Incentive-Compatible Mechanisms for Norm Monitoring in Open Multi-Agent Systems

Natasha Alechina NZA@CS.NOTT.AC.UK
University of Nottingham

Joseph Y. Halpern* HALPERN@CS.CORNELL.EDU
Cornell University

Ian A. Kash IANKASH@MICROSOFT.COM
Microsoft Research

Brian Logan BSL@CS.NOTT.AC.UK
University of Nottingham

Abstract

We consider the problem of detecting norm violations in open multi-agent systems (MAS). We show how, using ideas from *scrip systems*, we can design mechanisms where the agents comprising the MAS are incentivised to monitor the actions of other agents for norm violations. The cost of providing the incentives is not borne by the MAS and does not come from fines charged for norm violations (fines may be impossible to levy in a system where agents are free to leave and rejoin again under a different identity). Instead, monitoring incentives come from (scrip) fees for accessing the services provided by the MAS. In some cases, perfect monitoring (and hence enforcement) can be achieved: no norms will be violated in equilibrium. In other cases, we show that, while it is impossible to achieve perfect enforcement, we can get arbitrarily close; we can make the probability of a norm violation in equilibrium arbitrarily small. We show using simulations that our theoretical results, which apply to systems with a large number of agents, hold for multi-agent systems with as few as 1000 agents—the system rapidly converges to the steady-state distribution of scrip tokens necessary to ensure monitoring and then remains close to the steady state.

1. Introduction

Norms have been widely proposed as a means of coordinating and controlling the behaviour of agents in a multi-agent system (MAS). Norms specify the behaviours that agents should follow to achieve the objectives of the MAS. For example, the designer of a system to allow agents to post content (invitations to tender for work, prices of goods or services, etc.) may wish to ensure that the content posted is relevant, accurate and up to date.

In a MAS where norms must be enforced, the responsibility for enforcing norms lies with a system component termed the *normative organisation* (Dastani, Grossi, Meyer, & Tinnemeier, 2009), which continuously monitors the actions of the agents (and perhaps carries out other tasks on behalf of the MAS). If an action (or the state resulting from an action) would violate or violates a norm,

*. Supported in part by NSF grants IIS-0534064, IIS-0812045, IIS-0911036, and CCF-1214844, and by AFOSR grants FA9550-08-1-0438, FA9550-09-1-0266, and FA9550-12-1-0040, and ARO grant W911NF-09-1-0281.

the action is either prevented, or the agent that performed the action is penalised (incurs a sanction). The effective monitoring of agent actions is therefore key to enforcing norms in a MAS. However, in large systems with many agents, maintaining a separate component to monitor the actions of the agents may involve significant overhead for the MAS.

In this paper, we propose an approach to norm monitoring in open multi-agents systems in which the monitoring of agent actions is performed by the agents comprising the MAS. We term this *decentralised monitoring*. We focus on norms which prohibit certain actions (or the resulting state), for example, posting irrelevant or inaccurate content may be prohibited. The novelty of our approach is that the MAS does not need to bear the cost of paying for monitoring; at the same time we do not need to assume that fines can be levied on the agents who violate the norms and used to pay for monitoring, as done by Fagundes, Ossowski, and Meneguzzi (2014). The latter assumption does not hold for many open systems where the agents can always leave the system and, if needed, rejoin it later under a different identity. Hence, a key issue for our approach is how to incentivise the agents to monitor the actions of other agents. We show how, using ideas from *scrip systems* (Friedman, Halpern, & Kash, 2006; Kash, Friedman, & Halpern, 2012, 2015), we can design incentive-compatible mechanisms where the agents do the monitoring themselves. We can think of scrip as "virtual money" or "tokens". Performing an action costs a token, and detecting violations is rewarded with tokens. The main difference between our setting and that of (Kash et al., 2015) (KFH from now on) is that the agents are not always rewarded after they monitor, but only if they discover a violation. This requires a non-trivial adaptation of the techniques developed by KFH.

We consider two settings. In the first, the *inadvertent setting*, actions that violate a norm are assumed to be inadvertent or unintentional: violating a norm does not increase an agent's utility. In the second, the *strategic setting*, actions that violate the norm are intentional: violating the norm increases the agent's utility, and an agent chooses whether to try to violate the norm. We describe a mechanism that achieves *perfect enforcement* in the inadvertent setting; in equilibrium, all actions are monitored and hence there are no violations of the norm. In the strategic setting, we prove that there can be no equilibrium with perfect enforcement. However, the probability of violations can be made arbitrarily small: for all $\epsilon > 0$, we can design a mechanism where, in equilibrium, the probability of violations is ϵ. We show how all the key steps in our mechanisms can be decentralised, and how our ideas can be extended to *open* systems, where agents may enter and leave the system at any time. We also consider robustness, and show that the mechanisms we propose are m-*resilient*: no coalition of up to m agents can increase their utility through collusion. Finally, we show using simulations that our theoretical results hold for multi-agent systems with as few as 1000 agents. In particular, we show that the system rapidly converges to the steady-state distribution of scrip tokens necessary to ensure monitoring and then remains close to the steady state.

2. Background

Our approach builds substantially on the idea of using a scrip-based mechanism. In this section, we briefly recall that earlier work, both to make the current paper self contained, and to make it easier to point out similarities and differences between the work of KFH and the problems solved and the approach adopted in this paper.

KFH propose a scrip-based mechanism to minimise free riding when agents provide services in a peer-to-peer (P2P) network. In their setting, agents request a service (e.g., downloading a file),

and requests are fulfilled by other agents in the P2P network (e.g., those who have a copy of the file). Fulfilling requests has a utility cost, and the problem is to incentivise provision of the service. Service provision in a P2P network is modelled as a non-cooperative game. The game consists of an infinite number of rounds. At each round, an agent requesting a service is picked uniformly at random. Any other agent is able to satisfy the request with probability $\beta > 0$. If at least one agent volunteers to fulfil the request, one of the volunteers is chosen at random to perform the work, and the requester get a utility of 1. The chosen volunteer loses utility $\alpha < 1$. A standard assumption is made that the agents discount future payoffs by a factor of δ per time unit. This corresponds to the intuition that utility now is worth more than utility tomorrow, and makes it possible to take an agent's utility in an infinite game to be the sum of the utility that the agent obtains at every round of the game. In order to approximate requests being made in parallel, the time between rounds is assumed to be $1/n$, where n is the number of agents in the network. Thus, on average, agents make one request per round. The game $G(n, \delta, \alpha, \beta)$ is the game with n agents, discount factor δ, cost of satisfying a request α, and probability of being able to satisfy a request β. The following additional notation is used:

- p^t denotes the agent chosen to make a request in round t.
- $B_i^t \in \{0, 1\}$ denotes whether agent i can satisfy a request in round t. $B_i^t = 1$ with probability β.
- $V_i^t \in \{0, 1\}$ denotes whether i volunteers in round t ($V_i^t = 1$). Whether an agent volunteers is determined by the agent's strategy (which may depend on p^t).
- $v^t \in \{j \mid B_j^t V_j^t = 1\}$ denotes the agent chosen to satisfy the request. This agent is chosen uniformly at random from the set $\{j \mid B_j^t V_j^t = 1\}$.
- u_i^t denotes agent i's utility in round t.

The utility of agent i in round t is given by

$$u_i^t = \begin{cases} 1 & \text{if } i = p^t, \text{ and } \Sigma_{j \neq i} B_j^t V_j^t > 0 \\ -\alpha & \text{if } i = v^t \\ 0 & \text{otherwise} \end{cases}$$

and the total utility U_i for agent i is $\Sigma_{t=0}^{\infty} \delta^{t/n} u_i^t$. U_i is the discounted total of agent i's utility in each round. Note that the effective discount factor is $\delta^{1/n}$, since an increase in n leads to a shortening of the time between rounds.[1]

Note that we are implicitly assuming that if an agent volunteers and is chosen, then it satisfies the request (assuming it can). As far as Nash equilibrium is concerned, this assumption is made without loss of generality. An agent cannot get any more by volunteering and then not satisfying the request than it can get by simply not volunteering. Separating volunteering from actually doing what the agent volunteered to do becomes more significant when we allow coalitions. We return to this point when we discuss coalitions.

1. As pointed out by KFH (see their Lemma 3.1), there are some technical benefits to taking the effective discount factor to be $1 - (1 - \delta_{\tau(i)})/n$ rather than $\delta^{1/n}$. Our results hold with either choice, so have opted for the simpler approach here.

It is clear that in this setting, free riding is possible: an agent may never volunteer and at the same time gain utility when chosen to make the request. (This relies on some altruistic agents volunteering; otherwise, nobody gains any utility since there are no volunteers to satisfy any requests.) The solution proposed by KFH to reduce free riding uses techniques from *scrip systems*. Scrip is "virtual money". The key idea is to charge scrip tokens for accessing services provided by the P2P system: an agent who has a request satisfied, pays one scrip token to the agent who satisfied the request. It is now impossible to free ride, because scrip can only be earned by volunteering (and satisfying a request); an agent cannot have a request satisfied without having scrip to pay for it.

The amount of scrip tokens that agent i has at time t (i.e., at the beginning of round t) is denoted K_i^t. If i makes a request in round t and it is satisfied, then $K_i^{t+1} = K_i^t - 1$; if some agent $j \neq i$ makes a request in round t that is satisfied by i, then $K_i^{t+1} = K_i^t + 1$; otherwise, $K_i^{t+1} = K_i^t$. Formally, the amount of scrip agent i has at time $t+1$ is

$$K_i^{t+1} = \begin{cases} K_i^t - 1 & \text{if } i = p^t, \Sigma_{j \neq i} B_j^t V_j^t > 0, \text{ and } K_i^t > 0; \\ K_i^t + 1 & \text{if } i = v^t \text{ and } K_{p^t}^t > 0; \\ K_i^t & \text{otherwise.} \end{cases}$$

KFH show that, under some technical assumptions, there is an ϵ-Nash equilibrium where each agent is playing a threshold strategy: an agent volunteers when the amount of scrip tokens they have is below some threshold k. To be precise, a k-threshold strategy S_k is as follows:

$$V_i^t = \begin{cases} 1 & \text{if } i \neq p^t, K_{p^t}^t \geq 0, \text{ and } K_i^t < k; \\ 0 & \text{otherwise.} \end{cases}$$

The proof of the theorem relies on showing that, if agents follow a threshold strategy, then with high probability, the distribution of script tokens will reach a steady state, characterised by maximum entropy. (For more details on scrip dynamics, see Appendix A.)

3. Incentivising Monitoring

In this section, we formalise the "incentivisation game", show that there exists an equilibrium using threshold strategies with both unintentional and intention violations, and show that in this equilibrium all violations (in the case of unintentional violations) or most violations (in the case of intentional violations) are detected. We start by outlining a simple scenario that we use as a running example throughout the remainder of the paper.

We consider a MAS where agents want to post content on the web. There are norms regarding what may be posted; for example, copyrighted images should not be posted, and comments should not be abusive or defamatory. We assume that agents may occasionally submit posts that violate the norm. If such content appears on the web, this may cause considerable harm to the MAS (e.g., it can be fined or sued). Note that here we are viewing the MAS as a whole as an entity that can be fined or sued for norm violations, and which may incur the computational costs associated with monitoring. (As is standard in the MAS literature, we are implicitly assuming that the MAS is an entity independent of the agents that use it, which can, for example, be sued or fined.)

It is therefore in the MAS's interest that submitted posts are checked for compliance with the norm before they appear on the web. We assume that it is possible to check objectively if a particular item of content violates the norm. (For simplicity, we assume that if a post that violates the norm

is checked, the violation will be detected. We can easily modify our approach to handle the case where there is some probability ρ of the violation being caught.) Checking whether a post is 'bad' (violates the norm) requires some work, and incurs a small utility cost. Although checking requires some resources, we assume that if a violation is found, evidence of the violation can be provided that can be checked in negligible time (so we do not need to deal with disputes about whether content violates the norm). If the content does violate the norm, the post is discarded and no violation occurs. We assume a basic infrastructure that ensures that content posted by an agent is signed, and that the digital signatures can be trusted. The signature identifies the agent, and is interpreted as a statement by the agent that the content posted conforms to the norm. Note, however, that the infrastructure does not itself enforce the norm; it serves only to ensure auditability. We believe that such a separation of concerns is good design: the same basic infrastructure may be used by different systems with different norms.

There is a system-level objective that content conform to the norm, but since the cost to the MAS to check all posts may be prohibitive, we would like to distribute the monitoring of posts among the agents that use the system. Just as for the MAS, monitoring incurs a small negative utility for an agent. This means that agents must be appropriately incentivised to monitor. It should be clear that the ideas exemplified by this scenario are applicable far more broadly.

We formalise the posting and monitoring of content for norm violations as a non-cooperative game. This scenario (and the resulting game) is similar to the scenario considered by KFH, but differs in several key respects. In their setting, an agent requests a service, and the problem is to incentivise provision of the service; if the service is not provided, the requesting agent will not be satisfied. Here, it is not necessary that each post be monitored for the posting agent to be satisfied. We assume that, if no agent monitors, it is possible for the posting agent to post and benefit from it; however, a norm violation may be missed. This difference turns out to be not so significant. A more significant difference is that, in our setting, a post may violate the norm. This has no analogue in the setting of KFH, and does complicate matters, as we shall see. Despite this, many of the ideas used by KFH can be used in our setting. In particular, we adopt the idea of using *tokens* as payment for posting and as a reward for monitoring. In order to post, an agent must pay one token; finding a bad post is rewarded by receiving one or more tokens as payment. This encourages agents to volunteer to monitor posts. The exact mechanisms and amounts are discussed below.

We consider two scenarios, one in which bad posts are unintentional, and one in which they are strategic. These correspond to different games. We formalise these two scenarios and our approach for dealing with them in the next two subsections.

3.1 Unintentional Violation

In this scenario, bad posts happen with a constant probability b, but agents are unaware that they are violating the norm when they post something inappropriate. For technical reasons, we assume that b is a rational number (our results hold as long as we use a sufficiently good approximation to the true probability, so this assumption is really without loss of generality). The game in the inadvertent scenario is described by the following parameters:

- a finite set of n agents $1, \ldots, n$;

- the time between rounds is $1/n$;[2]
- in each round t an agent is picked at random to submit a post (we implicitly assume that agents always have something that they want to post);
- probability of a post being bad: b;
- utility of posting (to the agent doing the posting): 1 (we assume that the utility of posting is independent of whether what is posted violates the norm, since violations are unintentional);
- disutility of monitoring (to the agent doing the monitoring): $-\alpha$ (where $0 < \alpha < 1$);
- discount factor: $\delta \in (0, 1)$ (like KFH, we assume that agents discount future payoffs).

The game runs forever. We assume for simplicity that the system is homogeneous: all agents get the same utility for posting (1), the same disutility for monitoring ($-\alpha$), have the same probability of being chosen to post something ($1/n$), and have the same discount factor (δ). Using ideas from (Kash et al., 2012, 2015), we can extend the approach discussed here to deal with different *types* of agents, characterised by different parameters. Some agents may want to post more often; other agents may be less patient (so have a smaller discount factor); etc.

To incentivise monitoring, we use tokens as payment for posting and as a reward for monitoring. An agent must pay one token in order to post. Agents are rewarded with tokens only if they monitor and detect a bad post. We argue below that in order for the system to function successfully (agents being able to post, and some agents always available for monitoring), the 'right' amount to pay for finding a bad posting is $1/b$.[3] This means, in expectation, an agent gets one token for finding a bad posting. Thus, the price of a posting is equal to the expected cost of checking a posting.

We need some additional notation to describe what happens:

- $p^t \in \{1, \ldots, n\}$ is the agent chosen to submit a post in round t;
- $V_i^t \in \{0, 1\}$ denotes whether i volunteers in round t ($V_i^t = 1$). Whether an agent volunteers is determined by the agent's strategy (which may depend on p^t).
- $v^t \in \{0, \ldots, n\} \setminus \{p^t\}$; $v^t = j$ if agent $j \neq p^t$ such that $V_j^t = 1$ is chosen to monitor in round t, and $v^t = 0$ if no one is chosen to monitor in round t;
- $f^t \in \{0, 1\}$; $f^t = 0$ if the content posted in round t is good, $f^t = 1$ if it is bad.

Given that good and bad posts have the same utility (1), the utility of an agent i in round t is:

$$u_i^t = \begin{cases} 1 & \text{if } i = p^t, i \text{ has at least one token, and either } v^t = 0 \text{ or } f^t = 0; \\ -\alpha & \text{if } v^t = i; \\ 0 & \text{otherwise.} \end{cases}$$

Thus, an agent gets utility 1 in round t if it is chosen to submit a post ($p^t = i$), it has at least one token, and either the post is not monitored ($v^t = 0$) or it does not violate the norm ($f^t = 0$). Given the discount factor δ, the total utility U_i for agent i is $\Sigma_{t=0}^{\infty} \delta^{t/n}\, u_i^t$. Note that the number of tokens that an agent has does not affect the agent's utility. However, if an agent requires a token to post

2. The assumption that the time between rounds is $1/n$, which is also made by KFH, makes the analysis easier. It guarantees that, on average, each agent wants to post one message per time unit, independent of the total number of agents.
3. We are implicitly assuming that tokens are divisible into units such that it is possible to transfer $1/b$ tokens.

something, the number of tokens that an agent has does have an indirect impact on utility; if the agent has no tokens, then it will not be able to post anything, and thus will forego the opportunity to get utility 1.

Paying agents $1/b$ for finding a bad posting makes the situation similar to that considered by KFH, where the agent wanting work done pays one token, and the agent doing the work gets one token. However, the fact that in the current setting payment is made only if a post is found to be bad complicates matters. An expected payment of 1 token is not equivalent to an actual payment of 1 token! To understand the issue here, note that the most obvious way to deal with the payment of tokens is to have the agent who wants to post pay one token to the normative organisation, and then have the normative organisation pay $1/b$ tokens to the monitor if a violation is detected. But there are problems with this approach. If monitors have a long run of "bad luck" and do not find postings that violate the norm, agents will have very few tokens on average; on the other hand, if monitors get lucky, and find quite a few postings that violate the norm, agents will end up with many tokens on average. As pointed out by KFH, having both too few or too many tokens will cause problems. Intuitively, with too many tokens, (almost) everyone will have plenty of tokens, so no one will volunteer to monitor; with too few tokens, it will often be the case that the agent who is chosen to post will not have a token to pay for the post.[4] This problem does not occur in the setting of KFH, because there the payment made by the agent requesting a service always matches exactly the payment received by the agent providing the service.

We deal with this problem by having the agents rather than the normative organisation perform the role of the "bank". When agent i wants to post, it pays a randomly chosen agent who has fewer than the maximum number of tokens allowed (see below) 1 token; if an agent j monitors and finds a violation, a randomly chosen agent with at least $1/b$ tokens gives j $1/b$ tokens. This ensures that the number of tokens in 'circulation' remains constant. (Note also that since the agent that pays the monitoring agent is randomly chosen from among those agents with at least $1/b$ tokens, agents may post as long as they have a single token. If the posting agent were to pay the monitoring agent when a violation is detected, agents could not post unless they have $1 + 1/b$ tokens.) It may seem counterintuitive to have a random agent pay the penalty. Since violations are inadvertent and occur randomly, each agent pays the same amount in penalties in expectation as if we had charged the violator; this approach has the advantage of making the system run more smoothly.

We assume that all agents follow a *threshold strategy* when deciding whether to monitor. There is a fixed threshold k such that agents volunteer iff they have fewer than k tokens. It is easy to see that there is an equilibrium in threshold strategies if everyone uses a threshold of 0. In that case, no one ever volunteers to monitor a posting, so everyone gets to post, without monitoring. Of course, no agent has any incentive to deviate from this strategy. On the other hand, this equilibrium runs counter to the purposes of the MAS. We are thus interested in nontrivial equilibria in threshold strategies, where everyone uses a threshold $k > 0$.

KFH show that, in their setting, there is a nontrivial equilibrium in threshold strategies; more precisely, for all $\epsilon > 0$, there exists a δ sufficiently close to 1 and a threshold k such that as long as the discount factor is at least δ, all agents using a threshold of k is an ϵ-Nash equilibrium: no agent can gain more than ϵ by deviating.

We can get a similar result in our setting, using the banking idea above, where the maximum number of tokens any agent may have is $k + 1/b$. To summarise, if an agent i has at least one token

4. The situation would be even worse if the payment for detecting a violation were different from $1/b$; then after some time there would certainly be too few or too many tokens in the system.

and is chosen to submit a post ($i = p^t$), p^t gives a randomly chosen agent with fewer than $k + 1/b$ tokens one more. The posting agent p^t than asks for volunteers to act as monitor. All agents with fewer than k tokens volunteer. If at least one agent volunteers, one, v^t, is chosen at random to act as monitor. If v^t confirms the post conforms to the norm, it is posted. If v^t detects a violation of the norm, then the post is discarded, and a randomly chosen agent with at least $1/b$ tokens gives v^t $1/b$ tokens.

Theorem 1: *For all $\epsilon > 0$, there exist a δ sufficiently close to 1 and an n sufficiently large such that if all n agents have a discount factor $\delta' \geq \delta$, then there exists a k such that the mechanism above with all agents using a threshold of k is an ϵ-Nash equilibrium.*

The proof of Theorem 1 is similar in spirit to that of KFH. A proof sketch can be found in Appendix A.

Corollary 1 *The equilibrium the existence of which is stated in Theorem 1 achieves perfect norm enforcement; all bad posts will be detected.*

Although the theorem applies only if δ is "sufficiently close to 1" and n is "sufficiently large", simulations (reported in Section 5) show that, in practice, the distribution of tokens is reasonably stable with n as small as 1000.[5] These simulations also show that, rather than having one randomly chosen agent pay $1/b$ tokens if a violation is discovered, we can have $\lceil 1/b \rceil$ randomly chosen agents pay 1 token each. The latter approach may be more acceptable in some systems.

3.2 Strategic Violation

We now consider the scenario of strategic violation. In this scenario, we assume that when an agent is chosen to submit a post, it can either submit something good (i.e., that does not violate the norm) or something bad. The parameters of the game are the same as in Section 3.1, except that there is no longer a probability b of a posting being bad (the quality of a posting now becomes a strategic decision), and the utility of a bad posting is no longer 1, but $\kappa > 1$. (We must assume $\kappa > 1$ here, otherwise no agent would ever post anything bad: the utility of doing so is no higher than that of posting something good, and the violation may be detected.)

As before, monitoring agents get paid only if they find a bad post. With these assumptions, it is not hard to show that there does not exist an equilibrium with perfect enforcement.

Theorem 2: *In the setting of strategic violations, the game has no equilibrium with perfect enforcement.*

Proof: Suppose, by way of contradiction, that there exists an equilibrium with perfect enforcement. In this equilibrium, all attempts to make a bad posting are caught. Thus, no agent will use a strategy that gives a positive probability to making a bad posting, for that agent would get higher utility by posting something good instead of something bad. But then no agent would monitor. (It is important that, in our game, when an agent i posts something good, there is nothing that the other players can do to prevent i from getting the utility from the good posting. Also note that even if we had allowed agents to volunteer without monitoring, there would be no benefit to doing so.) But if there is no

5. In fact, simulations show that even with 100–1000 agents, performance is in a range that seems quite tolerable in practice.

actual monitoring, then agents should deviate and make bad postings, since they will not be caught. ∎

Although we cannot achieve perfect enforcement in the strategic setting, we can achieve the next best thing: we can make the probability of a bad posting as low as we want. More precisely, for all $\epsilon, \epsilon' > 0$, there is an ϵ-Nash equilibrium such that the probability of a bad post is ϵ'.

The idea now is that, with some probability, a submitted post will not be checked; there will be no attempt to get volunteers to monitor that posting. Let $c^t = 0$ if there is no monitoring in round t; $c^t = 1$ otherwise. The decision regarding whether to monitor is made *after* the poster submits their post (otherwise, the agent chosen to post will always post something bad in round t if $c^t = 0$). If $c^t = 0$, then whatever the poster submits is posted in that round, whether it is good or bad. As before, if an agent submits a bad post and there is monitoring, we assume that the bad post definitely will be detected and discarded, so the posting agent gets utility 0 in that round. The utility of agent i in round t now becomes

$$u_i^t = \begin{cases} 1 & \text{if } i = p^t \text{ and } f^t = 0; \\ \kappa & \text{if } i = p^t,\ f^t = 1, \text{ and either } c^t = 0 \text{ or } c^t = 1 \text{ and } v^t = 0; \\ -\alpha & \text{if } i = v^t; \\ 0 & \text{otherwise.} \end{cases}$$

Suppose that the normative organisation decides that postings will be monitored with probability $1 - 1/\kappa$. Further suppose that an agent uses a randomised algorithm: with probability β it submits a good posting, and with probability $1 - \beta$ it submits a bad posting. Note that the agent's expected payoff is then $\beta + (1 - \beta)(1/\kappa)\kappa = 1$, independent of β. Thus, we get an equilibrium in the single-shot game if monitoring occurs with probability $1 - 1/\kappa$ and agents submit bad postings with probability β, for all choices of β, provided that there is always guaranteed to be a monitor available. We will show that again there is an equilibrium in threshold policies. As long as there are not too many tokens in the system, there are bound to be some agents with fewer than the threshold number of tokens, so there will be a volunteer.

We assume the designer of the MAS specifies a value β^* (which, intuitively, should be a small 'tolerable' probability of a violation occurring). If a monitor that finds a problem is paid $1/\beta^*$ tokens, then essentially the same type of mechanism as that proposed for the case of unintentional violations will work, provided that we get bad postings with probability exactly β^*. So, perversely, in this setting, while all strategies are equally good for the poster, the MAS actually wants to *encourage* agents to post something bad with probability β^*, so that monitors again get an expected payment of 1 token. The way to do this is for the normative organisation to announce that it will track the number of bad postings, and if the fraction of postings that have been bad up to round t is β, checks will happen with probability $1 - \beta^*/(\beta\kappa)$. Thus, if $\beta = \beta^*$, then checks happen with probability $1 - 1/\kappa$, and we have an equilibrium. Moreover, the payment ($1/\beta$ tokens) is exactly what is needed to ensure that, in equilibrium, monitoring occurs with probability β^*. For if $\beta < \beta^*$, then the check will happen with probability less than $1 - 1/\kappa$, which means that agents will want to make more bad posts. On the other hand, if $\beta > \beta^*$, then monitoring will happen with probability greater than $1 - 1/\kappa$, and agents will want to make fewer bad posts. Thus, in equilibrium, we get bad posts with probability exactly β^*.

To summarise, we have the following mechanism, given a threshold k. If an agent is chosen to post, it submits bad content with probability β^* and good content with probability $1 - \beta^*$. After the agent has decided what to post and made the posting available, the normative organisation

decides whether the posting will be monitored. For an initial period (say 1,000 rounds), a posting is monitored with probability $1 - 1/\kappa$; afterwards, if the fraction of postings that have been discovered to be bad due to monitoring is β and β is more than (say) two standard deviations from β^*, then monitoring occurs with probability $1 - \beta^*/(\beta\kappa)$. If the decision has been made to monitor, and the posting agent has at least one token (so that a post can be made), the posting agent asks for volunteers and all agents with fewer than k tokens volunteer to monitor and one is chosen to be the monitor. As in the case of unintentional violations, if at least one agent volunteers, then the posting agent gives a randomly chosen agent with less than $k + 1/\beta^*$ tokens one more. If the monitor approves the posting, it is posted. If the monitor finds a problem with the posting, then a randomly chosen agent with at least $1/\beta^*$ tokens gives the monitor $1/\beta^*$ tokens.

Theorem 3: *For all $\epsilon > 0$, there exist a δ sufficiently close to 1 and an n sufficiently large such that if all n agents use a discount factor $\delta' \geq \delta$, then there exists a k such that the mechanism above with all agents using a threshold of k is an ϵ-Nash equilibrium.*

A proof sketch can be found in Appendix A.

Note that, in the equilibrium whose existence is stated in Theorem 3, the probability of a bad posting is β^*, as desired.

4. Extensions

In this section, we briefly consider several extensions to our approach. First, we expand our analysis to include the utility of the MAS, and show how social welfare can be maximised by controlling the number of tokens in the system. We then show how the role of the normative organisation can be minimised by decentralising all the key steps in our mechanisms, and how our ideas can be extended to *open* systems where agents may enter and leave the system at any time. Finally, we consider robustness, and show that the mechanisms we propose are m-*resilient*: no coalition of up to m agents can increase their utility through collusion.

4.1 Optimising Social Welfare

Suppose that the MAS gets utility $-C$ (where $C > 0$) for each norm violation. In the setting where bad posts are inadvertent, if there is no monitoring, the MAS suffers an expected loss of utility of bC in each round and the remaining players jointly get 1 unit of utility in each round (because a post is never discarded), so all the agents in the system get an expected utility of $1 - bC$ per round. With monitoring, if we assume for simplicity that an agent always has a token when it wants to post something and there is always a volunteer to monitor, players get a total expected utility of $1 - b - \alpha$ per round. Thus, as long as $C > (b + \alpha)/b$, we maximise social welfare by monitoring. This analysis does not change if occasionally an agent does not have a token to pay for a posting.

In the strategic setting, without monitoring, players will always post inappropriate material, so the total utility will be $\kappa - C$ per round.[6] With monitoring, assuming agents post something bad with probability β^*, as we have seen, the expected utility of an agent who makes a posting is 1 (independent of β^*, so the total expected utility in each round is $1 - (1 - 1/\kappa)\alpha$. So monitoring increases social welfare if $C > \kappa - 1 + (1 - 1/\kappa)\alpha$.

6. Of course, the utility of an inappropriate post may decrease if everyone is posting such material.

Although we have taken the MAS to be a separate entity with its own utility, in some cases the utility of the MAS is best thought of as shared among the individual agents posting content (e.g., in the case of a community or collaboratively maintained website). In this case, we can assume that the players collectively bear the costs of a bad posting. It is then certainly reasonable to assume that monitoring increases social welfare; otherwise the players would simply not bother. But even if the MAS is really an independent entity, whether or not monitoring increases social welfare, the MAS can ensure the monitoring equilibrium is the one that occurs by simply posting this equilibrium and asking agents to play it. As long as sufficiently many agents play it (where "sufficiently many" means that there are enough to ensure that there will always be a monitor), then no agent gains by deviating. Posting the equilibrium is also useful if new agents join the system (see Section 4.3). Note that the MAS can always threaten to shut down if there is no monitoring (which would be in its best interests). Given that, in the remainder of the discussion, we assume that agents play an equilibrium with monitoring.

Social welfare is then maximised if, every time an agent is chosen to submit a post, it has a token to pay for posting and there is an agent who is willing to volunteer to act as a monitor. KFH show that, in their setting, social welfare depends completely on the average number of tokens per agent. Social welfare increases monotonically as the average number of tokens per agent increases, until it reaches a critical point. The key point is that the threshold used in equilibrium *decreases* as the average number of tokens per agent increases. The critical point is the one where the average number of tokens per agent is equal to the threshold. At this point, no one is willing to volunteer, so social welfare drops immediately to 0.

Essentially the same arguments apply in our setting, although we need to be a little careful. In the setting of KFH, after an initial period, no agent has more than the threshold number of tokens (since once they hit the threshold, they stop volunteering). In our setting, since an agent can receive $1/b$ tokens (or $1/\beta^*$ in the case of strategic violations) in one round by discovering a violation, the maximum number of tokens than an agent can have is not k, but $k + 1/b$ (or $k + 1/\beta^*$), where k is the threshold. As long as the average number of tokens per agent is less than $k - (k-1)/n$, then there is always guaranteed to be a volunteer. Thus, social welfare is maximised if the average number of tokens per agent, a, is as large as possible, while still being less than $k-(k-1)/n$, where k is the equilibrium threshold corresponding to a, since a higher average increases the likelihood that an agent who is chosen to post can make a posting.

4.2 Minimising the Role of the Normative Organisation

Although, using our mechanisms, the normative organisation no longer has to monitor postings, it still has a role to play. The normative organisation:

- keeps track of the agents in the system (this is needed to ensure that all agents are aware of a call for volunteers);
- chooses an agent at random with fewer than $k + 1/b$ (or $k + 1/\beta^*$) tokens to receive a token from the agent posting;
- choses an agent at random with at least $1/b$ (or $1/\beta^*$) tokens to pay the monitor if the monitor detects a bad posting;
- keeps track of the number of bad postings and decides whether checking should be carried out in a given round in the mechanism for strategic violations.

There is actually no need for the normative organisation to do any of these things; we can distribute its role among the agents in the system. It is easy for the agents to maintain a list of the agents in the system (think of this as a large email list). Of course, it will have to be updated whenever an agent enters or leaves the system (see also Section 4.3).

Choosing an agent at random from among a group of agents to receive a token from a posting agent, to monitor, or pay for finding a bad post can be done in an incentive-compatible way (i.e., in a way that no agent has any incentive to deviate) using the leader-election algorithm of Abraham, Dolev, and Halpern (2013). Choosing a leader among a set of players is equivalent to choosing one monitor among a set of volunteers. In equilibrium, the algorithm of Abraham et al. (2013) results in each player having an equal chance of being chosen.

To handle the banking process, we can assume that each agent keeps track of how many tokens each agent has. All the transactions can be announced publicly (i.e., who is chosen at random to make a posting, who is chosen at random to get one token, etc.), so everyone can update the amounts appropriately. (We can minimise the communication required by having only a small subset of agents keep track of how many tokens each agent has, or by distributing the role of the bank, so that each agent keeps track of the amounts held by only a few other agents (Vishnumurthy, Chandrakumar, & Sirer, 2003).)

The leader-election algorithm of Abraham et al. can also be used to decide whether checking should be carried out in a given round in the second mechanism. To see how this works, first suppose that κ is an integer that is at most n. Choose a subset of κ agents, including agent 1. There is monitoring if the leader chosen among the κ players is not agent 1. This guarantees that checking is done with probability $1 - 1/\kappa$. We can easily modify this algorithm to compute any rational probability. If the detection of a bad post is announced publicly, agents can also keep track of the number of bad postings in the mechanism for strategic violation case, so that the probability of checking can be modified if needed.

4.3 Open Systems

We assume that our system is open; agents can enter and leave at any time. Dealing with agents leaving is straightforward: they are just deleted from the list of players. Noticing that an agent has left is also straightforward: an agent who does not attempt to make a posting when it is chosen to submit a post, or does not pay tokens to a monitor that detects a bad post when it is chosen to, will be assumed to have left. In the latter case, a new agent can be chosen to make a payment by rerunning the leader election algorithm.

Dealing with agents entering is almost as straightforward. We assume that there is a url where agents can post a message saying that they wish to join. They are then automatically added (by all agents) to the list of members. Like KFH, we assume that a new agent starts out with no tokens, so new entrants to the system cannot post anything. This prevents an agent from joining the system, not doing any monitoring until it runs out of tokens, then leaving the system and rejoining under a different identity. New entrants can acquire tokens by monitoring, or by receiving one token at random from an agent making a post.

There is one other issue that needs to be dealt with: if the number of agents changes significantly, the average number of tokens per agent will change. As noted in Section 4.1, this can have an effect on social welfare. To deal with this, we apply an idea suggested by KFH: we choose a factor F, and multiply the number of tokens that each agent has by F. The factor F is chosen so as to make

the average number of tokens close to the threshold again.[7] As long as all the agents are keeping track of the number of agents in the system (or can consult a public url where this information is available) and we are using public banking, then the agents can detect if the average number of tokens has deviated significantly from the target range, and apply the factor F in a distributed way.

4.4 Robustness

In a Nash equilibrium, no single agent can do better by deviating. However, a coalition may be able to do better by deviating in a coordinated way. In large systems, it seems quite likely that coalitions may form.

Abraham, Dolev, Gonen, and Halpern (2006) define a strategy profile to be m-*resilient* if no group of up to m agents can increase their utility by deviating. A Nash equilibrium is just a 1-resilient profile. It is not hard to show that our basic protocol, combined with "public banking", where all agents keep track of how many tokens are held by each agent, is m-resilient for all m. We assume that agents all keep track of of the number of tokens that each agent has, and compare notes at each step. (If the number of messages exchanged is a concern, it should suffice to compare notes far less often.) If there is disagreement between any pair of agents about the number of tokens held by an agent, then the system simply stops. Clearly, no group of agents gains an advantage by misrepresenting the status of the bank.

We suggested that the leader-election algorithm of Abraham et al. (2013) could be used to distribute the process of choosing a volunteer at random. This algorithm is m-resilient for all $m < n$ in a completely-connected network (which we have implicitly assumed—every agent can communicate with every other agent). Thus, we can put all the pieces together and still get m-resilience for reasonable-sized m. There is a further subtlety: in the strategic case, it becomes important to separate volunteering from monitoring if chosen, so that there are two strategic decisions: whether to volunteer and whether to monitor if chosen. Since postings are not anonymous (recall that we assume that all content is signed), in the strategic case, a coalition may gain by volunteering and then not checking for violations if one of the coalition members is chosen to monitor a bad posting by another coalition member. This means that an agent's post at round t is not checked if either $c^t = 0$ or the monitor chosen is a member of the poster's coalition. If coalitions have size at most m, the latter will happen with probability $\frac{m-1}{n}$. As long as $\frac{m-1}{n} < \beta^*$ we can easily modify the probability that checks are made to take this into account. If we want to deal with coalitions that are large relative to the total number of players, where $\frac{m-1}{n} \geq \beta^*$, we can have posts checked by groups of size h, who all have to agree that there is no violation. In this case, a deviation will be caught unless all the are in the coalition, which will happen with probability $\frac{(m-1)(m-2)\cdots(m-h)}{n^h}$. Again, as long as this probability is less that β^*, our approach will work. (Other minor modifications also need to be made to deal with the presence of more than one monitor; in particular, the payment needs to be split among monitors.) In practice, we expect that it will suffice to have $h \leq 3$. That said, this resilience claim relies on the assumption that the actions and agents are exactly those in our model. In the remainder of this section we discuss some alternatives and our intuitions for how they would affect our results, but we leave a full analysis to future work.

7. If the desired average number of tokens is t, an obvious alternative might be to have each agent entering the system start with t tokens. But, as pointed out by KFH, this leaves the system open to a *sybil attack*: a new player enters the system, makes posts, but never monitors. Once the agent runs out of tokens, it drops out of the system and re-enters the system using a different id. This approach may be less problematic in our setting if agents like to have their identity associated with a post, so there is some loss of utility in leaving the system and re-entering with a new id.

As pointed out by KFH, if it is possible to transfer funds between agents, yet another type of collusion is possible: coalitions can use a lower threshold because they can "insure" each other (i.e., if one agent runs out of tokens to pay for a posting, another agent in the coalition can loan it a token); this allows agents in the coalition to deviate by monitoring less frequently. However, this possibility does not arise in our setting with public banking: there is no procedure for one agent to loan a token to another agent.

If b is relatively small (so that $1/b$ is large) and an agent i could somehow arrange that another j would always be the one to check i's post (and vice versa), then i might be tempted to deliberately make a bad posting, so that j could find it and collect $1/b$. This is not a problem for Nash equilibrium, since it requires collusion on the part of two agents, but it is an issue if we want to prove 2-resilience. However, as our analysis shows, in equilibrium, there will be many volunteers. So as long as the choice among volunteers and the choice of agent required to pay if a violation is found are both made at random, it is not hard to show that this type of deviation will not result in a gain, no matter how large the coalition is, since the probability of someone in the coalition gaining $1/b$ tokens by deliberately deviating is equal to the probability of someone in the coalition having to pay $1/b$ tokens.

We have assumed synchronous communication. While allowing asynchrony would complicate our Nash equilibrium analysis, we do not expect asynchrony to have a significant effect on single-agent deviations. With coalitions, however, more care is needed. In particular, using the leader-election algorithm of Abraham et al. (2013) is only m resilient when $m < n/2$ with asynchronous communication.

In practice, as pointed out by Abraham et al. (2006), we may want even more. We may want to allow for a certain number of "irrational" or "malicious" players, who do not to seem to be acting according to their self-interests. This may be simply because we do not understand what motivates them; that is, we do not know their true utility functions. It may also be because of computer or system problems, or unfamiliarity with the system; these are agents that would act rationally if they could, but something is preventing them from doing so. Note that our proposal for dealing with the banking system, while sufficient to guarantee m-resilience, is not even what Abraham et al. (2006) call *1-immune*: one agent who wants to bring down the system can easily do so by lying about the status of the bank. We can deal with such "malicious" agents by using techniques of *Byzantine agreement* (Fischer, 1983) to get agreement on the bank status, as long as fewer than one-third of the agents are malicious. Indeed, if there is a public-key infrastructure, so that cryptography is available, we can actually use techniques of Byzantine agreement to handle an arbitrary number of malicious agents.

While we think that other aspects of our system are quite robust, and will degrade gracefully in the presence of such irrational or malicious behaviour, we do not have a formal proof. This is a topic that we believe deserves further exploration, since robustness is an important property in practice.

5. Simulations

In this section, we use simulations to quantitatively evaluate the reasonableness of our theoretical results. In particular, we show that (consistent with the results of KFH) the system is well behaved with as few as 1000 agents, converging to the steady-state distribution of tokens quickly and then staying close to it. We also examine an alternative version of our system where we take the payment of $1/b$ tokens from $1/b$ random agents rather than all from a single agent.

We report here results of simulations where there are twice as many tokens as agents (so the average number of tokens per agent—shown to be a key parameter by KFH—is two), agents used a threshold of $k = 5$, and, in the inadvertent setting, the probability of a bad posting is $b = 0.2$. (These choices are arbitrary; similar results hold for other settings of the parameters.) Results below refer to the notion of closeness of distributions of tokens. We represent each distribution as a vector that indicates the fraction of agents with each amount of tokens and then calculate the Euclidean distance between those vectors. We elected to use Euclidean distance to be consistent with the work of KFH. The claims below remain true for a variety of reasonable notions of closeness of distributions.

Figure 1 shows the results of starting the system near the steady-state distribution of tokens predicted by the theory and then running the system for 1 million rounds. It shows that the system stays quite close to the steady-state distribution with as few as 1000 agents. For larger numbers, the system is even closer. With larger numbers of agents, this simulation results in fewer rounds per agent, but an alternate version where we ran it for 1000 rounds per agent (omitted) produced visually indistinguishable results.

For 100 agents (see Figure 2), the maximal distance is 10 times larger than in Figure 1 (below 0.05 rather than below 0.005), which may still be tolerable in practice.

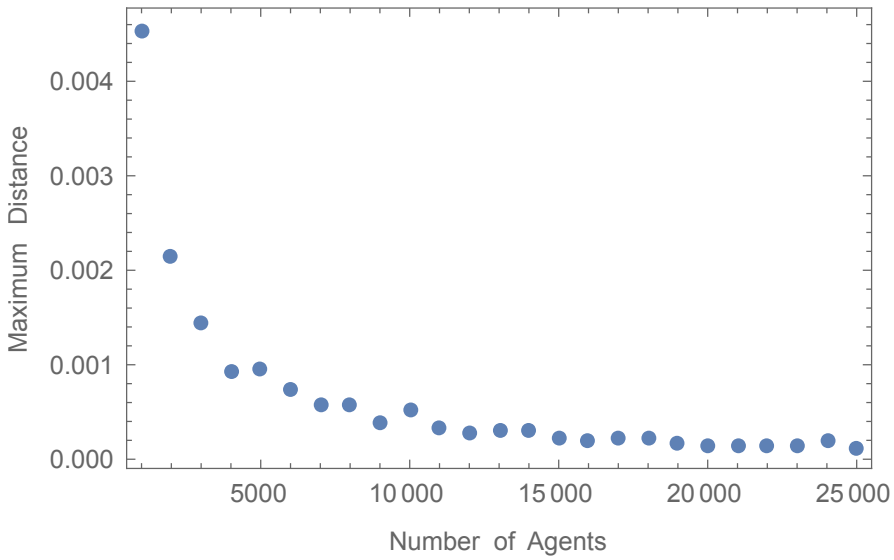

Figure 1: The system stays close to the steady-state distribution of tokens.

In practice, it may be more natural to start the system with some more convenient distribution, such as every agent having the same number of tokens, rather than the maximum-entropy distribution predicted by the theory (see Appendix A).

We simulated starting with the most extreme distribution possible (every agent has either 0 or $k+1/b-1$ tokens) to determine how long it took to get close to the steady-state distribution. Figure 3 shows that even in this unrealistic and extreme case, convergence takes only a small constant number of rounds per agent. (With these parameters, 5 rounds per agent suffice.) Figure 4, which fixes the number of agents at 1000, shows that the required time does not rise significantly even if we ask for very small distances.

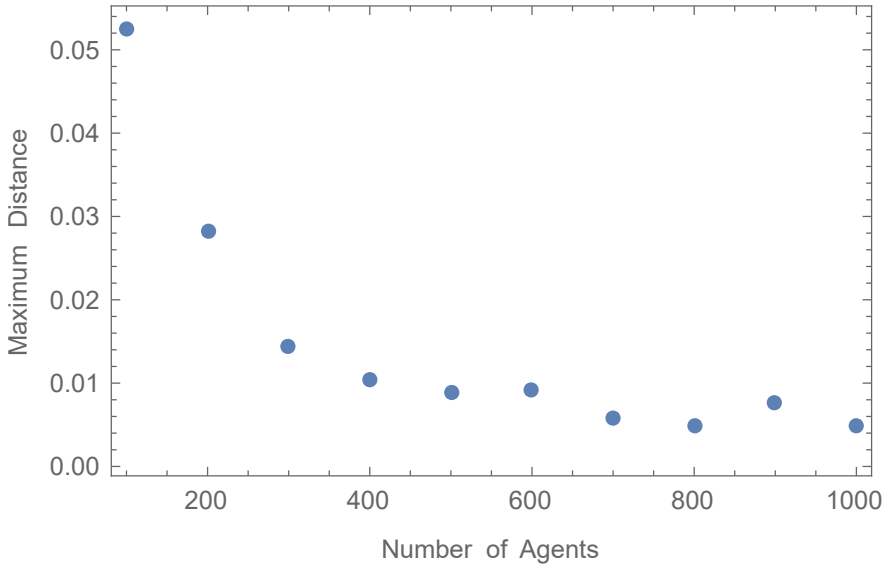

Figure 2: Distribution of tokens for smaller systems.

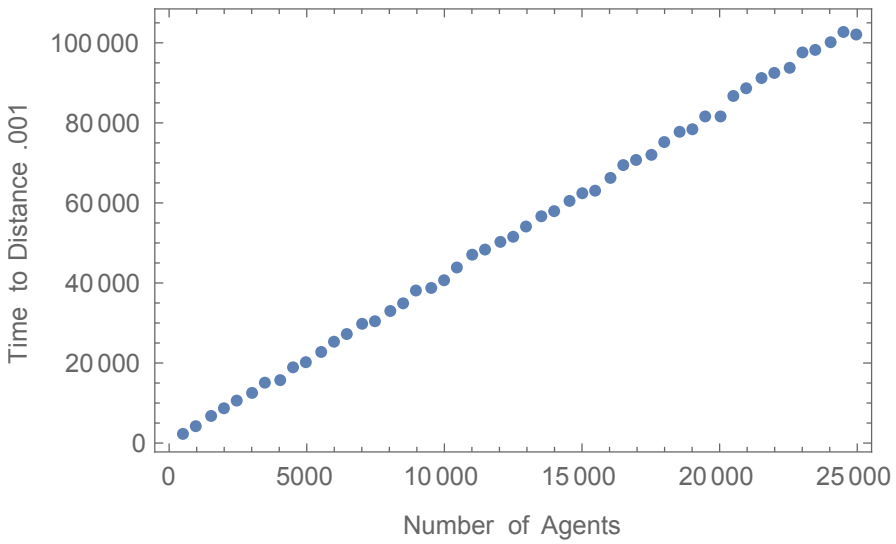

Figure 3: Convergence time to the near the steady-state distribution is a constant number of rounds per agent.

Our model has the somewhat undesirable feature that, when a payment must be made, all $1/b$ tokens are taken from a single agent. It seems more palatable to take a single token from $1/b$ agents instead. However, this invalidates the technique used to calculate the steady-state distribution of wealth. While other work has shown that it is possible to find other ways of calculating the steady-

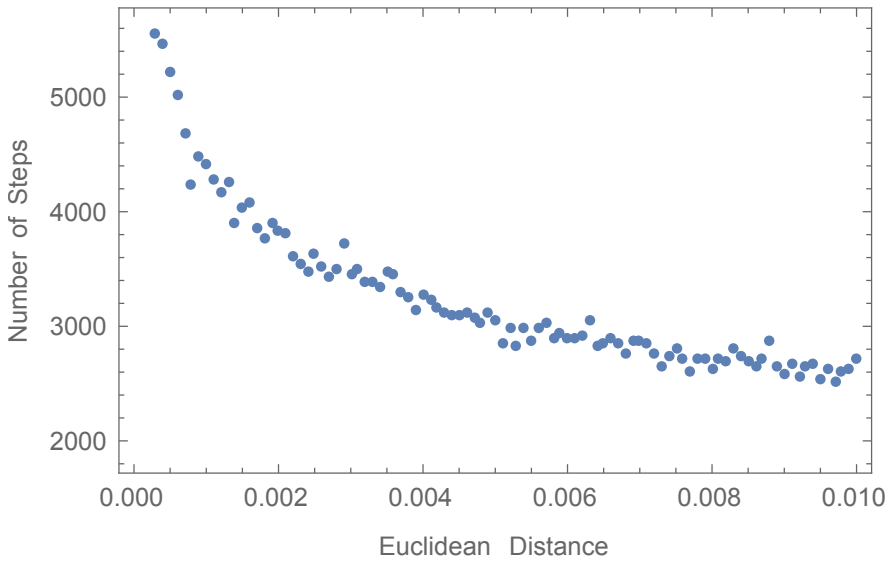

Figure 4: Convergence is fast even for very close distances.

state distribution in other settings where the analogous assumption is not made (Humbert, Manshaei, & Hubaux, 2011), this requires significant effort. Our intuition strongly suggested that charging $1/b$ agents one token each should have no effect on the results, although we could not find a formal proof. Therefore, we decided to empirically validate the existence of a steady-state distribution of tokens. To do so, we ran this alternate mechanism for 100 million steps, sampling the distribution every 20000 steps, and took the sample mean as our estimate of the true steady-state distribution. This change affected the final steady-state distribution more than we expected. The results are shown in Figures 5–8.

The blue dots in Figure 5 show the new steady-state distribution; the original distribution is described by the orange dots. The distance between these distributions is 0.152, which is two orders of magnitude larger than the variation around the steady-state distribution that we saw Figure 1.

We verified that this is in fact the steady-state distribution by rerunning all three simulations and calculating distances from it. The results, shown in Figures 6–8, are essentially the same. The results of this change seem positive overall: agents do not face sudden large drops in their supply of tokens and convergence is, if anything, modestly faster. There are more agents without tokens, but this could be mitigated by using a larger number of tokens.

6. Related Work

Our analysis of the behaviour and incentives of the token economy draws heavily on prior work on scrip systems Friedman et al., 2006; Kash et al., 2012, 2015. We adopt many of KFH's techniques, but extend their analysis to a variant model that applies to our setting. Other work has shown that changing the random volunteer procedure can improve welfare (Johnson, Simchi-Levi, & Sun, 2014) and that this approach still works if more than one agent must be hired to perform

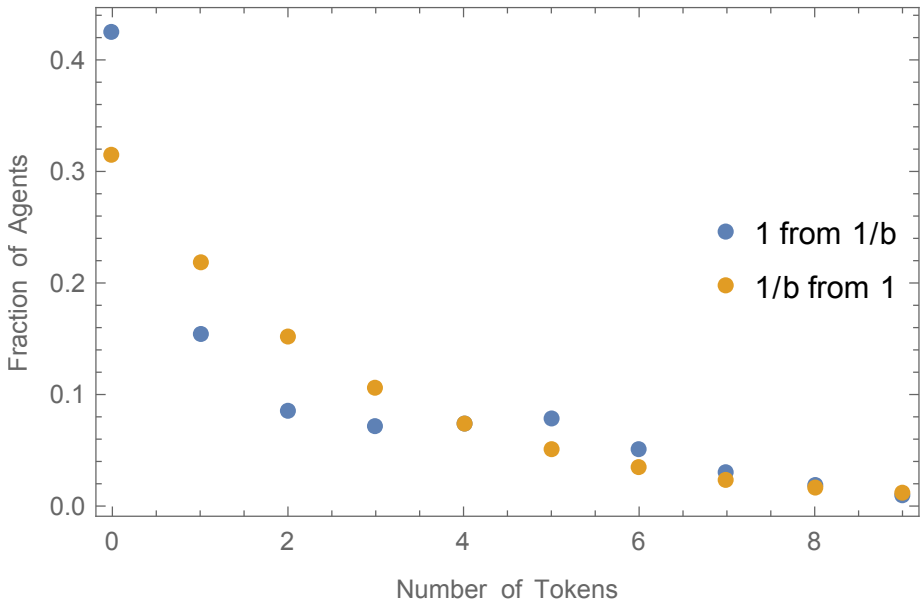

Figure 5: Original and Alternate distribution of tokens.

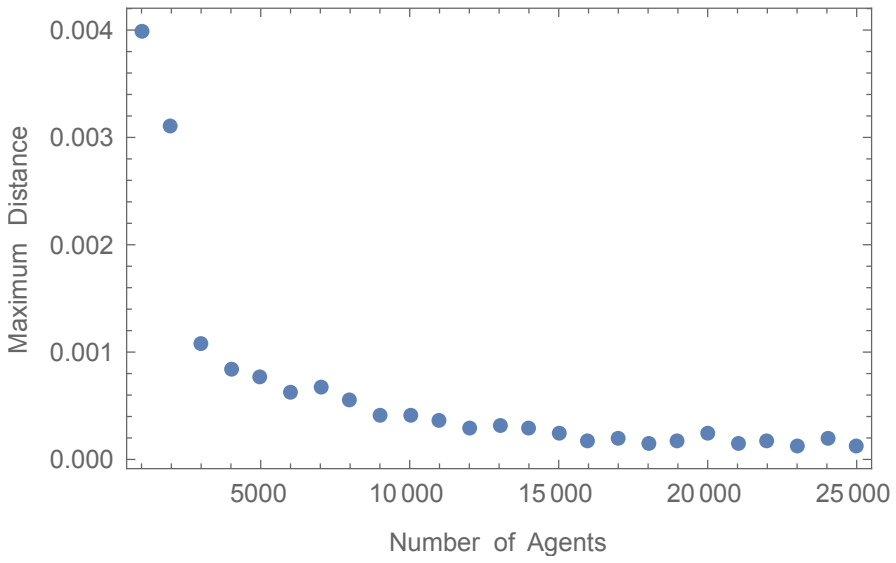

Figure 6: The system stays close to the steady-state distribution of tokens.

work (Humbert et al., 2011). Work from the systems community has looked at practical details such as the efficient implementation of a token bank (Vishnumurthy et al., 2003).

Our approach relies on a number of assumptions. In particular, we assume (1) that if a violation is detected, there is an easily-checkable incontestable 'witness' to the violation, that is, an easy way to convince others that there was indeed a violation (e.g., an inappropriate piece of text) and

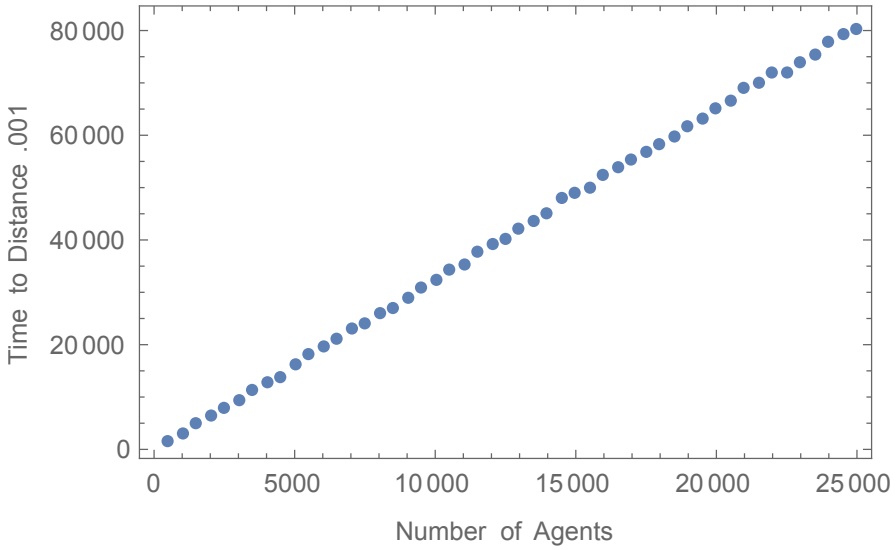

Figure 7: Convergence time to the near the steady-state distribution is a constant number of rounds per agent.

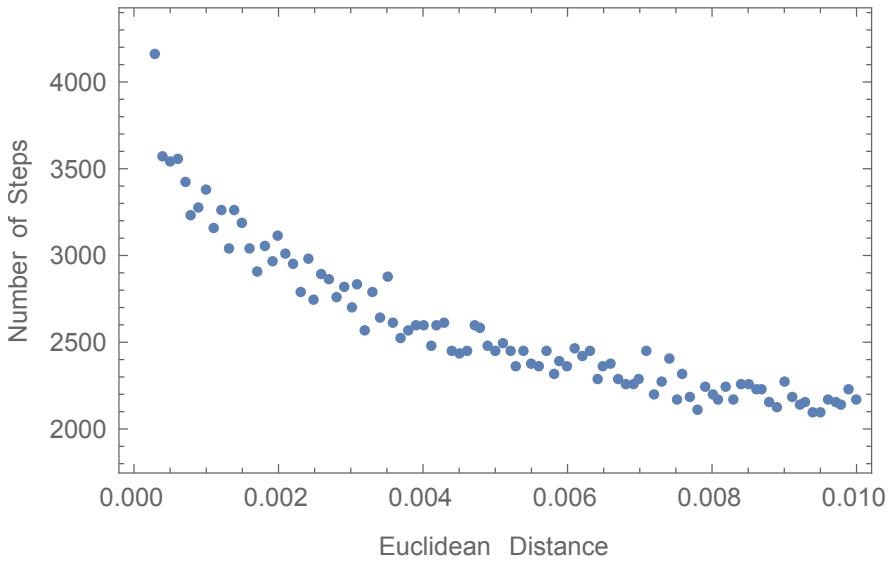

Figure 8: Convergence is fast even for very close distances.

(2) that a violation will be detected if an agent checks a post. In (Alechina, Halpern, Kash, & Logan, 2017) we consider an important setting where these assumptions do not hold. Specifically, we consider a setting where there is no objective definition of what constitutes a norm violation (let

alone an incontestable witness to a violation); rather, there are community standards as to what is acceptable and what constitutes a violation, and agents may legitimately disagree about borderline cases. Such settings arise frequently; think of Wikipedia and discussion fora where people comment on news stories. We show how, using scrip systems and ideas from peer prediction (see, e.g., (Miller, Resnick, & Zeckhauser, 2005)), we can keep the probability of undetected violations (submissions that the majority of the community would consider to be unacceptable) low, while being robust against collusion by the monitoring agents.

Another strand of related work is on game-theoretic models of norm emergence. Axelrod (1986) showed by means of simulations how norms could emerge given simple game rules where players punish each other for violations (and punish players who don't punish violations), and a number of norm enforcement mechanisms with good incentive properties have been analysed (Kandori, 1992; Ellison, 1994). Axelrod's work has been extended by Mahmoud, Griffiths, Keppens, and Luck (2013) to general scenarios and to incorporate learning. de Pinninck, Sierra, and Schorlemmer (2010) proposed a distributed norm enforcement mechanism that uses ostracism as punishment, and showed both analytically and experimentally that it provides an upper bound on the number of norm violations.

There is a significant amount of work in the MAS literature on infrastructures for implementing normative organisations, monitoring for norm violations, and compensating violations through sanctions. One common approach involves the use of additional components or agents to implement the normative organisation. For example, Boella and van der Torre (2003) propose 'defender agents' which detect and punish norm violations. Esteva, Rosell, Rodríguez-Aguilar, and Arcos (2004) propose the use of 'governors' to monitor and regiment message exchanges between agents; each agent is associated with a governor, and all interactions with other agents are filtered by the governor to ensure compliance with norms. Grizard, Vercouter, Stratulat, and Muller (2007), propose an approach in which a separate system of 'controller agents' monitor norm violations and apply reputational sanctions to 'application agents' in a MAS; application agents avoid interactions with other application agents that have low reputation, hence eventually excluding bad agents from the system. Modgil, Faci, Meneguzzi, Oren, Miles, and Luck (2009) propose a two-layer approach, in which 'trusted observers' relay observations of states of interest referenced by norms to 'monitor agents' responsible for determining whether a norm has been violated (a similar approach is described by Criado, Argente, Noriega, & Botti, 2012). Hübner, Boissier, Kitio, and Ricci (2010) describe an approach in which 'organizational agents' monitor interactions between agents mediated by 'organizational artifacts'. ? (?) have used simulation to investigate the effectiveness and costs of paying 'enforcement agents' to monitor norm violations in a wireless mobile grid scenario; the mechanism they propose for rewarding enforcement agents results in a cost to the MAS (in their setting, the telecommunications company), and they assume sanction-based enforcement (agents who violate the norm are punished by the telecommunications company). Testerink, Dastani, and Meyer (2014) consider the problem of monitoring and enforcement by a network of normative organisations in which each normative organisation has only partial information about the actions of the agents and is capable of only local enforcement (by sanctioning).

These approaches are distributed, but the responsibility for monitoring still lies with the normative organisation, and the cost of monitoring is borne by the MAS, either in the cost of running additional system components which monitor and regulate interactions (e.g., Boella & van der Torre, 2003; Esteva et al., 2004) or by paying some agents to monitor the rest (e.g., Balke, De Vos, & Padget, 2013). Fagundes et al. (2014) have explored the tradeoff between the efficiency and cost

of norm enforcement in stochastic environments, to identify scenarios in which monitoring can be funded by sanctions levied on violating agents while at the same time keeping the number of violations within a tolerable level. However, in an open multi-agent system, approaches in which norm enforcement is based on sanctioning (e.g., Grizard et al., 2007; Testerink et al., 2014) may be susceptible to sybil attacks; sanctioned agents may simply leave the system and rejoin under a different id. In contrast, in our approach, the cost of monitoring is borne by the agents using the MAS. Moreover, agents cannot benefit by dropping out and rejoining the system, and monitoring is m-resilient against collusion by monitoring agents.

One issue that arises in previous work on monitoring norms using agents in the MAS is that of 'meta-monitoring', that is, checking that the agents correctly report (all) norm violations; see, for example, (Grossi, Aldewereld, & Dignum, 2007; Hurwicz, 2008). This need for meta-monitoring does not arise in our setting due to the design of the mechanism. We assume that detected violations are accompanied by an easily checkable 'witness', and monitors are paid only if they correctly identify a violation. An agent that inadvertantly or maliciously claims to have detected a violation where there is none will not be paid; a rational agent therefore has no incentive to misreport. Similarly, no single agent can increase its utility by failing to report a violation, since agents are paid only for detected violations, not for checking per se. Nor can coalitions of up to m agents gain from failing to report a violation by a member of the coalition (for reasonable-sized m). If m is large relative to the number of agents in the system, we can have posts checked by groups of h agents who all have to agree that there is no violation. The latter case is better viewed as a form of 'group monitoring' rather than meta-monitoring, as the agents in the group check posts independently, rather than check the monitoring of another agent. As we mentioned earlier, in related work (Alechina et al., 2017), we consider settings where violations are not accompanied by an easily checkable witness. We show how the use of scrip systems can be extended to such settings. The approach that we use in that paper can also be viewed as providing meta-monitoring.

7. Conclusion

We propose an approach to norm monitoring and show that, for sufficiently large MAS, perfect monitoring (and hence enforcement) can be achieved when violations are inadvertent. When violations are strategic, the probability of a norm violation can be made arbitrarily small. This is achieved at no cost to the MAS and without assuming that fines can be used to pay for monitoring. Instead, we achieve perfect or near perfect enforcement using techniques adapted from scrip systems (Kash et al., 2015). Our approach is limited to monitoring norms which forbid single actions (or resulting states). In contrast, approaches such as (Esteva et al., 2004; Modgil et al., 2009; Hübner et al., 2010) are capable of monitoring conditional norms which specify complex behaviours, such as multi-step protocols. We leave extending our approach to such conditional norms to future work.

Appendix A. Sketch of Proof of Theorems 1 and 3

In the setting of KFH, there is someone who wants work done and an agent who is willing to do it. The agent who wants work done gives a token to the agent willing to do it (chosen among volunteers, just as in our setting). In our setting, there is an agent who wants to post something; it plays the same role as the agent who wants work done in the KFH system. But now the posting agent gives a token to a random agent, and the agent performing work (monitoring) gets paid only if he detects

a problem. Moreover, it is not paid by the agent doing the posting, but by a random agent; and it is not paid one token, but $1/b$ (or $1/\beta^*$) tokens. While these seem to be significant differences, the argument used by KFH to prove that there exists an equilibrium in threshold strategies goes through almost without change.

We briefly sketch the key features of the KFH argument here, and the differences in our setting. The argument has two main parts. Suppose that all agents follow the same threshold strategy k. The first part of the argument shows that each individual agent then has an (approximately) optimal strategy, that is, a best response, that is also a threshold strategy. The second part uses this best-response structure to conclude that an equilibrium exists.

So what determines the best response for a given agent? In particular, when should the agent volunteer? Clearly, the one thing that the agent is concerned about is that it will run out of tokens before it is next chosen to monitor, and thus not be able to post when it has the opportunity to do so. The likelihood of this happening depends on how many other volunteers there are each time the agent volunteers. To take an extreme case, if the agent can be sure that it will be the only volunteer, then it is safe to wait until it has one token left. It is unlikely that it will get a chance to make a posting twice before it is able to earn a token. On the other hand, if there will be lots of competition each time it volunteers, then it would be better to use a higher threshold.

So the first step in the proof is to get an accurate estimate of how many volunteers there will be at each step. Note that, once we fix the strategies of all agents, the evolution of the system is entirely determined by the random decisions. Thus, we analyse the Markov chain where the state of the system is characterised by how many tokens each agent has.

For simplicity, we take the Markov chain to consist of all states reachable from some fixed initial state, assuming that all players follow a fixed threshold-k strategy and all players have an integral number of tokens, bounded by k. In each state, it is straightforward to compute the probability of a transition to another state: each agent will be chosen to make a posting with probability $1/n$; we know whether that agent has a token (and thus can post something); we know exactly who will volunteer to monitor (all agents with less than k tokens); and, since a volunteer is chosen at random among the volunteers, we also know the likelihood that an agent will be chosen to monitor. The key observation is that, in the KFH setting, this Markov chain is *reversible*: the probability of a transition from a state s to a state $s' \neq s$ is the same as the probability of the reverse transition from s' to s.

In the KFH setting, reversibility is easy to check: the only such transition that can occur is from a state s where an agent i performs work for an agent j. Then in state s', j has one fewer token than in state s, and i has one more. This transition occurs with probability $1/nm$, where m is the number of agents in state s other than j who have fewer than k tokens: i is chosen in state s with probability $1/m$, and j is chosen with independent probability $1/n$. The transition from s' to s also has probability $1/nm$. Agent i is chosen in s' with probability $1/n$ (and i is guaranteed to have at least one token in s', since it received the token from j) and there are again m volunteers: all the ones that volunteered in state s other than i together with agent j (who must have less than k tokens in state s', since j gave a token to i).

In the inadvertent setting, we again take the Markov chain to consist of all states reachable with positive probability from some fixed starting state, with everyone following the threshold strategy. Reversibility still holds, but we have to work a little harder to show it. For simplicity, we assume that the act of the person wanting to do monitoring paying one token to a random agent with less the maximum number of tokens is distinct from the act where an agent who discovers a violation is

paid $1/b$ tokens by a random agent with at least $1/b$ tokens. In the first transition, the agent i who makes a post ends up with one less token and the random agent j who gets it has one more token. This transition happens with probability $1/nm$, where m is the number of agents other than i who have less than $k+1/b$ tokens. The reverse transition happens with the same probability: j is chosen to post something with probability $1/n$, j can make a posting since it received a token from i, and the agents who have less than $k+1/b$ tokens other than j is still m: all the agents who had less than $k+1/b$ tokens in state s, together with i (who must have less than $k+1/b$ tokens, since it gave a token to j).

Now consider the transition from s to s' where i gives j $1/b$ tokens because j discovered a violation. This transition occurs with probability b/mm', where m is the number of agents other than i with less than k tokens in s (these are the volunteers) and m' is the number of agents with at least $1/b$ tokens. Again the reverse transition happens with the same probability: now j must have at least $1/b$ tokens and i must have less than k tokens, so i is a potential volunteer and j is an agent who can pay i.

The rest of the argument now continues like that of KFH, so we just sketch the details. Clearly the Markov chain is finite, given our assumption that b is rational: we can easily bound the number of possible states, since b is rational and players have at most $k+1/b$ tokens.[8] It immediately follows from reversibility and our assumptions that the states in the Markov chain consist of all states reachable from some initial starting state that Markov chain is *irreducible*: every state is reachable from every other state. Finally, the same arguments as in KFH show that the Markov chain is *aperiodic*: for every state s, there exist no cycles from s to itself such that the gcd of their lengths is 1.

It is well known (Resnick, 1992) that every finite, reversible, aperiodic, and irreducible Markov chain has a *limit distribution* π (where $\pi(s)$ is the fraction of the time that the Markov chain is in state s) and in this limit distribution, all states are equally likely. However, we are not so interested in the probability of a given state; we are interested in the probability of the distribution of tokens; in particular we are interested in the fraction of agents that have less than the threshold number of tokens, because this will tell us how many volunteers there will be. To understand the difference, consider a system with two tokens. There are n states where one agent has both the tokens, and $\binom{n}{2}$ states where two agents have one token each. So although all states are equally likely, the distribution where $2/n$ of the agents have one token and the rest have none is far more likely than the distribution where $1/n$ of the agents have two tokens, and the rest have none.

Since our Markov Chain has the same set of states and the same limit distribution distribution (over states) as that in KFH, we can directly apply their result to conclude as that in KFH, we can directly apply their result to conclude that the distribution that maximises entropy is overwhelmingly more likely than the rest. So, as long as n is sufficiently large, the number of agents with less than the threshold k of tokens is, with arbitrarily high probability, arbitrarily close to γn, where γ is the probability of having less than k tokens according to the maximum entropy distribution.

Once the agent knows how much competition she will face, it is easy to compute a best response, and to show that there is a best response in threshold strategies. We cannot immediately apply the results from KFH, because the exact transition probabilities in the MDP a single agent faces depend on the token payment process, which differs in our setting. Nevertheless, this does not actually

8. Here is where we need the technical assumption that b is rational.

substantively change the argument, so repeating their proof, mutatis mutandis, completes our proof that agents have an ϵ-best response which is a threshold strategy.

For completeness, we give a sketch of this argument; the reader should consult KFH for further details. Consider the MDP an agent would face if there were always exactly γn other agents volunteering. Intuitively, because the more tokens an agent has, the longer it will take to spend them all, there are diminishing returns to acquiring more tokens, and a fixed cost to working, so a threshold policy is optimal. There is one further subtlety in this new setting that did not arise in the KFH setting. In our setting, agents with at least $1/b$ tokens may have that many tokens removed when a violation is discovered by another agent, so it is conceivable that an agent with less than $1/b$ tokens would prefer to not volunteer and be immune to this. However, taking δ sufficiently close to 1 ensures this is not optimal. It remains to show that the error introduced by our assumption that there were always exactly γn other volunteers can be "wrapped up" in the ϵ; the KFH argument for this applies unchanged. In particular, with sufficiently large n, the number of volunteers will be arbitrarily close to this an arbitrarily large fraction of the time, and for δ close to 1, the utility loss in the transient period before the system approaches the limit distribution is an arbitrarily small fraction of the total.

We now turn to the second part of the proof. Let $BR(k)$ denote the agent's best-response threshold strategy if all other agents are using threshold k. KFH show that $BR(k)$ is monotonically increasing in k; moreover, for δ sufficiently close to 1, there exists k such that $BR(k) > k$. To see why, suppose that k is increased. This means that agents are more willing to volunteer, so there is more competition for opportunities. Since it is harder to earn tokens, the value of already having one increases, and so the optimal threshold increases. Similarly, increasing δ means that agents are more willing to do work now for future benefit, so $BR(k)$ can be made arbitrarily large by increasing it. (The same derivations as in KFH show this, mutatis mutandis.)

Given this structure on BR, by Tarski's fixed-point theorem, BR has a fixed point that is greater than k. That fixed point is an ϵ-Nash equilibrium, for if $BR(k) = k$ and all agents are playing a threshold strategy, then playing a threshold of k is an ϵ-best response. This completes the proof sketch of Theorem 1.

The argument in the case of Theorem 3 is quite similar. Here the setting appears more complicated, because agents have two strategic choices: deciding whether or not to post bad material and deciding whether or not to volunteer. However, as shown earlier, our choice of parameters guarantees that an agent's payoff is independent of its strategy regarding whether or not to post bad material. Indeed, if we fix agent i's strategy for deciding whether or not to volunteer and fix the strategies of all other agents, all choices of strategy for deciding whether or not to post bad material, even ones that are history-dependent and correlated with agent i's threshold and the strategies of all other agents, give i the same utility. So, we can assume without loss of generality that, in equilibrium, agent i makes a bad posting with probability β^*.

Now essentially the same argument as that used in the proof of Theorem 1 shows that there is an equilibrium where agents' make their choices regarding volunteering according to some threshold strategy. In particular, the reversibility argument still holds; the probability just needs to be multiplied by $1 - 1/\kappa$. Similarly, the remaining arguments can be repeated, with similar updates to the exact probabilities.

References

Abraham, I., Dolev, D., Gonen, R., & Halpern, J. Y. (2006). Distributed computing meets game theory: robust mechanisms for rational secret sharing and multiparty computation. In *Proc. 25th ACM Symposium on Principles of Distributed Computing*, pp. 53–62.

Abraham, I., Dolev, D., & Halpern, J. Y. (2013). Distributed protocols for leader election: A game-theoretic perspective. In Afek, Y. (Ed.), *Distributed Computing - 27th International Symposium, DISC 2013, Jerusalem, Israel, October 14-18, 2013. Proceedings*, Vol. 8205 of *Lecture Notes in Computer Science*, pp. 61–75. Springer.

Alechina, N., Halpern, J. Y., Kash, I. A., & Logan, B. (2017). Incentivising monitoring in open normative systems. In Singh, S. P., & Markovitch, S. (Eds.), *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*, pp. 305–311, San Francisco, USA. AAAI, AAAI Press.

Axelrod, R. (1986). An evolutionary approach to norms. *The American Political Science Review*, *80*(4), 1095–1111.

Balke, T., De Vos, M., & Padget, J. (2013). Evaluating the cost of enforcement by agent-based simulation: A wireless mobile grid example. In Boella, G., Elkind, E., Savarimuthu, B. T. R., Dignum, F., & Purvis, M. K. (Eds.), *PRIMA 2013: Principles and Practice of Multi-Agent Systems*, Vol. 8291 of *Lecture Notes in Computer Science*, pp. 21–36. Springer Berlin Heidelberg.

Boella, G., & van der Torre, L. W. N. (2003). Norm governed multiagent systems: The delegation of control to autonomous agents. In *2003 IEEE/WIC International Conference on Intelligent Agent Technology (IAT 2003)*, pp. 329–335. IEEE Computer Society.

Criado, N., Argente, E., Noriega, P., & Botti, V. J. (2012). A distributed architecture for enforcing norms in open MAS. In Dechesne, F., Hattori, H., ter Mors, A., Such, J. M., Weyns, D., & Dignum, F. (Eds.), *Advanced Agent Technology - AAMAS 2011 Workshops, AMPLE, AOSE, ARMS, DOCM3AS, ITMAS, Taipei, Taiwan, May 2-6, 2011. Revised Selected Papers*, Vol. 7068 of *Lecture Notes in Computer Science*, pp. 457–471. Springer.

Dastani, M., Grossi, D., Meyer, J.-J. C., & Tinnemeier, N. (2009). Normative multi-agent programs and their logics. In Meyer, J.-J. C., & Broersen, J. (Eds.), *Knowledge Representation for Agents and Multi-Agent Systems*, Vol. 5605 of *Lecture Notes in Computer Science*, pp. 16–31. Springer Berlin Heidelberg.

de Pinninck, A. P., Sierra, C., & Schorlemmer, W. M. (2010). A multiagent network for peer norm enforcement. *Autonomous Agents and Multi-Agent Systems*, *21*(3), 397–424.

Ellison, G. (1994). Cooperation in the prisoner's dilemma with anonymous random matching. *Review of Economic Studies*, *61*, 567–588.

Esteva, M., Rosell, B., Rodríguez-Aguilar, J. A., & Arcos, J. L. (2004). AMELI: An agent-based middleware for electronic institutions. In *3rd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004), 19-23 August 2004, New York, NY, USA*, pp. 236–243. IEEE Computer Society.

Fagundes, M. S., Ossowski, S., & Meneguzzi, F. (2014). Analyzing the tradeoff between efficiency and cost of norm enforcement in stochastic environments. In *21st European Conference on Artificial Intelligence (ECAI 2014)*, pp. 1003–1004. IOS Press.

Fischer, M. J. (1983). The consensus problem in unreliable distributed systems. In Karpinski, M. (Ed.), *Foundations of Computation Theory*, Lecture Notes in Computer Science, Volume 185, pp. 127–140. Springer, Berlin/New York.

Friedman, E. J., Halpern, J. Y., & Kash, I. A. (2006). Efficiency and Nash equilibria in a scrip system for P2P networks. In *Proceedings 7th ACM Conference on Electronic Commerce (EC-2006)*, pp. 140–149. ACM.

Grizard, A., Vercouter, L., Stratulat, T., & Muller, G. (2007). A peer-to-peer normative system to achieve social order. In Noriega, P., Vzquez-Salceda, J., Boella, G., Boissier, O., Dignum, V., Fornara, N., & Matson, E. (Eds.), *Coordination, Organizations, Institutions, and Norms in Agent Systems II*, Lecture Notes in Computer Science, pp. 274–289. Springer Berlin Heidelberg.

Grossi, D., Aldewereld, H., & Dignum, F. (2007). Ubi lex, ibi poena: Designing norm enforcement in e-institutions. In Noriega, P., Vzquez-Salceda, J., Boella, G., Boissier, O., Dignum, V., Fornara, N., & Matson, E. (Eds.), *Coordination, Organizations, Institutions, and Norms in Agent Systems II*, Vol. 4386 of *Lecture Notes in Computer Science*, pp. 101–114. Springer Berlin Heidelberg.

Hübner, J., Boissier, O., Kitio, R., & Ricci, A. (2010). Instrumenting multi-agent organisations with organisational artifacts and agents. *Autonomous Agents and Multi-Agent Systems*, *20*, 369–400.

Humbert, M., Manshaei, H., & Hubaux, J.-P. (2011). One-to-n scrip systems for cooperative privacy-enhancing technologies. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pp. 682–692.

Hurwicz, L. (2008). But who will guard the guardians?. *American Economic Review*, *98*(3), 577–585.

Johnson, K., Simchi-Levi, D., & Sun, P. (2014). Analyzing scrip systems. *Operations Research*, *62*(3), 524–534.

Kandori, M. (1992). Social norms and community enforcement. *Review of Economic Studies*, *59*, 63–80.

Kash, I. A., Friedman, E. J., & Halpern, J. Y. (2012). Optimizing scrip systems: crashes, altruists, hoarders, sybils and collusion. *Distributed Computing*, *25*(5), 335–357.

Kash, I. A., Friedman, E. J., & Halpern, J. Y. (2015). An equilibrium analysis of scrip systems. *ACM Transactions on Economics and Computation*, *3*(3). Article No. 13.

Mahmoud, S., Griffiths, N., Keppens, J., & Luck, M. (2013). Norm emergence through dynamic policy adaptation in scale free networks. In Aldewereld, H., & Sichman, J. S. (Eds.), *Coordination, Organizations, Institutions, and Norms in Agent Systems VIII - 14th International Workshop, COIN 2012, Revised Selected Papers*, Vol. 7756 of *Lecture Notes in Computer Science*, pp. 123–140. Springer.

Miller, N., Resnick, P., & Zeckhauser, R. (2005). Eliciting informative feedback: The peer-prediction method. *Management Science*, *51*(9), 1359–1373.

Modgil, S., Faci, N., Meneguzzi, F. R., Oren, N., Miles, S., & Luck, M. (2009). A framework for monitoring agent-based normative systems. In Sierra, C., Castelfranchi, C., Decker, K. S., & Sichman, J. S. (Eds.), *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, pp. 153–160, Budapest, Hungary. IFAAMAS.

Resnick, S. I. (1992). *Adventures in Stochastic Processes*. Birkhauser.

Testerink, B., Dastani, M., & Meyer, J.-J. (2014). Norms in distributed organizations. In Balke, T., Dignum, F., van Riemsdijk, M. B., & Chopra, A. K. (Eds.), *Coordination, Organizations, Institutions, and Norms in Agent Systems IX*, Lecture Notes in Computer Science, pp. 120–135. Springer International Publishing.

Vishnumurthy, V., Chandrakumar, S., & Sirer, E. G. (2003). KARMA: a secure economic framework for peer-to-peer resource sharing. In *First Workshop on Economics of Peer-to-Peer Systems (P2PECON)*.