MANAGING TRUST IN DIFFUSION ADAPTIVE NETWORKS WITH MALICIOUS AGENTS

Konstantinos Ntemos[*], *Nicholas Kalouptsidis*[*], *and Nicholas Kolokotronis*[†]

[*] Dept. of Informatics and Telecomm., University of Athens, 15784 Athens, Greece
[†] Dept. of Informatics and Telecomm., University of Peloponnese, 22100 Tripolis, Greece

ABSTRACT

In this paper, we consider the problem of information sharing over adaptive networks, where a diffusion strategy is used to estimate a common parameter. We introduce a *new* model that takes into account the presence of both selfish and malicious intelligent agents that adjust their behavior to maximize their own benefits. The interactions among agents are modeled as a stochastic game with incomplete information and partially observable actions. To stimulate cooperation amongst selfish agents and thwart malicious behavior, a trust management system relying on a voting scheme is employed. Agents act as independent learners, using the Q–learning algorithm. The simulation results illustrate the severe impact of falsified information on estimation accuracy along with the noticeable improvements gained by stimulating cooperation and truth–telling, with the proposed trust management mechanism.

***Index Terms*—** Trust management, multi–agent systems, independent learning, adaptive networks, voting schemes.

1. INTRODUCTION

In the past years, numerous works have focused on analyzing the behavior of autonomous agents, with possibly conflicting goals, in the context of wireless network security using game theory [4]. To stimulate cooperation among selfish agents, or enforce malicious agents to comply with a predefined policy, *trust* or *reputation* mechanisms are employed; an agent's actions history is summarized into a single value, which is commonly referred to as *trust* or *reputation*, depending on the context. An important class of trust mechanisms are based on *reciprocity* principles, where an agent can be punished for the actions taken towards its peers according to a prescribed punishment strategy. Reciprocity mechanisms are subdivided into *direct* and *indirect* [2, 8, 10] —in direct, the interactions history between two agents determines how they will interact in the future. In a highly dynamic network, where the same pair of agents rarely interacts, indirect reciprocity is used, and an agent's trust depends upon its past interactions with all the agents. Such schemes have important applications in packet forwarding [2, 8], trust evaluation [5], and intrusion detection [1], amongst others. The application of game theory to foster cooperation for distributed in–network processing, has only recently received attention. To be more precise, a reputation mechanism was proposed in [11] to give incentives to selfish agents to cooperate in estimating a common parameter. The authors assume autonomous selfish agents whose interaction is modeled as on one–shot game (the agents are randomly paired); moreover, the nodes are bounded–rational, a notion common in multi–agent systems [7].

In this paper we investigate the distributed parameter estimation problem under the presence of both selfish and malicious agents using the rich framework of *stochastic game theory*. In the game model, we let agents have incomplete information and partially observe actions, as they do not know whether their peers are malicious or not and they are uncertain about their actions. To combat uncertainty, we allow agents utilize independent estimates of the unknown parameter from their neighborhood in order to detect the action taken by a particular neighbor. We introduce a trust scheme that estimates an agent's trustworthiness based on its past interactions, and formulate the agents as independent learners. Simulation results show how agents benefit from the proposed mechanism in terms of estimation accuracy, which is due to the adherence with high probability of all agents to honest policies.

The rest of the paper is organized as follows. In Section 2 we present the proposed network model and the system design follows in Section 3. Simulation results and concluding remarks are given in Sections 4, 5 respectively.

2. SYSTEM MODEL

The basic constituents of the system model are introduced in the following subsections. Hereinafter, letters in boldface are column vectors $\boldsymbol{x}$ if lowercase, or matrices $\boldsymbol{X}$ otherwise; $\boldsymbol{x}^{\intercal}$ denotes the transpose. Lowercase letters are used for scalars and variables, whereas uppercase letters for sets. $\mathcal{N}(\mu, \sigma^2)$ is the Gaussian distribution with mean μ and variance σ^2.

Network structure. Consider a set of n agents with sensing,

This work was supported by the research projects ART–IN–SPACE and HANDiCAMS. The project ART–IN–SPACE is co–financed by the European Union (European Social Fund) and Greek national funds through the operational program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) — Research funding program ARISTEIA I, grant no. 1115. The project HANDiCAMS acknowledges the financial support of the Future and Emerging Technologies (FET) programme, within the 7th Framework Programme for Research of the European Commission, under FET–Open grant no. 323944.

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

computing, and communication capabilities coexisting in a network; let $\mathcal{G} = \langle N, E\rangle$ be the associated graph, where the agents are labeled by the elements of N and E indicates direct links. The nodes linked to $i \in N$ (incl. node i) form its neighborhood that is denoted by N_i; we also define $n_i = |N_i|$ and $N_i^* = N_i \setminus \{i\}$. Agents are distinguished into three types: malicious (M), selfish (S) and honest (H) and they pursue different goals depending on their type $\tau \in T = \{\mathsf{M}, \mathsf{S}, \mathsf{H}\}$. In doing so, they employ private information collected by their own sensors and information received by the neighbors.

Let the unknown parameters of the network be packed into the vector $\boldsymbol{\theta} \in \mathbb{R}^m$. Selfish and honest nodes seek to estimate $\boldsymbol{\theta}$. The difference is that honest nodes follow some predetermined policy, while the selfish aim at estimating $\boldsymbol{\theta}$ at the lowest possible cooperation cost. On the other hand, malicious nodes seek to estimate *only* a part, say $\boldsymbol{\theta}_{\mathsf{L}} \in \mathbb{R}^l$, of $\boldsymbol{\theta} = (\boldsymbol{\theta}_{\mathsf{L}}\ \boldsymbol{\theta}_{\mathsf{R}})$, and to impede the other nodes from achieving their estimation goal[1]. Agents a priori recognize the value of information received by others in enhancing estimation performance. However, the types of the agents are unknown. Thus, in the presence of malicious nodes, agents need to decide if the information received by other agents can be trusted or not. Hence, the agents need to detect other agents' actions and decide who they should trust. They are assisted in these decisions by a *trust management system* (TMS) that computes and updates the trust values of all agents, which are then made available to the network.

Private information and data sharing. Nodes collect information by means of their own sensors and their neighbors. At time t, node i has access to the regressor $\boldsymbol{u}_i(t) \in \mathbb{R}^m$ and an observation $d_i(t) \in \mathbb{R}$. These data constitute the *private* information of i and are linked with the true parameter via

$$d_i(t) = \boldsymbol{u}_i(t)^{\mathsf{T}}\boldsymbol{\theta} + v_i(t) \tag{1}$$

where $v_i(t) \sim \mathcal{N}(0, \sigma_i^2)$ is the measurement noise. Besides the above sensing data, node i receives information from its neighbors; let $\boldsymbol{\zeta}'_j(t)$ denote the information *received* from agent $j \in N_i^*$. The vector $\boldsymbol{\zeta}'_j(t)$ augments agent's i private information with additional data which, if trustworthy, can improve estimation performance compared to what is achieved using private data only. If $\boldsymbol{\zeta}_j(t)$ is node's j estimate of $\boldsymbol{\theta}$ at time t, each received signal satisfies the following transmission model

$$\boldsymbol{\zeta}'_j(t) = a_{ji}(t)^2 \cdot \boldsymbol{\zeta}_j(t) - \tfrac{1}{2}a_{ji}(t)(1 - a_{ji}(t)) \cdot \boldsymbol{\eta}_j(t) \tag{2}$$

where for simplicity we assume noise–free communications. The parameter $a_{ji} \in \{0, \pm 1\}$ denotes the transmission decision of agent i at time t. Note that $a_{ji} = 0$ corresponds to the selfish action since it implies $\boldsymbol{\zeta}'_j(t) = \mathbf{0}$, that is, no data is sent. The honest action $a_{ji} = 1$ implies that the true estimate $\boldsymbol{\zeta}_j(t)$ is sent, while the malicious action $a_{ji} = -1$ generates the erroneous estimate $\boldsymbol{\zeta}_j(t) + \boldsymbol{\eta}_j(t)$. In order to degrade the estimation performance of node i, node j adds Gaussian noise $\boldsymbol{\eta}_j(t) \sim \mathcal{N}(\boldsymbol{\mu}_j, \rho_j^2\boldsymbol{J})$, where we let $\boldsymbol{\mu}_j = (\mathbf{0}_{\mathsf{L}}\ \boldsymbol{\mu}_{j,\mathsf{R}})$ and $\boldsymbol{J} = \mathrm{diag}(\mathbf{0}_{\mathsf{L}}\ \mathbf{1}_{\mathsf{R}})$ due to the assumption that malicious nodes are only interested in estimating $\boldsymbol{\theta}_{\mathsf{L}}$. We further assume that the action $a_{ji} = -1$ can be taken by malicious agents only.

Decisions, states and rewards. The model we consider for agents' interactions deviates from the commonly used random pair–matching [2, 8, 9, 11], and falls into the category of multi–agent systems [7]. The agents take transmission and reception decisions; the transmission decisions are represented by the transmission actions a_{ij} and depend on the agent's type τ, since the set of possible transmission actions is

$$A(\tau) = \begin{cases} \{-1, 0, 1\}, & \text{if } \tau = \mathsf{M}, \\ \{0, 1\}, & \text{otherwise.} \end{cases} \tag{3}$$

At the receiver side, the agents must decide whether each received signal is true or intensionally falsified. This decision relies on the trustworthiness of the agents sharing estimates, and such information is included in the state.

The state $\boldsymbol{s} = (s_i\ \boldsymbol{s}_{-i})$ consists of the trust values of node i and its neighbors $\boldsymbol{s}_{-i} = (s_j)_{j \in N_i^*}$; the closer to 1 the value of $s_i \in B = \{0, 1/2, 1\}$ is, the more trusted node i is considered. The set $\mathbb{S} = \{s_i\}_{i \in N}$ of all trust values is assumed to be *shared* information and becomes available through the TMS. The state transition probabilities $\Pr(\boldsymbol{s}' \,|\, \boldsymbol{s}, \boldsymbol{a}_i, \boldsymbol{a}_{-i}, \boldsymbol{\tau})$ of the stochastic game are also defined by the TMS, where $\boldsymbol{s}', \boldsymbol{s}$ are the new and current state respectively, $\boldsymbol{\tau} = (\tau_i\ \boldsymbol{\tau}_{-i})$ gathers the types of nodes in N_i, while we have $\boldsymbol{a}_i = (a_{ij})_{j \in N_i^*}$ and $\boldsymbol{a}_{-i} = (a_{ji})_{j \in N_i^*}$.

To simplify the notation, the dependence of variables on time t is omitted whenever it is clear from the context. The *instantaneous reward* of agent i at time t is decomposed as

$$R_{i,t}(\boldsymbol{s}, \pi_i, \boldsymbol{a}_{-i}, \tau_i) = f_{i,t}(\boldsymbol{s}, \boldsymbol{a}_{-i}) + g_{i,t}(\boldsymbol{s}, \pi_i, \tau_i) \tag{4}$$

where $\pi_i(t)$ is the transmission policy followed by agent i in time t (i.e. transmission actions as functions of the state), $f_{i,t}$ equals the profit gained by receiving information from neighboring nodes, while $g_{i,t}$ is associated with the gains and costs resulting from transmission actions. Each agent i wants to maximize the discounted sum of instantaneous rewards over a finite time horizon k

$$\max_{\pi_i(0), \ldots, \pi_i(k-1)} \sum_{t=0}^{k-1} \delta_i^t\, \mathbb{E}[R_{i,t}(\boldsymbol{s}(t), \pi_i(t), \boldsymbol{a}_{-i}(t), \tau_i)] \tag{5}$$

where $\delta_i \in (0, 1)$ is a *discount factor* that defines the relative importance given by agent i to short–term rewards.

The above formulation constitutes a stochastic game with incomplete information since the types of the agents are unknown and their actions are not fully observable.

[1] Malicious behavior can be found in e.g. cognitive radio networks, where a malicious agent could aim at reporting erroneous spectrum occupancy measurements so that only he/she can exploit an available spectrum hole.

3. SYSTEM DESIGN

The proposed system design encompasses a sequence of steps timed as follows:

a. Selection of transmission policy;
b. Clustered action detection;
c. Trusted adapt–then–combine (TATC) strategy for adaptive parameter estimation; and
d. Voting and trust update.

Details of the above protocol are given next.

Selection of transmission policy. Since the *state–action space* can be large and information is incomplete, the stochastic game is hard to solve. To cope with these issues, we confine to the set $\mathit{\Pi} = \{\mathit{\Pi}(\tau) : \tau \in T\}$ of admissible policies, which is a subset of *Markovian policies*. Each type of agent is associated with a *desired policy*, which is a mapping $\pi_\tau : B \to A(\tau)$, $\tau \in T$. We assume that the honest agents behave as prescribed by the TMS, by helping only the trusted agents; thus, we let the agents take the same action towards neighbors with the same trust value, and we have $\pi_{\mathsf{H}}(s) = 1$ if $s = 1$, and $\pi_{\mathsf{H}}(s) = 0$ otherwise. This behavior is similar to that of *obedient agents* in [9]. Thus, $\mathit{\Pi}(\mathsf{H}) = \{\pi_{\mathsf{H}}\}$.

The selfish agents aim at enjoying the cooperation benefits while minimizing their cooperation costs (called *free–riders* in [9]). Hence, their desired policy is not to share information, implying that $\pi_{\mathsf{S}}(s) = 0, \forall s \in B$. In a similar fashion, the desired policy of malicious agents is sending false estimates, according to (2), therefore $\pi_{\mathsf{M}}(s) = -1, \forall s \in B$. Malicious and selfish agents are *rational* and thus can take an alternative policy if that is more beneficial; that is, selfish and malicious agents are *strategic*, and explore between the desired policy and π_{H} using the learning algorithm. Therefore $\mathit{\Pi}(\mathsf{S}) = \{\pi_{\mathsf{S}}, \pi_{\mathsf{H}}\}$ and $\mathit{\Pi}(\mathsf{M}) = \{\pi_{\mathsf{M}}, \pi_{\mathsf{H}}\}$.

Instantaneous reward. The function $f_{i,t}$ in (4) represents the gains offered by cooperation in estimation performance, as perceived by agent i. It is given by

$$f_{i,t}(\boldsymbol{s}, \boldsymbol{a}_{-i}) = \|\boldsymbol{\theta}_i(t) - \boldsymbol{\zeta}_i(t)\|_2^2 \tag{6}$$

where $\boldsymbol{\theta}_i(t)$ is node's i estimate of $\boldsymbol{\theta}$ at time t when using the estimates received by its neighbors. If most neighbors choose the honest policy π_{H}, then a rational malicious or selfish node could see that it would possibly be beneficial to be considered trusted in a certain neighborhood. The function $g_{i,t}$ includes communication costs $c \in \mathbb{R}^+$ to send $\boldsymbol{\zeta}'_i(t)$ to its neighbors, and *illegal gains* $e > c$ if acting maliciously; it is given by

$$g_{i,t}(\boldsymbol{s}, \pi_i, \tau_i) = \begin{cases} e\,s_i - c, & \text{if } \pi_i = \pi_{\mathsf{M}},\ \tau_i = \mathsf{M} \\ -c, & \text{if } \pi_i = \pi_{\mathsf{H}} \\ 0, & \text{if } \pi_i = \pi_{\mathsf{S}},\ \tau_i = \mathsf{S} \end{cases} \tag{7}$$

where the illegal gains represent the profit obtained by either degrading the estimation performance or the trust value of its neighbors. Note that $g_{i,t}$ depends on s_i as well.

Clustered action detection. Recall that the actions of the agents are not observable. This contrasts many works in the trust literature that either assume observable actions or that some *monitoring* mechanism exists allowing perfect action detection (*see* [8]). Each node i decides on the transmission actions of its neighbors using its own estimate $\boldsymbol{\zeta}_i(t)$, the received signals $\{\boldsymbol{\zeta}'_j(t)\}_{j \in N_i^*}$ and the trust values made publicly available by the TMS. For convenience, we assume that the selfish action is observable. The action detection is performed by the k–means clustering algorithm, which classifies the estimates at time t in two clusters $\mathcal{C}_{\mathsf{M}}, \mathcal{C}_{\mathsf{H}}$; containing the agents detected to have taken the malicious and honest action respectively —$\mathcal{C}_{\mathsf{H}}$ is the the cluster that includes agent i.

TATC strategy for parameter estimation. The proposed design employs the *adapt–then–combine* (ATC) strategy [11] for information exchange and parameter estimation. In this case, the node's i estimate of $\boldsymbol{\theta}$ at time t is denoted by $\boldsymbol{\theta}_i(t)$, and is computed in two steps. First, node i adapts $\boldsymbol{\theta}_i(t-1)$, using new local measurements, in order to obtain the intermediate estimate

$$\boldsymbol{\zeta}_i(t) = \boldsymbol{\theta}_i(t-1) + \nu_i \boldsymbol{u}_i(t)\big[d_i(t) - \boldsymbol{u}_i(t)^{\mathsf{T}}\boldsymbol{\theta}_i(t-1)\big] \tag{8}$$

where $\nu_i > 0$ is a suitable step–size parameter, assumed to be sufficiently small for all nodes; without loss of generality, we let $\nu_i = \nu$, for all $i \in N$. Then, node i combines the estimates received from its neighbors to yield a new estimate via

$$\boldsymbol{\theta}_i(t) = \sum_{j \in N_i} \gamma_{ij}(\boldsymbol{s}, \boldsymbol{a}_{-i}) \cdot \boldsymbol{\zeta}'_j(t) \tag{9}$$

where $\gamma_{ij}(\boldsymbol{s}, \boldsymbol{a}_{-i}) \in [0,1]$ is the weight given by node i to its neighbor j, and the received vector $\boldsymbol{\zeta}'_j(t)$ is given by (2). The weights are chosen so that $\sum_{j \in N_i} \gamma_{ij}(\boldsymbol{s}, \boldsymbol{a}_{-i}) = 1$, for all nodes $i \in N$.

Let $s_i^{\mathsf{H}} = \max\{0, 2s_i - 1\}$, for $i \in N$. The above allows node i to realize the combine step given by (9) in a secure way, by utilizing at time t the following weights

$$\gamma_{ij}(\boldsymbol{s}, \boldsymbol{a}_{-i}) = \begin{cases} 1/h_i, & \text{if } j \in \mathcal{C}_{\mathsf{H}} \text{ and } j = i \\ s_j^{\mathsf{H}}/h_i, & \text{if } j \in \mathcal{C}_{\mathsf{H}} \text{ and } j \neq i \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where $h_i = 1 + \sum_{l \in \mathcal{C}_{\mathsf{H}}^*} s_l^{\mathsf{H}}$ and $\mathcal{C}_{\mathsf{H}}^* = \mathcal{C}_{\mathsf{H}} \setminus \{i\}$. Thus, not only do we confine ourselves to nodes having taken a presumably honest action, but also to nodes who have proved to be honest in the past interactions. Clearly, the more noisy the erroneous estimates exchanged by a malicious node become, the higher the probability of correct detection is. In this case, (10) just yields a uniform combination of trusted estimates.

Voting and trust update. The state transitions, meaning the update of the trust values, depend on how agents' actions are perceived by their neighbors and on what evaluations they subsequently submit to the TMS to derive an instantaneous trust score. The agents vote about the trustworthiness of their neighbors on the basis of the action detection outcome. Let $z_{ji} \in B$ be the vote of agent $j \in N_i^*$ for i, at time t. We assume that honest and selfish agents *vote correctly*, in the sense that they report the detected actions of their neighbors and therefore $z_{ji} = (1 + \widehat{a}_{ij})/2$. On the other hand, it is more realistic to let the vote of malicious nodes be a *random variable* [1]. Let $z_{ji} = \frac{1}{2}$ if $\widehat{a}_{ij} = 0$, since the selfish action was assumed to be observable; furthermore we let

$$\Pr\Big(z_{ji} = \tfrac{1+\widehat{a}_{ij}}{2} \,\Big|\, \widehat{a}_{ij} \neq 0\Big) = p \qquad (11)$$

for the malicious agents to vote correctly, for some $p \in [0, 1]$. The votes are aggregated by the TMS to get an instantaneous trust score

$$x_i = \sum_{j \in N_i^*} w_{ij}\, z_{ji}\,, \quad \forall i \in N \qquad (12)$$

where the weights $w_{ij} = s_j \big/ \sum_{l \in N_i^*} s_l$ are used by the TMS. The instantaneous trust score is then used to compute the trust value of agent i at time $t + 1$ as follows

$$s_i' = (1 - \beta)\, x_i + \beta\, s_i \qquad (13)$$

where s_i' is then rounded to the nearest value of B, and the scalar $\beta \in (0, 1)$ is a forgetting factor; smaller values of β give more importance to recent actions. From the above analysis, it is seen that malicious nodes have the maximum impact on the network whenever they are considered to be honest by the TMS. This is also reflected in the reward function $g_{i,t}$, where the illegal gains of agent i are maximized for $s_i = 1$.

Learning algorithm. As the transition probabilities are unknown due to the non–observable actions and the lack of knowledge of the nodes' types, agents should resort to a learning method to evaluate their policy over time. Q–learning is a well known *reinforcement learning* algorithm for the single agent decision making problem, whose extension however into a multi–agent scenario is known to be hard [6]. To overcome this difficulty we assume that each agent is an independent learner in this non–stationary environment and does not make any reasoning on other agents' future actions. Agents select an action at each time with an q–greedy policy and update their Q–function as

$$\begin{aligned} Q_{i,t+1}(s_i, \pi_i) = Q_{i,t}(s_i, \pi_i) + \omega_t \Big(& R_{i,t}(\boldsymbol{s}, \pi_i, \boldsymbol{a}_{-i}, \tau_i) \qquad (14) \\ & + \delta_i \max_{\pi_i'} \big\{ Q_{i,t}(s_i', \pi_i') - Q_{i,t}(s_i, \pi_i) \big\} \Big) \end{aligned}$$

where $\omega_t \in \mathbb{R}^+$ and s_i', π_i' stand for the trust value and policy at time $t + 1$. Due to the non–stationarity of the environment on $f_{i,t}$ and on the other agents' policies (chosen at each time t), we let the agents *explore* with a probability $q \in (0, 1)$ and *exploit* with a probability $1 - q$ the knowledge accumulated through time. Other algorithms that have been proposed in the literature, such as fictitious play and Nash–Q, do not fit in our case, as the strategies of malicious and selfish agents are not stationary (a requirement for fictitious play) and the other nodes' types are unknown. Therefore, an agent is not able to compute the expected Nash equilibrium behavior of its peers (a requirement for Nash–Q).

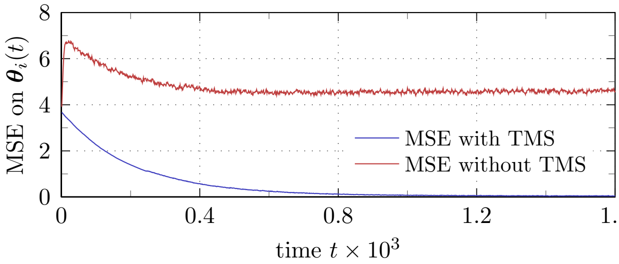

Fig. 1. Expected MSE of an honest agent with and without a trust management system.

4. SIMULATION RESULTS

Throughout the simulations we consider *fully connected* networks, implying that each node can exchange information with any other network node, which is the usual setup in most of the works in distributed parameter estimation and diffusion adaptive networks. The global parameter $\boldsymbol{\theta} = (\boldsymbol{\theta}_{\mathsf{L}}\ \boldsymbol{\theta}_{\mathsf{R}}) \in \mathbb{R}^m$ has length $m = 20$, whereas $\boldsymbol{\theta}_{\mathsf{L}} \in \mathbb{R}^l$ has length $l = 10$. For simplicity, we assume the measurement noise $v_i(t)$ in (1) has the same variance $\sigma_i^2 = 10^{-2}$ for all nodes, and the step–size parameter in the adaptation step (8) equals $\nu_i = 10^{-2}$. The noise generated by malicious agents to corrupt the estimates is Gaussian $\boldsymbol{\eta}_i(t) \sim \mathcal{N}((\mathbf{0}_{\mathsf{L}}\ \mathbf{3}_{\mathsf{R}}), \boldsymbol{J})$. Furthermore, we assume that malicious agents exhibit colluding behavior, and we let $p = 0$ in the simulations (i.e. they always vote for the opposite from the detected action). The other parameters' values are as follows: the forgetting factor $\beta = 0.3$ in voting, the discount factor $\delta_i = 0.95$ and the learning rate $\omega_t = 0.95$. In addition, we define the transmission cost and the illegal gain in (7) as $c = 5 \cdot 10^{-4}$ and $e = 10^{-3}$ respectively. The simulations are executed for a finite time horizon $k = 1.6 \cdot 10^3$, where the exploration probability is $q = 0.4$. In all network settings considered next, 100 Monte Carlo simulations are performed and average results are depicted.

First, assume a network with an honest, a selfish, and a malicious agent, where the noise $\boldsymbol{\eta}_i(t)$ is such that the action detection probability of malicious actions is high. Using the mean square error (MSE) to measure the performance gains in estimation, we compare in Fig. 1 the MSE achieved by an agent when using or not the trust management system. While the MSE converges to zero when using the proposed TMS, this is shown to be impossible without it. Thus, the existence of defensive mechanisms in order to detect, and if possible to prevent, the malicious behavior is necessary in an untrusted environment.

The case of a network with $n = 4$ agents is depicted in Fig. 2, where three agents are honest, and the remaining one

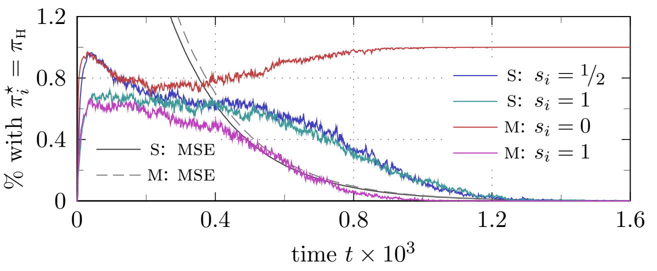

Fig. 2. Fraction of times the optimal policy of a non–honest agent is the honest policy π_H for a given state.

(node i) being either selfish or malicious. Depending on the trust value s_i, we show the fraction of times that the optimal policy $\pi_i^\star = \arg\max_{\pi_i} Q_{i,t}(s_i, \pi_i)$ equals the honest policy π_H. It is seen that the optimal policy for a selfish agent is to be honest most of the time, and specifically at the beginning where the estimation gains are high. However, as the value of $f_{i,t}$ approaches 0, there exists a time instant t_0 beyond which cooperation is not beneficial and the optimal policy is π_S. The findings are similar for the malicious agent, except when it is considered to be untrusted by the TMS. In that case, π_H is the malicious agent's optimal policy (with high probability), as it aims at having a high trust value to harm its peers.

The last simulation deals with a larger network of $n = 70$ nodes with 10 malicious and 20 selfish agents. In Fig. 3a, we show the percentage of selfish and malicious nodes whose optimal policy is π_H regardless their state. The sharp decrease on the percentage of selfish agents was explained above. The optimal policy of most malicious agents is initially π_H and π_M subsequently. Their erratic behavior is due to the fact that they take π_M if $s_i = 1$, and when they do so, this is detected by their neighbors and the voting mechanism. Fig. 3b shows that the state evolution of selfish and malicious nodes decreases, which implies a sound voting scheme.

5. CONCLUSIONS

The problem of sharing information over adaptive networks, in the presence of malicious and selfish agents, was studied in this paper. It was shown that for proper parameter values, the malicious actions can be successfully detected, and the agents acting in a malicious way are then assigned low instantaneous trust value by the TMS. Under the proposed game model, the agents were seen to follow a truth telling strategy for a period that is beneficial for them. Ongoing research seeks to yield necessary and sufficient conditions for truth telling strategies, as well as, to derive Nash equilibria.

6. REFERENCES

[1] J. Baras, "Security and trust for wireless autonomic networks: systems and control methods," *Eur. J. Control*, vol. 13, no. 2–3, pp. 105–133, 2007.

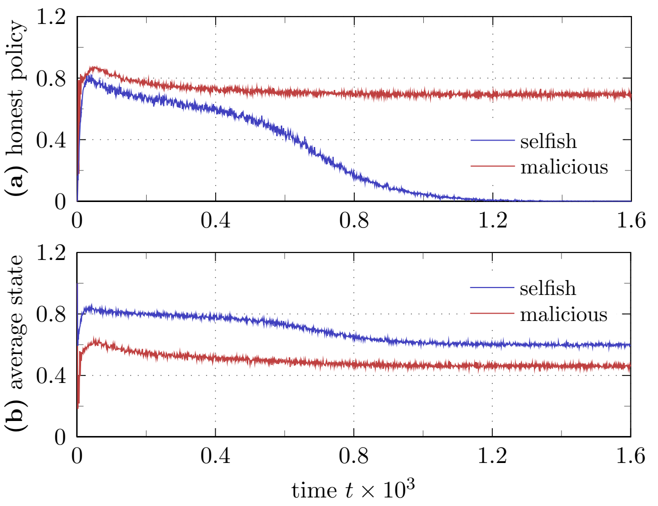

Fig. 3. (a) Percentage of nodes having as optimal the honest policy π_H, and (b) the associated expected state.

[2] Y. Chen and K.J. R. Liu, "Indirect reciprocity game modelling for cooperation stimulation in cognitive networks," *IEEE Trans. Commun.*, vol. 59, no. 1, pp. 159–168, 2011.

[3] X. Jiang, *et al.*, "Game-based trust establishment for mobile ad hoc networks," in proc. *WRI CMC*, pp. 475–479, 2009.

[4] M. Manshaei, *et al.*, "Game theory meets network security and privacy," *ACM Comput. Surveys*, vol. 45, no. 3, article 25, 2013.

[5] Y. Sun, *et al.*, "Information theoretic framework of trust modeling and evaluation for ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 305–317, 2006.

[6] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, MIT press, 1998.

[7] K. Tuyls and G. Weiss, "Multiagent learning: basics, challenges, and prospects," *AI Mag.*, vol. 33, no. 3, pp. 41–52, 2012.

[8] L. Xiao, *et al.*, "Indirect reciprocity security game for large-scale wireless networks," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 4, pp. 1368–1380, 2012.

[9] J. Xu and M. van der Schaar, "Social norm design for information exchange systems with limited observations," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 11, pp. 2126–2135, 2012.

[10] H. Yu, *et al.*, "A survey of multi-agent trust management systems," *IEEE Access*, vol. 1, no. 1, pp. 35–50, 2013.

[11] C.-K. Yu, M. van der Schaar, and A. Sayed, "Reputation design for adaptive networks with selfish agents," in proc. *IEEE SPAWC*, pp. 160–164, 2013.