions, with some converging more quickly than others. To address this imbalance, we can restrict learning updates to occur only at the end of each round, rather than after each individual action, so that boldness and vengefulness are reconsidered once in each round and evolve at the same speed. The aim here is to change the probability of action significantly when losing, while changing it much less when winning, providing more opportunities to adapt to good performance.

While basic Q-learning is not appropriate because of the lack of a probability of taking action, PHC-WOLF suffers from a disproportionate update of probabilities of action. Nevertheless, the use of the variable learning rate approach in PHC-WOLF is valuable in providing a means of updating the boldness and vengefulness values in determining which action to take. However, since agents that perform well need not change strategy, we can consider only one learning rate. The next section details our algorithm, inspired by this approach.

4.2 BV Learning

To address the concerns raised above, in this section, we introduce our BV learning algorithm. This requires an understanding of the relevant agent actions and their effect on boldness and vengefulness, as summarised in Table 1, which outlines the different actions available to an agent and the consequences of each on the agent's score.

Now, since boldness is responsible for defecting, an agent that obtains a good score as a result of defecting should increase its boldness, and an agent that finds defection detrimental to its performance should decrease its boldness. Learning suitable values for vengefulness is more complicated, since while it is responsible for both punishment and metapunishment, these also cause enforcement costs that decrease an agent's score. Low vengefulness allows an agent to avoid paying an enforcement cost, but can result in receiving metapunishment. Vengefulness thus requires a consideration of all these aspects. This intuition is formalised as

Table 1. Effects of decisions on score

Decision	**Effects**
Defect	Gain temptation payoff Hurts all other agents Potentially suffer punishment cost
Cooperate	—
Punish	Punisher pays enforcement cost Defector pays punishment cost
Not punish	Potentially suffer metapunishment (incurring punishment cost)
Metapunish	Punisher pays enforcement cost Defector pays punishment cost
Not metapunish	—

in Algorithm 1, as follows. (Note that we use subscripts to indicate the relevant agent only when needed.)

First, in order to determine the unique effect of each individual action on agent performance, note that we decompose the single combined total score (TS) of the original model into distinct components, each reflecting the effect of different classes of actions. The defection-cooperation action brings about a change only if an agent defects (Line 9): the agent's score increases by a *temptation payoff*, T (Line 10), but it *hurts* all others in the population, whose scores decrease by H (line 12), where H is a negative number that is thus added to the score. If an agent cooperates, no scores change. We can therefore use just one distinct value to keep track of this score, referred to as the *defection score* (DS), and which determines whether to increase or decrease boldness.

Conversely, punishment and metapunishment both have two-sided consequences: if an agent j sees agent i defect in one of its opportunities (o) to do so, with probability S_o (Line 13), and decides to punish it (which it does with probability V_j; Line 14), i incurs a punishment cost, P, to its DS (Line 15), while the punishing agent incurs an enforcement cost, E, to a different score, its *punishment score*, PS (Line 16). Note that both P and E are negative values, so they are added to the total when determining an overall value. As the name suggests, PS captures the total score obtained by an agent as a result of punishing another, and applies to both punishment and metapunishment (enforcement costs). There is also a different change (resulting from potential subsequent received metapunishment) if it decides not to punish (Line 17). If j does not punish i, and another agent k sees this in the same way as previously (Line 19), and decides to metapunish (Line 20), then k incurs an enforcement cost, E, to its PS, and j incurs a punishment cost P to its *no punishment score*, NPS. (An agent's NPS is obtained from not punishing, and comprises the metapunishment cost alone.)

In Axelrod's original model, those agents that are one standard deviation or more below the mean are eliminated and replaced in the subsequent population

Algorithm 1 The Simulation Control Loop: *simulation*$(T, H, P, E, \gamma, \delta)$

for each agent i **do**
  {***Initialising***}
  $B_i = random()$ {**Random generator that uses uniform distribution**}
  $V_i = random()$ {**Random generator that uses uniform distribution**}
for each round **do**
  for each agent i **do**
    {***Decision making***}
    for each opportunity to defect o **do**
      if $B_i > S_o$ **then**
        $DS_i = DS_i + T$
        for each agent j: $j \neq i$ **do**
          $TS_j = TS_j + H$
          if see(j,i,S_o) **then**
            if punish (j, i, V_j) **then**
              $DS_i = DS_i + P$
              $PS_j = PS_j + E$
            else
              for each agent $k : k \neq i \wedge k \neq j$ **do**
                if see(k,j,S_o) **then**
                  if punish (k, j, V_k) **then**
                    $PS_k = PS_k + E$
                    $NPS_j = NPS_j + P$
  $Temp = 0$
  for each agent i **do**
    $TS_i = TS_i + DS_i + PS_i + NPS_i$
    $Temp = Temp + TS_i$
  $AvgS = Temp/no\_agents$
  for each agent i **do**
    {***Learning***}
    if $TS_i < AvgS$ **then** {$AvgS$ **is the mean score of all agents**}
      if explore(γ) **then**
        $B_i = random()$
        $V_i = random()$
        if $DS_i < 0$ **then**
          $B_i = max(B_i - \delta, 0)$
        else
          $B_i = min(B_i + \delta, 1)$
        if $PS_i < NPS_i$ **then**
          $V_i = max(V_i - \delta, 0)$
        else
          $V_i = min(V_i + \delta, 1)$

generation with new agents following the strategy captured by the boldness and vengefulness values of those agents that are one standard deviation or more above the mean. Thus, poorly performing agents are replaced by those that perform much better. In contrast, in our model, we distinguish more simply between good and poor performance, with only agents that score below the mean reconsidering their strategy. Thus, for each agent, we combine the various component scores into a total, TS and, if the agent is performing poorly (in relation to the average score, $AvgS$ in Line 30), we reconsider its boldness and vengefulness. Note that this average score is established through the lines in the algorithm around 27.

Now, in order to ensure we allow a degree of exploration (similar to mutation in the original model's evolutionary approach, to provide comparability) and to enable an agent to step out of the learning trend, here we adopt an *exploration rate*, γ, which regulates adoption of random strategies from the available strategies universe (Line 31). If the agent does not explore then, if defection is the cause of a low score (Line 34), an agent decreases its boldness, and increases it otherwise. Similarly, agents increase their vengefulness if they find that the effect of not punishing is worse than the effect of punishing (Line 38), and decrease vengefulness if the situation is reversed. As both PS and NPS represent the result of two mutually exclusive actions, their difference for a particular agent determines the change to be applied to vengefulness. For example, if $PS > NPS$, then punishment has some value, and vengefulness should be increased.

Finally, given a decision on whether to modify an agent's strategy, the degree of the change, or *learning rate* (δ), must also be considered. Since vengefulness and boldness have eight possible values from $\frac{0}{7}$ to $\frac{7}{7}$, we adopt the conservative approach of increasing or decreasing by one level at each point, corresponding to a learning rate of $\delta = \frac{1}{7}$. Thus, an agent with boldness of $\frac{5}{7}$ and vengefulness of $\frac{3}{7}$ that decides to defect less and punish more will decrease its boldness to $\frac{4}{7}$ and increase its vengefulness to $\frac{4}{7}$.

4.3 Evaluation

The algorithm is designed to mimic the behaviour of Axelrod's evolutionary approach as much as possible, while relaxing Axelrod's unrealistic assumptions. This allows us to replicate Axelrod's results and investigate his approach in more realistic problem domains. The analysis of a sample run reveals that agents with low vengefulness and agents with high boldness start changing their strategies. Here, agents with high boldness defect frequently, and are punished as a result, leading to a very low DS, in turn causing these agents to decrease their boldness. Agents with low vengefulness do not punish and are consequently frequently metapunished; as a result, their PS is much better (lower in magnitude) than their NPS, causing them to increase their vengefulness. The population eventually converges to comprise only agents with high vengefulness and low boldness. While noise is still introduced via the exploration rate causing random strategy adoption, the learning capability enables agents with such random strategies to adapt quickly to the trend of the population.

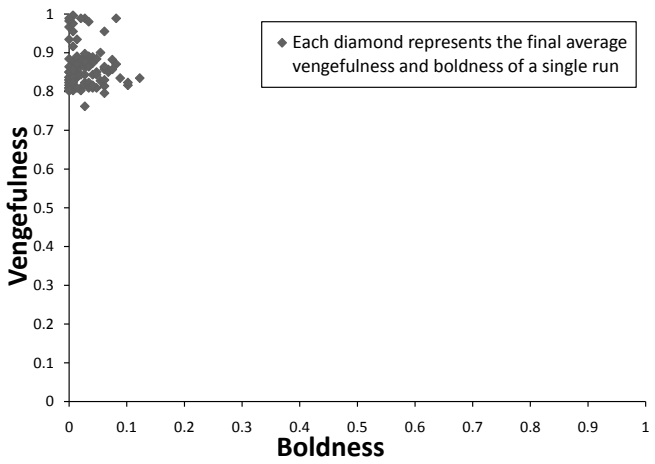

Fig. 5. Strategy improvement (with $\gamma = 0.01$)

As before, we also consider the problem of ensuring that an original defection is observed in order to provide a metapunishment. Introducing this constraint into our new algorithm, we ran experiments over different periods, with results indicating that norm establishment is robust in all runs. An example run for 1,000,000 timesteps is shown in Figure 6. This is because agents that use this new learning algorithm only change their strategy incrementally without wholesale change at any single point. The effect of a mutant with low vengefulness is not significant since, while the mutant might survive for a short period and cause some agents to change their vengefulness, any such change will be slight. It thus does not prevent such agents from detecting the mutant subsequently, in turn causing the mutant to increase its vengefulness.

5 Related Work

In multi-agent systems, research on norm propagation can be divided into two distinct approaches: top-down and bottom-up. In the *top-down* approach, a norm is introduced through a certain *authority*, which is then responsible for the monitoring and enforcement of this norm. In the *bottom-up* approach, agents discover and learn about the norm as a direct result of their interactions and, in most cases, there is no central authority that can enforce such norms. The former approach has been studied and analysed by many (for example, [22, 1, 3]), and this is not the focus of the work in this paper. In contrast, the bottom-up approach, also known as *norm emergence*, has not received the same sort of attention and this is just what this paper addresses.

Nevertheless, the basic notions underlying norm emergence, as understood in this paper, have themselves also been recognised and considered previously. For

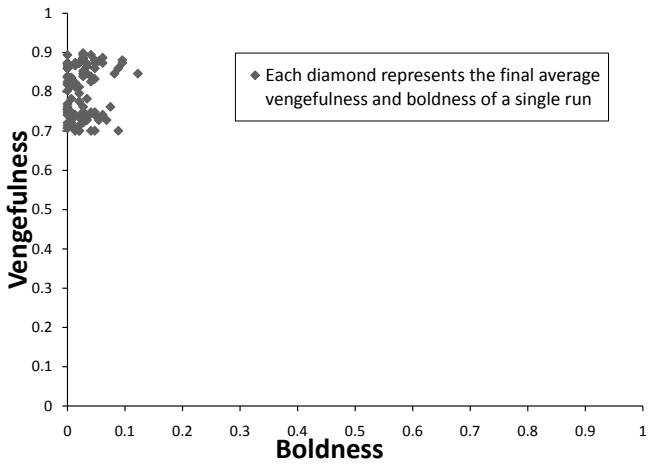

Fig. 6. Strategy improvement with defection observation (with $\gamma = 0.01$)

example, like the work in this paper, Epstein [7] also used imitation techniques in the context of norm emergence. In his model, agents must decide which side of a road to drive on, where the decision of each agent is determined by observation of which side of the road already has more agents driving on it, within a particular area. In this respect, agents imitate what the majority of their neighbours are doing.

Similarly, Savarimuthu et al. [15] also use imitation in their work, which considers the *ultimatum game* in the context of providing advice to agents on whether to change their norms in order to enhance performance. In the ultimatum game, two agents must decide how to share a certain amount of money between them, starting with one agent offering a certain division of the money to the other. If the second agent agrees, then the money is divided between the agents according to the proposal, otherwise both agents gain nothing. Here, each agent has a personal norm that defines its proposal strategy. In addition, agents are able to request advice regarding their proposal strategy from only one agent, the *leader*, which is believed to have the best performance in the requesting agent's neighbourhood. Moreover, agents are capable of accepting of refusing the advice according to their autonomy level.

In relation to the learning aspects of our work, different forms of learning have also been used by other researchers. For example, Walker et al. [20] used a simple strategic update function in their model, based on Conte et al.'s [5] work. In their model, agents wander around searching for food in order to gain energy. However, since this movement causes them to lose energy, they need to find as much food as they can, and incurring the least movement in doing so. For this reason, agents follow different strategies, and change from one strategy to another according to a *majority rule*, which instructs an agent to switch to

another strategy if it finds that the other strategy is used by more agents that its current strategy.

A more complex form of learning has been used by Mukherjee et al. [11, 17], who adopt Q-learning and some of its variants (WOLF-PHC and *fictitious play*) to show the effect of learning on norm emergence. They experimented with two different scenarios, first of homogeneous learning agents (where all agents have the same learning algorithm), and second heterogeneous learning agents (where agents can have different learning algorithms). Their results suggest that norm emergence is achieved in both situations, but is slower in heterogeneous environments.

In addition, some researchers (for example, [14, 19, 12]) have also considered the effect of various types of interaction networks on the *achievement* and *speed* of norm emergence, with results indicating that different types of networks give different outcomes. Though this is an interesting and valuable area to consider, it is outside the scope of this particular paper, so we say no more about it here. Nevertheless, our approach, as reported in the previous sections, is consistent with the broad approach taken by these previous effrorts in terms of analysing the different factors that affect norm emergence. Indeed, the aim of our work is to investigate the effects of metanorms on norm emergence, particularly when metanorms are integrated in a model that reflects key characteristics of distributed systems.

6 Conclusion

In systems of self-interested autonomous agents we often need to establish cooperative norms to ensure the desired functionality. Axelrod's work on norm emergence [2] gives valuable insight into the mechanisms and conditions in which such norms may be established. However, there are two major limitations. First, as Mahmoud et al. [10] have shown previously, and explained in detail, norms collapse even in the metanorms game for extended runs. Second, the model suffers from limitations relating to assumptions of omniscience. In response to this latter point of concern, this paper has (i) investigated those aspects of Axelrod's investigation that are unreasonable in real-world domains, and (ii) proposed *BV learning* as an alternative mechanism for norm establishment that avoids these limitations.

More specifically, we replaced the evolutionary approach with a learning interpretation in which, rather than remove and replicate agents, we allow them to learn from others. Two techniques were considered: copying from a single agent and copying from a group. The former suffers the same problems of long term norm collapse associated with Axelrod's approach [10] but, by avoiding strategies that only perform well in restricted settings, the latter addresses the problems and brings about norm establishment. In addition, we addressed Axelrod's assumption of omniscience, in which agents considering metapunishment are not explicitly required to *see* the original defection. By doing so, however, the

metapunishment activity in the population, for stabilising an established norm, decreases and leads to norm collapse.

Since learning strategies from *others* (either individuals or groups) is unable to establish norms for cooperation (and is, in addition, unrealistic since it assumes that agent strategies are not private), we have developed an alternative, *BV learning*, in which agents learn from their *own* experiences. Through this approach we have shown that not only is it possible to avoid the unrealistic assumption of knowledge of others' strategies, but also that by enabling individuals to incrementally change their strategies we can avoid norm collapse, even with observation constraints on metapunishment.

In term of future work, our aim is to focus on applying the model to interaction networks in order to analyse how different network structures can impact on the achievement of norm emergence. In particular, our current model is limited in that the algorithm relies on agents comparing their own score to the average score of all other agents to determine if learning is warranted. This constrains our move towards turning Axelrod's model into something more suitable for real-world distributed systems and, in consequence, we aim to enable agents to estimate their learning needs based on their own, individual, experience by monitoring their past performance. Moreover, we also plan to investigate the possibility of integrating *dynamic* punishments, rather than the current static ones (that are fixed regardless of what has happened), by which agents can modify the punishments they impose on others according to available information about the severity of violation, or according to whether the violating agent is a repeat offender, and if so, how many times.

References

1. H. Aldewereld, F. Dignum, A. García-Camino, P. Noriega, J. A. Rodríguez-Aguilar, and C. Sierra. Operationalisation of norms for usage in electronic institutions. In *AAMAS '06: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 223–225. ACM, 2006.
2. R. Axelrod. An evolutionary approach to norms. *American Political Science Review*, 80(4):1095–1111, 1986.
3. M. Boman. Norms in artificial decision making. *Artificial Intelligence and Law*, 7(1):17–35, 1999.
4. M. Bowling and M. Veloso. Rational and convergent learning in stochastic games. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 1021–1026, 2001.
5. C. Castelfranchi and R. Conte. Simulative understanding of norm functionalities in social groups. In Gilbert N. Conte R., editor, *Artificial Societies: The Computer Simulation of Social Life*, pages 252–267. UCL Press, 1995.
6. A. P. de Pinninck, C. Sierra, and W. M. Schorlemmer. Friends no more: norm enforcement in multiagent systems. In *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 640–642, 2007.
7. J. M. Epstein. Learning to be thoughtless: Social norms and individual computation. *Computational Economics*, 18(1):9–24, 2001.

8. F. Flentge, D. Polani, and T. Uthmann. Modelling the emergence of possession norms using memes. *Journal of Artificial Societies and Social Simulation*, 4(4), 2001.
9. J. M. Galan and L. R. Izquierdo. Appearances can be deceiving: Lessons learned re-implementing Axelrod's evolutionary approach to norms. *Journal of Artificial Societies and Social Simulation*, 8(3), 2005.
10. S. Mahmoud, N. Griffiths, J. Keppens, and M. Luck. An analysis of norm emergence in axelrods model. In *NorMAS'10: Proceedings of the Fifth International Workshop on Normative Multi-Agent Systems*. AISB, 2010.
11. P. Mukherjee, S. Sen, and S. Airiau. Emergence of norms with biased interactions in heterogeneous agent societies. In *Web Intelligence and Intelligent Agent Technology Workshops, 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, pages 512–515, 2007.
12. M. Nakamaru and S. A. Levin. Spread of two linked social norms on complex interaction networks. *Journal of Theoretical Biology*, 230(1):57 – 64, 2004.
13. R. Riolo, M. Cohen, and R. Axelrod. Evolution of cooperation without reciprocity. *Nature*, 414:441–443, 2001.
14. B. T. R. Savarimuthu, S. Cranefield, M. Purvis, and M. Purvis. Norm emergence in agent societies formed by dynamically changing networks. In *IAT '07: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 464–470, 2007.
15. B. T. R. Savarimuthu, Stephen Cranefield, Maryam Purvis, and Martin Purvis. Role model based mechanism for norm emergence in artificial agent societies. In *Coordination, Organizations, Institutions, and Norms in Agent Systems III, COIN 2007 International Workshops*, volume 4870 of *Lecture Notes in Computer Science*, pages 203–217. Springer, 2008.
16. B. T. R. Savarimuthu, M. Purvis, M. Purvis, and S. Cranefield. Social norm emergence in virtual agent societies. In *Declarative Agent Languages and Technologies VI*, volume 5397 of *Lecture Notes in Computer Science*, pages 18–28, 2009.
17. S. Sen and S. Airiau. Emergence of norms through social learning. In *IJCAI 2007: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence*, pages 1507–1512. Morgan Kaufmann Publishers Inc., 2007.
18. Y. Shoham and M. Tennenholtz. On social laws for artificial agent societies: off-line design. *Artificial Intelligence*, 73(1-2):231–252, February 1995.
19. D. Villatoro, S. Sen, and J. Sabater-Mir. Topology and memory effect on convention emergence. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technologies*, pages 233–240. IEEE, 2009.
20. A. Walker and M. Wooldridge. Understanding the Emergence of Conventions in Multi-Agent Systems. In V. Lesser, editor, *Proceedings of the First International Conference on Multi-Agent Systems*, pages 384–389. MIT Press, 1995.
21. C. J. C. H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3–4):279–292, 1992.
22. F. López y López, M. Luck, and M. 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. ACM, 2002.
23. T. Yamashita, K. Izumi, and K. Kurumatani. An investigation into the use of group dynamics for solving social dilemmas. In P. Davidsson, B. Logan, and K. Takadama, editors, *Multi-Agent and Multi-Agent-Based Simulation*, volume 3415 of *Lecture Notes in Artificial Intelligence*, pages 185–194. Springer, 2005.