Socially Intelligent Genetic Agents for the Emergence of Explicit Norms

Rishabh Agrawal
North Carolina State University
Raleigh, NC, United States
ragrawa3@ncsu.edu

Nirav Ajmeri
University of Bristol
Bristol, United Kingdom
nirav.ajmeri@bristol.ac.uk

Munindar P. Singh
North Carolina State University
Raleigh, NC, United States
mpsingh@ncsu.edu

ABSTRACT

Norms help regulate a society. Norms can naturally emerge from interactions in a society, including conflicts between existing norms and how those conflicts are resolved. We study explicit norms, those that are represented in some structured form by the agents. Previous studies on emergence have not tackled explicit norms.

We propose an approach for the emergence of explicit norms by developing agents who provide and reason about explanations for norm violations in deciding what sanctions to produce and identifying alternative norms. These agents use a genetic algorithm to create and evolve a set of norms and reinforcement learning to learn the value of these norms. We find that providing and evaluating explanations leads to the emergence of norms that provide better cohesion and goal satisfaction for member agents. Our results are stable for societies with differing attitudes of generosity.

CCS CONCEPTS

• **Computing methodologies → Multiagent systems**.

KEYWORDS

Norm Emergence, Normative Systems, Explicit Norms, Simulation

ACM Reference Format:
Rishabh Agrawal, Nirav Ajmeri, and Munindar P. Singh. 2021. Socially Intelligent Genetic Agents for the Emergence of Explicit Norms. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Norms encourage coordination and prosocial interactions in a society [24]. For example, ignoring a phone call in a meeting is a social norm that helps avoid disruption. Importantly, norms may conflict with one another and be resolved in different ways [3, 15, 17, 29]. In conflict, an agent must decide which norms to follow and which to violate. For example, *picking up an urgent call* and *ignoring a call during meeting* are both norms. But if you receive an urgent call during a meeting, one or the other norm must be violated.

Where do norms come from? We consider *norm emergence* to be the decentralized evolution of norms [24], driven by signaling between agents. A norm violation may result in a negative *sanction*, i.e., a scowl on the face of an attendee. If the scowl is outweighed by the benefits of picking up an urgent call, the newly emerged norm would be *ignore a call during meeting, unless it is an urgent call.* Alternatively, if the sanction is severe, the emerged norm could be *pick up an urgent call, unless you are in a meeting.*

Norms can be *implicit* (encoded in behaviors that are common in a society [24]) or *explicit* (explicitly maintained and reasoned about like laws [12, 34] and other regulations [13, 14]). The norm emergence literature focuses on implicit norms. Where explicit norms are studied is in *propagation*, instead of emergence, or in *synthesis*, where norms are centrally designed and evolved [22, 23].

In contrast to the above, we focus on the emergence of *explicit* norms. Morris-Martin et al. [24] lend support to our motivation for decentralized online synthesis of explicit norms. Ajmeri et al. [1] show that sharing contextual information in case of norm violations facilitates norm emergence for implicit norms and results in increased goal satisfaction and cohesion (i.e., perception of norm compliance) by helping agents understand the context in which the norm was violated and learn contextual boundaries.

We adopt the idea of agents providing *explanations* to justify a norm violation. Explanations can help achieve mutual understanding of norms, e.g., for photo sharing [26]; promote user acceptance and to engender trust [4]; and change people's attitudes [36].

Research Objective. Our research objective is to understand how creating and sharing *explanations* for norm violations facilitates the emergence of norms. Here, we develop agents who can produce and reason with explanations. Explicit norms are conducive to explanations of when they are satisfied or violated. However, to realize our objective requires a method for emergence with respect to explicit norms. Accordingly, we propose a method for developing agents that uses rule learning [19] genetic agents. Based on the foregoing, we identify these research questions:

RQ_G (Goal) *Do societies composed of agents who provide and evaluate explanations for norm violations achieve higher goal satisfaction for their members than other societies?*

RQ_C (Cohesion) *Does providing and evaluating explanations lead to the emergence of norms that increase social cohesion?*

Contributions. We make two main contributions. First, we develop a Socially Intelligent Genetic Agent (SIGA), especially an *XSIGA*, which explains its actions and incorporates others' explanations, to learn explicit norms. Second, as the rest of this paper shows, using SIGAs, we answer both RQs positively: (1) Sharing explanations to explain actions lead to improved goal satisfaction; and (2) sharing explanations to explain actions lead to emergence of norms with increased cohesion.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Organization. The rest of the paper is organized as follows. Section 2 presents our method for explicit norm emergence, including how it generates and evaluates explanations. Section 3 describes the simulation environment used to run the experiments. Section 4 presents the experiments run and the discusses the results obtained including threats to validity and mitigation. Section 5 concludes with directions for future work and discussion on related works.

2 METHOD: REALIZING A SIGA

A SIGA's actions are governed by explicit norms. A SIGA includes components for: (1) learning norms based on rewards (sanctions and payoffs) received for each action and (2) discovery to produce new norms and delete old norms of low value.

When generating an explanation for an action, an XSIGA directly presents the supporting norms as justification. When evaluating an explanation, an XSIGA compares the presented norms with its own norms and any applicable norms that have been violated. It assesses whether the supporting norms are more valuable than the violated norm. It accordingly returns a sanction or reward to the other agent.

2.1 Setting and Running Example

Our conception is of a sociotechnical system comprising social entities or *stakeholders* and technical entities, including agents and resources. Each agent has one *primary* stakeholder, whom the agent represents, and potentially many *secondary* stakeholders, who are affected by the agent's actions [1].

We adopt a phone ringer application [1] to explain our method. An agent is responsible for ringing the phone of its primary stakeholder when a call is received or to keep it silent. The secondary stakeholders are the caller and the people in the vicinity of the primary stakeholders, whose privacy may be disturbed due to the ringing of the phone. Actions available to the agent are *ring* or *ignore*. The agent decides to ring the phone or keep it silent based on the norms it follows. The secondary stakeholders may sanction the primary stakeholder if they don't agree with its action.

Suppose Alice's agent follows three norms: *always ring an urgent call*, *always ring a call from a family member*, and *ignore calls during a meeting*. Suppose Bob's agent follows the norms: *always ring an urgent call*, *ignore calls during a meeting*. Suppose Alice gets an urgent call from a family member during a meeting. Her agent compares the relative value of each action in terms of the norms that support it and chooses an action. If the combined value of *always ring an urgent call* and *always ring a call from a family member* is more than the norm *ignore calls during a meeting*, it rings the phone. If Alice's agent does not provide an explanation for its action, Bob's agent would think that the only applicable norm was *ignore calls during a meeting*, which has been violated, and it would sanction Alice. If Alice presents the supporting norms, *always ring an urgent call* and *always ring a call from a family member* as an explanation, these norms are evaluated by Bob's agent. Suppose Charlie's agent follows not *always ring a call from a family member*, but *always ring an urgent call.* If Charlie's agent assigns a higher value to *always ring an urgent call* over *ignore calls during a meeting*, it would not issue a sanction. Otherwise, it would reject the explanation and sanction Alice. This sanction would be used by Alice to adjust the value of its three norms.

2.2 Norm Representation

Norms in our conception are commitments [32]. A commitment norm is a tuple of Commitment(subject, object, *antecedent*, *consequent*). It is directed and conditional [32]. It comprises an antecedent that determines when it is applicable, a consequent that determines when it completes (is satisfied or violated), a subject agent on whom it is focused, and an object agent to whom it is directed. Using this scheme, the norm *always ring a call from a family member* is formalized as: Commitment(Callee, Caller, callerRel = *family*, action=*ring*), where the callee is the subject, caller is the object, callerRel = *family* is the antecedent and the action *ring* is the consequent.

2.3 Norm Learning and Discovery

A rule-based approach to implement norms provides a happy middle ground between flexibility and ease of implementation [32]. Each agent stores norms as rules that it learns, evaluates, and evolves.

Here, the antecedents are conjunctions of key-value pairs, e.g., callerRel = family, urgent = true. The consequents are actions to be taken to satisfy the norm. Thus, a norm maps to a rule of the form:

IF *antecedent* THEN *consequent*

We adapt *eXtended Learning Classifiers* (XCS), a rule learning algorithm based on reinforcement learning [5, 33], which evolves rules by learning from rewards obtained from the environment in response to actions. This algorithm enables agents to discover norms and to learn their values. Specifically, we

(1) map the sanctions and payoffs received by an agent and aggregate them to rewards;
(2) map norms to IF-THEN rules to be manipulated by XCS; and
(3) define crossover and mutation operations for norms to enable norm discovery.

Figure 1 shows a flowchart of the method as it chooses an action based on rules and handles the reward to update rules.

XCS operates in two modes, *exploration* (take random actions to find new norms) and *exploitation* (apply learned rules and their weights to choose an action). We adopt the ϵ-greedy technique, choosing exploration with probability ϵ.

Each rule has associated parameters of *fitness* (worth of the rule in making accurate predictions), *reward prediction*, *prediction error* (of the reward), and *numerosity* (how many "copies" of the rule exist, indicating robustness against accidental deletion [33]).

2.3.1 *Create Match Set.*

The *match set* is the set of rules that are activated in a given context. This means that the antecedents of the corresponding norms are true in the given context.

2.3.2 *Cover Context.*

Covering ensures that rules with sufficiently many actions are available for a given context. This is done to maintain diversity in rules, so that we don't overfit to the initial conditions. Rules generated through covering are added to the agent's ruleset. The new rules are generated by randomly selecting some subset of the context to be the antecedent of the rule. This ensures that the antecedent would be true in the given context. The consequent is randomly selected from available actions. Covering

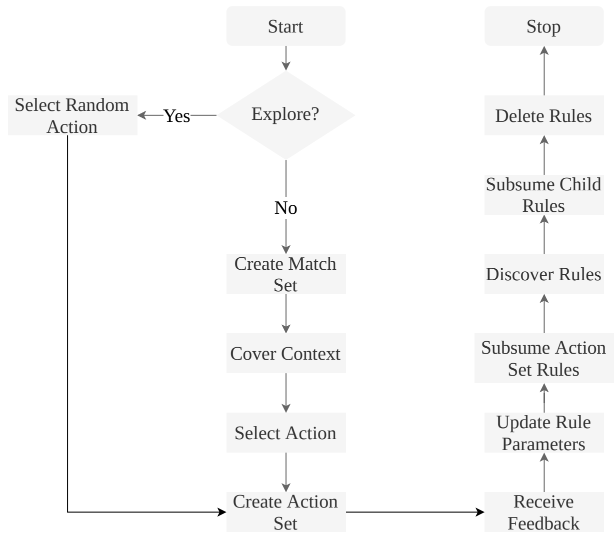

Figure 1: Method in brief: A SIGA's norm use and discovery.

is essential at the beginning, since each agent has an empty ruleset, meaning its match set is empty.

2.3.3 Select Action. The expected value of each action aggregates the fitness-weighted reward predictions of rules supporting that action. In terms of norms, this is the expected reward of following each norm. Pick the available action with highest expected value.

2.3.4 Create Action Set. The matching rules that support the chosen action form the *action set*. We create this set by identifying which rules had suggested the action chosen in the previous step.

2.3.5 Update Rule Parameters. Upon receiving a reward from the environment, the agent updates its parameters to update the value estimation of rules. Because this is a single-step problem, we use the Widrow-Hoff update [35], instead of Q-learning [6]. Only rules in the action set are updated as we don't know the impact of following any other action. Reward prediction is updated using Equation 1, where p is the reward prediction, r is the reward received, and β is a hyperparameter controlling the rate of learning [33].

$$p \leftarrow p + \beta(r - p) \tag{1}$$

Fitness update is based on accuracy κ. We update the prediction error, ε, using Equation 2 [33]. In Equation 3, ε_0 is the error threshold below which we assume a rule to be accurate. Here, υ is the parameter that changes the relationship between error and accuracy to increase the difference in fitness levels between two rules that are close in prediction error [33]. This is a way of preferring a less error-prone rule during discovery. And, α is the scaling factor used to raise the least error-prone non-accurate classifier to be close to an accurate classifier [33].

$$\varepsilon \leftarrow \varepsilon + \beta(|r - p| - \varepsilon) \tag{2}$$

$$\kappa = \begin{cases} 1 & \text{if } \varepsilon < \varepsilon_0 \\ \alpha(\frac{\varepsilon}{\varepsilon_0})^{-\upsilon} & \text{otherwise} \end{cases} \tag{3}$$

We normalize accuracy to κ' using Equation 4 [33], and update fitness F using Widrow-Hoff update [35]:

$$\kappa' = \frac{\kappa}{\sum_{cl \in |A|} \kappa_{cl}} \tag{4}$$

$$F \leftarrow F + \beta(\kappa' - F) \tag{5}$$

2.3.6 Subsume Action Set Rules. Subsumption means replacing a more error-prone (**reward prediction**), less general rule with an existing less error-prone, more general rule. This process provides a generalization pressure to the algorithm. If the first rule below is more error-prone than the second, it may be replaced by the second, thus increasing the latter's numerosity.

IF urgent=true $\wedge$ callerRel = friend THEN ring
IF urgent=true THEN ring

2.3.7 Discover Rules. We apply a genetic algorithm (GA) to generate new rules from current rules of high fitness. GA selects parents using *tournament selection* [33]. Some rules are randomly selected to compete in a *tournament*. We use 30% of the action set size as the tournament size. The fittest among them is chosen as a parent, repeating the process to identify two parents. The parents are selected with replacement and can possibly be the same rule. Two children are then generated using *crossover* and *mutation* operations, which we define for norms. The new rules are added to the population of rules maintained by the agent. To allow the rules to stabilize, we breed them only when the average experience (number of times a rule has been selected) of the rules in the current action set is above a certain threshold. We use single-point crossover, in which the values are randomly swapped for each contextual property present in either parent's antecedent.

In XCS, mutation is used to create a more general or more specific rule by randomly flipping bits in the parent encoding. For norms, we randomly remove or add using the $\wedge$ operator key-value pairs in the antecedent of the norm being mutated. For example, if the antecedent of a norm is {callerRel = friend $\wedge$ urgent = true}, we may mutate it to {urgent = true} by removing a pair or to {callerRel = friend $\wedge$ urgent = true $\wedge$ calleeLoc = home}, if the location in the current context is home.

2.3.8 Subsume Child Rules. A child rule is subsumed into its parent if the parent is a more generic version of the child and its error is less than a threshold. This is called *genetic subsumption*. It happens just after the creation of the child rules.

2.3.9 Delete Rules. A hyperparameter defines the maximum number of rules a SIGA can keep. To stay within that threshold, we apply Kovacs' deletion scheme 3 [18], which deletes unfit rules with a higher probability than others.

2.4 Explanations

An explanation is comprised of norms that support the action taken, i.e., the action set identified above. An agent evaluating the explanation finds norms it follows in the explanation plus the norms that have been violated. It adds these rules to the match set and then follows the same procedure as action selection by performing fitness-weighted aggregation of reward prediction. If the action selected matches the observed action, the sanction is avoided.

3 SIMULATION SCENARIO: RINGER

This scenario is based on our running example and is implemented using MASON [20]. The simulation consists of a population of agents. There are five shared locations where agents can interact: homes (H), parties (P), meetings (M), a library (L) and an emergence room (ER). Some of these locations (H, P, and M) have an associated relationship circle. Each home has a family circle, each party has a friend circle, and each meeting has a colleague circle. People of the same circle share that relationship. Agents stay at one location for a random number of steps chosen from a Gaussian distribution with mean of 60 steps and standard deviation of 30, with the number of steps restricted to the range $[30, 90]$. Then they move to another location. They are more likely to enter a location that is associated with their own circles (75% probability) than a location with which they don't have an association (25% probability). For example, they are more likely to enter their own homes than a stranger's home.

At each timestep, each agent makes a call to another agent with probability chosen randomly from a Gaussian distribution with mean of 5% and standard deviation of 1%. There is a 25% probability of calling a family member, 25% of calling a colleague, 25% of calling a friend, and 25% of calling a stranger. On receiving a call, the callee evaluates its norms and context to determine if it should ring the phone or keep it silent.

All agents have goals based on their role (Table 1). The degree to which the agents are affected by the promotion or demotion of these goals determine the payoff that they receive. The payoff may differ based on location and relationship.

Table 1: Agent goals.

Role	Goal	Action
Caller	reach callee either urgently or casually	Answer
Callee	to be reachable	Answer
	to not be disturbed	Ignore
	to not disturb others	Ignore
Neighbor	to not be disturbed	Ignore

Tables 2, 3, and 4 present the callee, caller, and neighbor payoffs, respectively. Neighbor payoffs and sanctions may be impacted by the explanation provided, depending on whether the explanation was accepted or not. Table 5 summarizes the expected payoff of each situation in this case.

Table 2: Callee payoff based on urgency and relationship.

Caller Relationship	Callee Action	Casual	Urgent
Family, Friend, Colleague	Answer	0.50	1.00
	Ignore	0.00	−0.50
Stranger	Answer	−1.50	0.50
	Ignore	1.50	−0.25

Table 3: Caller payoff based on urgency of the call.

Callee Action	Casual	Urgent
Answer	0.50	1.00
Ignore	−0.50	−1.00

Table 4: Neighbor payoff based on location of the call.

Callee's Action	ER	H	L	M	P
Answer	1.00	0.67	−1.00	−1.00	−0.33
Ignore	−1.00	−0.33	1.00	1.00	0.67

Table 5: Neighbor payoff based on evaluated explanations.

Callee Action	Neighbor Expects	ER	H	L	M	P
Answer	Answer	1.00	0.67	1.00	1.00	0.67
Answer	Ignore	−1.00	−0.33	−1.00	−1.00	−0.33
Ignore	Answer	−1.00	−0.33	−1.00	−1.00	−0.33
Ignore	Ignore	1.00	0.67	1.00	1.00	0.67

3.1 Contextual Properties

The properties of the context that determine the agent action are callee's location (home, party, meeting, library, and ER), relationship with caller (family, friend, colleague, stranger), and call urgency.

3.2 Types of Societies

Agents optimize their norms as per a utility calculated by weighted sum of payoffs received by each stakeholder. We define types of societies based on the generosity of their members in terms of the weight they place on the welfare (payoff) of their peers.

Selfish Members give weight only to their own payoff.
Pragmatic Members give equal weight to the payoff of everyone.
Considerate Members give weight only to the payoffs of others.
Mixed Members are a mix with 25% selfish, 25% considerate, and 50% pragmatic agents.

4 EXPERIMENTS AND RESULTS

To address our research questions, we run simulations of pragmatic, selfish, considerate, and mixed agent societies using three kinds of agents: We evaluate three types of agents in this work.

Fixed agent These agents follow a fixed set of norms (Table 6). When norms conflict they choose a random action.
NSIGAs These agents maintain and evolve a set of explicit norms following our proposed mechanism. When they violate a norm, they accept the sanction and use that feedback to guide learning. They do not offer explanations.
XSIGAs These agents maintain and evolve a set of explicit norms following our mechanism, and offer explanations of their actions that are evaluated by secondary stakeholders to determine if the primary stakeholder should be sanctioned.

Table 6: Norms followed by a fixed agent.

Location	Response
Emergency Room (ER)	Answer
Home (H)	Answer
Library (L)	Ignore
Meeting (M)	Ignore
Party (P)	Answer

Circle	Casual	Urgent
Colleague	Answer	Answer
Family	Answer	Answer
Friend	Answer	Answer
Stranger	Ignore	Answer

4.1 Evaluation Metrics and Hypotheses

Our metrics include *Social Experience*, *Cohesion* and *Adoption*.

Social Experience measures the degree of goal satisfaction delivered by an agent [1]. We compute social experience as the weighted aggregate of payoffs received by all the stakeholders as the result of an agent's actions [1]. The weights depend on the nature of the agent (selfish, considerate, pragmatic) as defined in Section 3.2.

$$\text{Social Experience} = \frac{\text{Weighted sum of payoffs for each action}}{\text{Total number of actions}} \tag{6}$$

Cohesion measures the perception of norm compliance [1], and is defined as a ratio of agent's actions perceived as norm compliant by agents who are affected by each action divided by the total number of interactions.

$$\text{Cohesion} = \frac{\text{\# of agents who perceive action as norm compliant}}{\text{\# of interactions}} \tag{7}$$

Adoption measures the percentage of agents in a society who follow a particular norm. A norm is said to have emerged when a sufficiently high percentage of the population complies with the norm [10]. Recognizing literature [7, 16, 31], we consider 90% adoption as the threshold for norm emergence.

$$\text{Adoption of a norm} = \frac{\text{Number of agents following the norm}}{\text{Total number of agents}} \tag{8}$$

We evaluate the following hypotheses: $\mathbf{H_1}$ and $\mathbf{H_2}$ relate to $\mathbf{RQ_G}$; $\mathbf{H_3}$ and $\mathbf{H_4}$ relate to $\mathbf{RQ_C}$; and $\mathbf{H_5}$ relates to our overall research objective of studying the impact of explanations on norm emergence. Each hypothesis compares the *XSIGA* approach with the other approaches. We omit the corresponding null hypotheses for brevity.

- $\mathbf{H_1}$ XSIGAs perform better than *Fixed* agents in terms of *social experience.*
- $\mathbf{H_2}$ XSIGAs perform better than *NSIGAs* in terms of *social experience.*
- $\mathbf{H_3}$ XSIGAs performs better than *Fixed* agents in terms of *cohesion.*
- $\mathbf{H_4}$ XSIGAs performs better than *NSIGAs* in terms of *cohesion.*
- $\mathbf{H_5}$ XSIGAs performs better than *NSIGAs* in terms of *adoption* of emerged norms.

We report results for each simulation run eight times for 10,000 timesteps. To evaluate these hypotheses, we conduct a paired t-test and compute Cohen's d to measure the effect size.

4.2 Experiment: Pragmatic Agent Society

Table 7 summarizes the social experience, cohesion and adoption metrics yielded by the three agent types in a pragmatic society. Figure 2 compares the social experience plots for Fixed, NSIGAs, and XSIGAs agents in a pragmatic society. We find the social experience, cohesion and adoption yielded by XSIGAs to be better ($p < 0.01$; d$>$ 0.8, indicating large effect) than the two baselines.

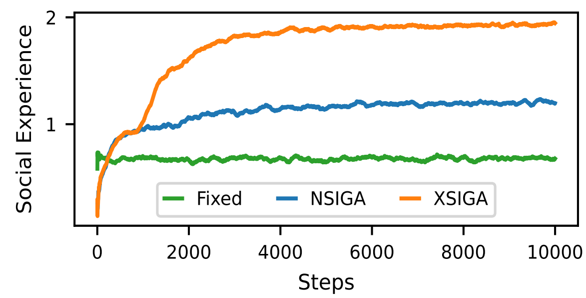

Figure 2: Social Experience in Pragmatic Society.

Table 7: Social experience and cohesion in pragmatic society.

Agent Type	Experience		Cohesion		Adoption	
	Mean	SD	Mean	SD	Mean	SD
Fixed	0.68	0.01	27.06%	0.10	–	–
NSIGA	1.21	0.01	55.65%	0.34	96.49%	0.10
XSIGA	1.94	0.01	88.81%	0.25	98.45%	0.22

Figure 3 shows adoption of norms for different approaches in pragmatic society. We focus on the SIGA approaches because Fixed agents do not adopt new norms. Each dot is a norm. Providing explanations help in identifying useful norms and encourage their adoption, while discouraging the adoption of useless norms. As a result, it leads to a more extreme distribution of adoption—see Figure 3.

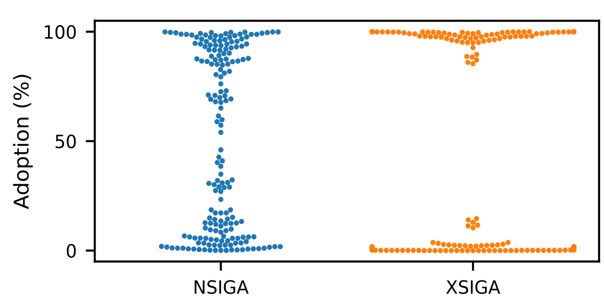

Figure 3: Adoption of norms: Pragmatic Society.

Table 8 shows the norms that emerge using SIGAs. The norm emerged with XSIGA is to always ring. By ringing a call, the agent provides a better experience to the caller. The neighbor's experience depends on the acceptance or rejection of the explanation. The callee experience can be negative in some cases like ringing a casual call by a stranger, but the pragmatic agent values payoffs of all agents and thus chooses to ring a call. NSIGAs get harsher penalties from their neighbors for violating norms as they cannot provide explanations. So they become more cautious about violations. As a result, the emerged norm is to ring an urgent call or call by a known person or call in an ER because these provide sufficient positive payoffs to overcome possible negative neighbor reactions. Whereas Figure 3 shows several norms with adoption more than 90%, Table 8 lists only one emerged norm for XSIGAs, *ring all calls*. This is because this norm is more general than all other emerged norms, such as *ring urgent calls*.

Table 8: Norms in Pragmatic Society, XSIGAs and NSIGAs.

Antecedent	Consequent	Adoption
XSIGAs		
true	ring	91.2%
NSIGAs		
urgent = true	ring	97.4%
callerRel = colleague	ring	93.1%
callerRel = family	ring	92.7%
calleeLoc = home	ring	92.4%
callerRel = friend	ring	92.2%
calleeLoc = ER	ring	92.2%

4.3 Experiment: Selfish Agent Society

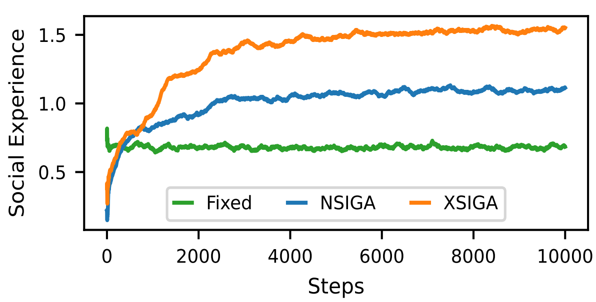

Figure 4: Social Experience in Selfish Society.

Table 9 summarizes the social experience, cohesion and adoption metrics yielded by the three agent types in a selfish society. Figure 4 compares the social experience plots for Fixed, NSIGAs, and XSIGAs agents in a selfish society. We find the social experience and cohesion yielded by XSIGAs to be better ($p < 0.01$; d$>$ 0.8, indicating large effect) than the two baselines, and thus reject the null hypotheses corresponding to H_1–H_4. For adoption we find that

Table 9: Social experience and cohesion in selfish society.

Agent Type	Experience		Cohesion		Adoption	
	Mean	SD	Mean	SD	Mean	SD
Fixed	0.68	0.02	27.09%	0.13	–	–
NSIGA	1.10	0.01	49.20%	0.13	96.07%	0.22
XSIGA	1.55	0.01	68.46%	0.23	95.94%	0.31

the difference in mean values is not statistically significant, thus we fail to reject the null hypothesis corresponding to H_5.

Figure 5 shows adoption of norms for different approaches in a selfish society. Providing explanations has a lower impact on emergence in a selfish society than others. This is because an explanation is meant to explain the agent's actions to the other agents to get a positive response. But selfish agents don't give weight to anyone's payoff other than their own and thus do not give value to responses from other agents. As a result, they only learn norms that benefit themselves without caring about evaluation of explanation. They still provide and evaluate explanations, but they don't let it influence the norm they learn. This is the reason that the mean adoption among emerged norms is similar for XSIGAs and NSIGAs. Providing an explanation is better in terms of social experience and cohesion because neighbors are also selfish agents who can accept an explanation based on a norm learnt due to selfish tendencies. This acceptance of explanation lead to better social experience and cohesion, even if it doesn't impact the exact norms that emerge.

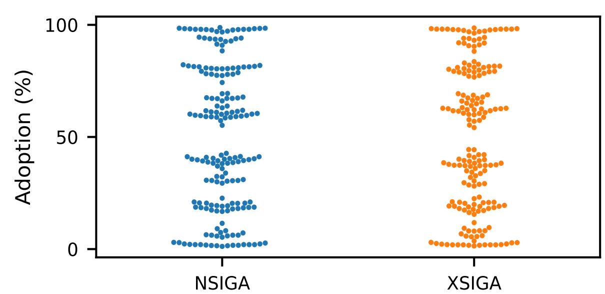

Figure 5: Adoption of norms: Selfish Society.

Table 10 shows norms that have emerged using SIGAs. The norms emerged for XSIGAs and NSIGAs are similar, in line with our expectation that norms emerging in a selfish society won't be impacted by providing explanations. The emerged norm is to pick up an urgent call and to always ignore a casual call from a stranger. This is because a selfish agent does not value the payoff of neighbors or callers. It ignores a casual call from a stranger even in a nonrestrictive location like ER where the neighbors expect it to ring a call.

4.4 Experiment: Considerate Agent Society

Table 11 summarizes the social experience, cohesion and adoption metrics yielded by the three agent types in a considerate society.

Table 10: Norms in Selfish Society, XSIGAs and NSIGAs.

Antecedent	Consequent	Adoption
XSIGAs		
urgent = true	ring	91.4%
urgent = false ∧ callerRel = stranger	ignore	90.2%
NSIGAs		
urgent = true	ring	91.4%
urgent = false ∧ callerRel = stranger	ignore	90.2%

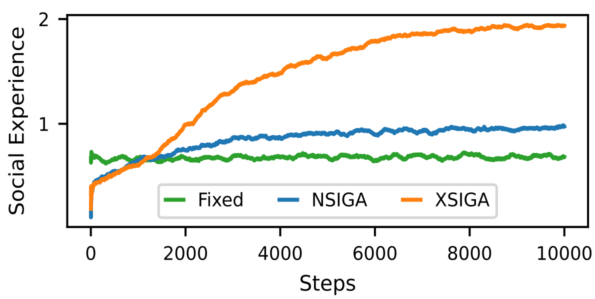

Figure 6: Social Experience in Considerate Society.

Figure 6 compares the social experience plots for Fixed, NSIGAs, and XSIGAs agents in a considerate society. We find the social experience, cohesion and adoption yielded by XSIGAs to be better ($p < 0.01$; d> 0.8, indicating large effect) than the two baselines, and thus reject the null hypotheses corresponding to H_1–H_5.

Table 11: Social experience, cohesion: considerate society.

Agent Type	Experience		Cohesion		Adoption	
	Mean	SD	Mean	SD	Mean	SD
Fixed	0.69	0.01	27.11%	0.14	–	–
NSIGA	0.97	0.02	69.72%	0.24	93.70%	0.23
XSIGA	1.93	0.01	77.48%	0.63	97.18%	0.30

Figure 7 shows adoption of norms for different approaches in a considerate society. As in a pragmatic society, providing explanations had a polarizing effect on adoption. This helps in emergence of norms by increasing adoption of the emerged norms.

Table 12 shows explicit norms that have emerged using SIGAs. For XSIGAs, the effective norm is to ring in all cases due to similar reasons as the pragmatic agent. We have shown that even within the norm, a more specialized version like ringing an urgent call has higher adoption than ringing a casual call. For NSIGAs, the agent needs to be more careful with the neighbor payoff, as for pragmatic agents. As a result, the most adopted norms are based on location and urgency.

4.5 Experiment: Mixed Agent Society

Table 13 summarizes the social experience, cohesion and adoption metrics yielded by the three agent types in a mixed society. Figure 8

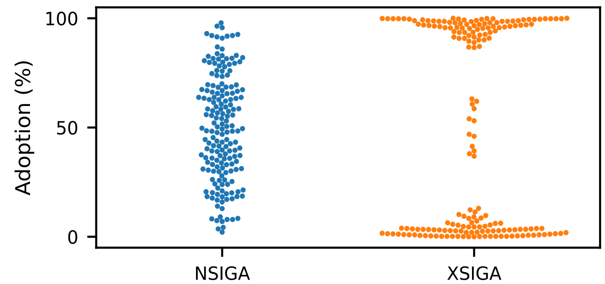

Figure 7: Adoption of norms: Considerate Society.

Table 12: Norms in Considerate Society, XSIGAs and NSIGAs.

Antecedent	Consequent	Adoption
XSIGAs		
urgent = true	ring	97.4%
urgent = false	ring	92.2%
NSIGAs		
urgent = true ∧ calleeLoc = home	ring	92.0%
urgent = true ∧ calleeLoc = ER	ring	92.3%
urgent = false ∧ callerRel = stranger	ignore	90.2%

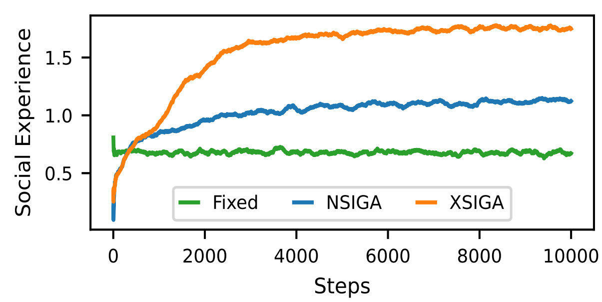

Figure 8: Social Experience in Mixed Society.

compares the social experience plots for Fixed, NSIGAs, and XSIGAs agents in a mixed society. We find the social experience, cohesion, and adoption yielded by XSIGAs to be better ($p < 0.01$; d> 0.8, indicating large effect) than the two baselines, and thus reject the null hypotheses corresponding to H_1–H_5.

Table 13: Social experience and cohesion in mixed society.

Agent Type	Experience		Cohesion		Adoption	
	Mean	SD	Mean	SD	Mean	SD
Fixed	0.67	0.01	26.99%	0.11	–	–
NSIGA	1.12	0.01	57.55%	0.22	95.09%	0.18
XSIGA	1.76	0.01	77.69%	0.29	95.79%	0.25

Figure 9 shows adoption of norms for different approaches in mixed society. Same polarizing effect as pragmatic and considerate societies, can be observed in this case as well.

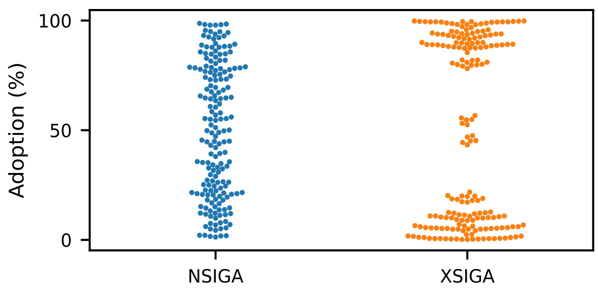

Figure 9: Adoption of norms: Mixed Society.

Tables 14 shows explicit norms that have emerged using SIGAs. For a society of XSIGAs, the emerged norm is to ring urgent calls and calls from known people. This is more specific than the pragmatic society norm of always ringing because there are also selfish agents who prefer to ignore casual calls from strangers. For the society of NSIGAs, we see that callee's location is important in adopted norms because of neighbor payoffs, as in the other societies.

Table 14: Norms in Mixed Society, XSIGAs and NSIGAs.

Antecedent	Consequent	Adoption
XSIGAs		
urgent = true	ring	96.8%
callerRel = colleague	ring	92.0%
callerRel = friend	ring	91.8%
callerRel = family	ring	91.4%
NSIGAs		
urgent = true ∧ calleeLoc = home	ring	95.5%
urgent = true ∧ calleeLoc = ER	ring	95.0%
urgent = true ∧ callerRel = friend	ring	94.9%
urgent = true ∧ callerRel = family	ring	94.5%
urgent = true ∧ callerRel = colleague	ring	93.8%

4.6 Threats to Validity

We identified three threats and mitigated two of them. First, our evaluation is based on manually created payoffs. Getting correct valuation of user experience from actual users is difficult. We mitigate this threat by adapting data from the literature. Appendix D (in supplement) demonstrates the stability of our results on other payoffs. Second, user attitudes differ in the relative value placed on the welfare of other people. We represent societies with different types of attitudes to show the robustness of our approach. Third, simulation as an evaluation methodology always omits some details of a real society. Our model gains some realism by incorporating locations, relationship circles, and different attitudes of generosity, but the threat is not completely mitigated. Note that, however, our focus is not to model a real society but to show emergence of explicit norms.

5 DISCUSSION

We propose an approach for explicit norm emergence using socially intelligent genetic agents (SIGA), which can provide and evaluate explanations. We find that societies composed of XSIGAs have better social experience and cohesion than the baselines and that providing and evaluating explanations leads to better adoption of emerged norms, except for a selfish society.

This work can be extended in many ways. We can explore how to generate explanations with more generic terminology to preserve privacy. The tradeoff of privacy concerns and potential advantages of providing a specific explanation may be studied. Using hierarchical ontology of terms could also be used to learn more complex boundaries of the norms.

SIGAs can be enhanced to incorporate value preferences, ethics and fairness in explicit norm emergence [2, 27, 28, 30].

Mosca and Such [26] propose an explainable agent for multiuser privacy scenarios and generate explanations for the decisions of the agent to be evaluated by a human. In contrast, our work focuses on generating normative explanations that can be evaluated by another agent for adoption. Hind et al. [11] generate explanations for the decisions of an AI agent. They use a machine learning model to predict both actions and explanations. In our method, the explanations are the norms used for arriving at the decision and thus justify the action they explain.

Morales et al. [23] perform synthesis of normative system using an evolutionary process like ours. But theirs is an offline and centralized mechanism to evolve norms, whereas in our work norms evolve in an online and decentralized manner. Also, our work is driven by sanctions, whereas they do not consider sanctions and instead align norms to individual goals of an agent.

Mashayekhi [21] proposes a decentralized norm emergence framework called *Cha* for normative systems which is driven by conflict detection. Norms are created which can avoid conflicts. On the other hand, SIGAs are driven by sanctions and sharing explanations.

Morris-Martin et al. [25] address the problem of creating a norm emergence framework. They suggest the use of synthesizer agents, a central Oracle, and a central normative system. The agents can suggest changes to this system. In contrast, we maintain norms individually and explicitly and evolve them by each agent without a central entity. We use sanctions and explanations to drive the norms to convergence.

Morales et al. [22] study centralized norm emergence where conflict scenarios are recognized and norms adapted to handle the conflicts. This is opposed to our work where agents themselves are responsible for following and enforcing the norms.

Hao et al. [9] proposes two heuristic collective learning frameworks which use reinforcement learning to learn the best response for each state. A SIGA uses genetic exploration to evolve the set of antecedents used in its set norms, instead of learning for all states.

Dell'Anna et al. [8] uses Bayesian Networks to revise sanctions associated with enforced norms. In contrast, SIGA revises the norms themselves while the sanctions stay the same.

Ajmeri et al. [1] share not explanations but the context in which the decision was made to violate the norm. This context is evaluated by other agents to decide if they would have done the same action in this context as a basis for sanctioning. In contrast, we share the explicit norms that influenced a decision and others reason about these norms. Ajmeri et al. provide an implicit norm system that learns actions to be taken in different contexts, whereas our work is an explicit norm system that explicitly learns and reasons about norms. They reveal the entire context, whereas our explanation reveals only the relevant parts of the context. For example, if someone follows the norm of picking up all urgent calls and received an urgent call from a family member, our explanation would reveal the norm of picking up urgent calls and keep the nature of relationship with the caller private as it was not a contributing factor.

REFERENCES

[1] Nirav Ajmeri, Hui Guo, Pradeep K. Murukannaiah, and Munindar P. Singh. 2018. Robust Norm Emergence by Revealing and Reasoning about Context: Socially Intelligent Agents for Enhancing Privacy. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI, Stockholm, 28–34. https://doi.org/10.24963/ijcai.2018/4

[2] Nirav Ajmeri, Hui Guo, Pradeep K. Murukannaiah, and Munindar P. Singh. 2020. Elessar: Ethics in Norm-Aware Agents. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. IFAAMAS, Auckland, 16–24. https://doi.org/10.5555/3398761.3398769

[3] Nirav Ajmeri, Jiaming Jiang, Rada Chirkova, Jon Doyle, and Munindar P. Singh. 2016. Coco: Runtime Reasoning about Conflicting Commitments. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*. AAAI Press, New York, 17–23.

[4] Or Biran and Courtenay Cotton. 2017. Explanation and justification in machine learning: A survey. In *IJCAI-17 Workshop on Explainable AI (XAI)*, Vol. 8. IJCAI, Melbourne, Australia, 8–13.

[5] Martin V Butz and Stewart W Wilson. 2000. An algorithmic description of XCS. In *International Workshop on Learning Classifier Systems*. Springer, 253–272.

[6] Watkins Chris. 1989. *Learning from Delayed Rewards*. Ph.D. Dissertation. University of Cambridge.

[7] Jordi Delgado. 2002. Emergence of social conventions in complex networks. *Artificial Intelligence* 141, 1-2 (2002), 171–185.

[8] Davide Dell'Anna, Mehdi Dastani, and Fabiano Dalpiaz. 2020. Runtime Revision of Sanctions in Normative Multi-Agent Systems. *Autonomous Agents and Multi-Agent Systems (JAAMAS)* 34, 2 (2020), 1–54. https://doi.org/10.1007/s10458-020-09465-8

[9] Jianye Hao, Jun Sun, Guangyong Chen, Zan Wang, Chao Yu, and Zhong Ming. 2018. Efficient and Robust Emergence of Norms through Heuristic Collective Learning. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* 12, 4 (2018), 23:1–23:20. https://doi.org/10.1145/3127498

[10] Chris Haynes, Michael Luck, Peter McBurney, Samhar Mahmoud, Tomáš Vítek, and Simon Miles. 2017. Engineering the Emergence of Norms: A Review. *The Knowledge Engineering Review* 32 (2017), e18.

[11] Michael Hind, Dennis Wei, Murray Campbell, Noel C. F. Codella, Amit Dhurandhar, Aleksandra Mojsilović, Karthikeyan Natesan Ramamurthy, and Kush R. Varshney. 2019. TED: Teaching AI to Explain Its Decisions. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society* (Honolulu, HI, USA) *(AIES '19)*. Association for Computing Machinery, New York, NY, USA, 123–129. https://doi.org/10.1145/3306618.3314273

[12] Wesley Newcomb Hohfeld. 1919. *Fundamental Legal Conceptions as Applied in Judicial Reasoning and other Legal Essays*. Yale University Press, New Haven, Connecticut. A 1919 printing of articles from 1913.

[13] Andrew J. I. Jones and Marek J. Sergot. 1993. On the Characterisation of Law and Computer Systems: The Normative Systems Perspective. In *Deontic Logic in Computer Science: Normative System Specification*, John-Jules Ch. Meyer and Roel J. Wieringa (Eds.). John Wiley and Sons, Chichester, United Kingdom, Chapter 12, 275–307.

[14] Özgür Kafalı, Nirav Ajmeri, and Munindar P. Singh. 2020. Desen: Specification of Sociotechnical Systems via Patterns of Regulation and Control. *ACM Transactions on Software Engineering and Methodology (TOSEM)* 29, 1 (Feb. 2020), 7:1–7:50. https://doi.org/10.1145/3365664

[15] Daniel Kasenberg and Matthias Scheutz. 2018. Norm conflict resolution in stochastic domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[16] James E. Kittock. 1993. Emergent conventions and the structure of multi-agent systems. In *Proceedings of the Santa Fe Institute Complex Systems Summer School*, Vol. 6. Citeseer, 1–14.

[17] Martin J Kollingbaum, Wamberto W Vasconcelos, Andres García-Camino, and Tim J Norman. 2007. Managing conflict resolution in norm-regulated environments. In *International Workshop on Engineering Societies in the Agents World*. Springer, 55–71.

[18] Tim Kovacs. 1999. Deletion Schemes for Classifier Systems. In *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation (GECCO)*. Morgan Kaufmann Publishers Inc., Orlando, 329–336.

[19] Han Liu, Alexander Gegov, and Mihaela Cocea. 2015. *Rule Based Systems for Big Data: A Machine Learning Approach*. Vol. 13. Springer.

[20] Sean Luke, Claudio Cioffi-Revilla, Liviu Panait, Keith Sullivan, and Gabriel Balan. 2005. MASON: A Multiagent Simulation Environment. *Simulation: Transactions of the Society for Modeling and Simulation International* 81, 7 (July 2005), 517–527.

[21] Mehdi Mashayekhi. 2016. *Regulation of Open Normative Multiagent Systems*. Ph.D. Dissertation. Department of Computer Science, North Carolina State University, Raleigh. http://www.lib.ncsu.edu/resolver/1840.20/33373

[22] Javier Morales, Maite López-Sánchez, Juan A. Rodríguez-Aguilar, Michael Wooldridge, and Wamberto Weber Vasconcelos. 2014. Minimality and Simplicity in the On-line Automated Synthesis of Normative Systems. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems, AAMAS*. IFAAMAS, Paris, 109–116.

[23] Javier Morales, Michael Wooldridge, Juan A. Rodríguez-Aguilar, and Maite López-Sánchez. 2018. Off-line Synthesis of Evolutionarily Stable Normative Systems. *Autonomous Agents and Multi-Agent Systems (JAAMAS)* 32, 5 (01 Sep 2018), 635–671.

[24] Andreasa Morris-Martin, Marina De Vos, and Julian Padget. 2019. Norm Emergence in Multiagent Systems: A Viewpoint Paper. *Autonomous Agents and Multi-Agent Systems (JAAMAS)* 33, 6 (2019), 706–749.

[25] Andreasa Morris-Martin, Marina De Vos, and Julian A. Padget. 2020. A Norm Emergence Framework for Normative MAS - Position Paper. *CoRR* abs/2004.02575 (2020). arXiv:2004.02575 https://arxiv.org/abs/2004.02575

[26] Francesca Mosca and Jose M Such. 2021. *ELVIRA: an Explainable Agent for Value and Utility-driven Multiuser Privacy*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 916–924.

[27] Pradeep K. Murukannaiah, Nirav Ajmeri, Catholijn M. Jonker, and Munindar P. Singh. 2020. New Foundations of Ethical Multiagent Systems. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. IFAAMAS, Auckland, 1706–1710. https://doi.org/10.5555/3398761.3398958 Blue Sky Ideas Track.

[28] Fernando P. Santos, Jorge M. Pacheco, Ana Paiva, and Francisco C. Santos. 2019. Evolution of Collective Fairness in Hybrid Populations of Humans and Agents. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*. AAAI, Honolulu, 6146–6153.

[29] Jéssica S. Santos, Jean O. Zahn, Eduardo A. Silvestre, Viviane T. Silva, and Wamberto W. Vasconcelos. 2017. Detection and Resolution of Normative Conflicts in Multi-Agent Systems: A Literature Survey. *Autonomous Agents and Multi-Agent Systems (JAAMAS)* 31, 6 (Nov. 2017), 1236–1282.

[30] Marc Serramia, Maite López-Sánchez, Juan A. Rodríguez-Aguilar, Manel Rodríguez, Michael Wooldridge, Javier Morales, and Carlos Ansótegui. 2018. Moral Values in Norm Decision Making. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. IFAAMAS, Stockholm, 1294–1302. https://doi.org/10.5555/3237383.3237891

[31] Yoav Shoham and Moshe Tennenholtz. 1997. On the Emergence of Social Conventions: Modeling, Analysis, and Simulations. *Artificial Intelligence* 94, 1–2 (1997), 139–166.

[32] Munindar P. Singh. 2013. Norms As a Basis for Governing Sociotechnical Systems. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 1, Article 21 (Dec. 2013), 23 pages.

[33] Ryan J. Urbanowicz and Will N. Browne. 2017. *Introduction to Learning Classifier Systems*. Springer.

[34] Georg Henrik Von Wright. 1963. *Norm and Action: A Logical Enquiry*. Humanities Press, New York.

[35] Bernard Widrow and Marcian E. Hoff. 1960. *Adaptive Switching Circuits*. Technical Report. Stanford Univ Stanford Electronics Labs.

[36] L. Richard Ye and Paul E. Johnson. 1995. The impact of explanation facilities on user acceptance of expert systems advice. *MIS Quarterly* (1995), 157–172.