International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2698-2708
©International Research Publication House. http://www.irphouse.com

An Algorithm for Iterated Game to Handle Self-Obsessed Behavior in a Noisy Social Communication Network

Moirangthem Tiken Singh[1] **and Surajit Borkotokey**[2]

[1]*Department of Computer Science and Engineering, Dibrugarh University Institute of Engineering and Technology, Dibrugarh-786004, Assam, India.*

[2]*Department of Mathematic, Dibrugarh University, Dibrugarh - 786004, Assam, India.*

ORCID: [1]0000-0002-5819-4149, [2]0000-0001-8447-4403

ABSTRACT

The paper proposes a selfish control algorithm which manages the problem of selfishness in a social communication network by adapting and changing zero determinant strategy, a stable variant of two players iterated Prisoner's Dilemma game. The algorithm considers noise in the communication environment which may cause an error at a node while it is interpreting the actions taken by the opponent. We further investigate the relationship between uncertainty level/noise/error with the strategy of the proposed algorithm. The paper also includes comparative performance analysis of controlling selfishness in comparison with a different memory-one strategy like Tit for Tat (TFT), Generous Tit For Tat (GTFT), Generous Zero Determinant (GenZD) using numerical and simulation analysis. The paper further examines the moral judgment of the proposed algorithm considering the matrices like Eigenmosses, Eigenjesus, Good-Partner and Cooperation rating. Here, we show through extensive simulation and analysis that the proposed algorithm control selfishness with the proper moral judgment and performs better than other strategies in an uncertain environment.

Keywords: Social Communication Network; Selfishness; Noise; Memory-one Strategy; Morality Matrices; iterated Prisoner's Dilemma; Zero Determinant Strategy.

INTRODUCTION

Social Communication Network (SCN) [22] like Delay Tolerance Network (DTN) [4] or Opportunistic Mobile Communication Network (OMCN) [7] is a multi-hop mobile communication network. These systems have no path of communication from one end to another end. Transmission of the message in such SCN takes place in one of the two ways: either by delivering the message to the destination mobile node by the source mobile node of the message without involving other mobile nodes, or by delivering the messages to the target node by a mobile node other than the source. If it does the communication through the first case only, then the overall performance of the system get degraded as it delivers the message only when the source node and the destination node are within the transmission contact. So, the good communication network will be a communication system where both cases are used for the transmission of messages.

All the mobile nodes in SCN are free riders. They have limited resources, e.g., power, bandwidth etc. These limitations force the mobile users to drop the messages sent by other nodes to be relayed through them. If every node behaves coherently the same, the efficiency of the SCN will fall drastically [10]. In that case, every node has to deliver messages at the destination on its own. So in the long run, relaying message is the most efficient option for every node i.e., all the nodes need to work cooperatively to form a cooperative social network.

The nodes in the social communication network need to take part in the communication process. However, due to limited resources, mobile nodes may behave selfishly [19] exploiting the selfless (altruism) mobile nodes. Every node in the SCN should address exploitation by selfish nodes by adapting its strategy complying that such self-centered mobile nodes do not use them. So every mobile node must decide whether to forward or drop the message when it receives the message destined for other. We represent the model of such interaction among nodes as two players dilemma game where a mobile node can choose to forward (F) or drop (D) the message relaying through it. Researchers have proposed various ideas to manage selfish behavior. All the proposed works such as [11, 15, 23, 24, 25] are based on the enforcement of cooperation using reciprocity concept.

Majority of the methods collects the behavioral information of the contacting node and use later for deciding whether to help those nodes. For example the mobile node [15] use Generous Tit for Tat (GTFT) [1] for controlling the selfish nodes. The disadvantage in this model emerges when (i) selfish mobile nodes have a static strategy or are impervious in responding to choices of GTFT nodes, (ii) GTFT's nodes end up generously cooperating whereas the traditional approaches like All Defection (alld) or Tit for Tat (TFT) [1] do not cooperate at all and (iii) GTFT's nodes end up cooperating with the selfish node if the GTFT nodes are indifferent to other node's previous decision or when the other's node takes decision randomly. The paper [20] showed that Generous Zero-determinant (GenZD) perform better than GTFT when they interacted with random nodes. As a result, we compare the performance of our strategy with the memory-one strategy like GTFT, TFT, and GenZD. In [11] evolutionary game theory techniques are used to adapt nodes to the changing environment of the social structure. However, the stability of strategies is not discussed in their work. Till now the methods proposed by different researches assume that the strategies interpreted by an node are followed easily by itself. However, due to noise in the communication network, this assumption leads to inefficient results, and complicated the environment. This is because the other nodes do not necessarily know whether a given action is an error or a deliberate choice. Due to this existing models that use strategies like GTFT perform poorly due to the presence of noise.

In this paper,

- An algorithm is defined to control the free riding selfish nodes in a noisy communication environment using Zero-determinant strategy [13].
- A trade-off is proposed between the noise and our algorithm to control the selfishness of the nodes.
- The proposed algorithm is shown to be better than TFT and its variants in punishing deviated nodes.
- Finally, the moral judgment of all the strategies used in this work are compared.

Since we consider that the communication network signals between nodes are not perfect therefore they are subject to interference with other sources. For the algorithm to work, the nodes need to identify the latest strategy followed by other nodes considering these noise parameters. This information, in turn, helps the algorithm to decide which plan to follow during interactions with other nodes i.e., every node following our strategy will be subject to indirect reciprocity. The advantage of our model is that it needs only a history of a single policy pursued by its interacting nodes in the previous round.

1. LITERATURE SURVEY

Many researchers have done extensive work to control selfishness among the nodes in the social communication networks. All the studies consider that due to limited resources, the nodes in the SCN network do not always want to help the other nodes and each node in the network tries to exploit the other nodes' resources. Different cooperation-enforcing schemes have been proposed to promote cooperation between nodes in such systems. The credit-based, reputation-based, and game theory based models are defined to fosters collaboration among the nodes.

The Boltzmann-Gibbs leaning algorithm to select best cooperative coalition structure for taking part in message transmission in [12]. They uses the Shapley solution concept to provide an incentive to the relay node for delivering the messages to its destination. The condition for Nash equilibrium is also defined for the coalition selection strategy. [18] proposed an incentive-based scheme for achieving cooperation among nodes. According to this work, it requires nodes to join a coalitions structure. The selfish nodes are stimulated with reward points to take part in the message forwarding. It determines the amount of incentives for the node using Shapely's solution. However, the proposed scheme does not prescribe how to verify whether a particular node, though, receives the incentive, has delivered the packet to the desired destination or not.

In credit-based approaches, [15, 23] virtual currency or pricing acts increase the interest for participation in the communication. SMART [25] uses a non-game strategy that utilizes a multi-layered credit coin (virtual currency) scheme. However, the use of virtual currency increases the complexity of an already complex network.

[21] proposed a service priority-based incentive scheme using no virtual coins. They allows the system to cope with the undesirable effects of credit-based incentive schemes. Their model uses the reputation incentive metric to stimulate nodes to cooperate. In this model, nodes that relay more message have the higher service priority and get a higher delivery ratio. However, this work assumes that the authority exist for controlling and registering the nodes used for communication. This assumption may not be workable in all Social Communication Networks.

[5] uses reputation and monitoring system together to control selfish node in VANET. They collect reputation when the nodes contact each other and compare with the global reputation value of the network. The monitoring unit upgrade the reputation score of the node whenever a node is communicated. However, calculation of global reputation will be a difficult task for the network as the delay tolerance networks are the temporal networks. The proposed scheme does not specify the characteristics following which the monitoring unit will monitor.

iDetect [3] isolates the low reputation nodes and does not allow them to take part in the communication process. It uses the reputation observed by other nodes to determine the reputation of the nodes. However, this situation leads to the information cascade [6]. It is the condition where the private belief of a node is overwhelmed by the herd behavior.

[2] uses the reputation to detect the cooperation pattern of a node. Every node monitors the behavior of other node and uses the feedback system to report any anomaly. So, it passes the extra token in the network for monitoring and testing the behavior of the nodes. Also, they use a centralized authority system information for the reputation matrix which may sometime unfeasible for the temporal networks.

DISCUSS [11] uses the evolutionary game for controlling the selfish nodes. Every node checks the payoff of its neighbors at a regular interval and adapts its strategy with the aim to increase its payoff. This approach, however, may lead to the worst communication network as everyone try to maximize her own payoff only.

The preliminary conference paper [17] of this work has changed the zero determinant strategy to control selfishness among nodes and has compared with the strategy like GTFT and TFT. However, it did not specify how the node knows the behavior of its contacting nodes. The work neither provide the details of the trade-off between the noise level in the environment, nor the paper has analyzed the ethics of the proposed algorithm. We here in this work extend main article through a rigorous analysis of the relationships between noise and the strategies chosen.

2. MATERIALS AND METHODS

To study the behavior of the nodes in a network, we make the following assumptions:

- Interacting nodes have network traffic and need each other to relay messages.
- Independent of her own action, each node prefers her co-players to be cooperative in forwarding her relayed message.
- Within the group, the dropper (selfish) is strictly advantageous than the forwarder (cooperator).
- Mutual cooperation in forwarding is preferred over mutual defection from forwarding in the social network structure.

Let $N = \{1, 2, 3, 4 \ldots, n\}$ be the set of nodes. To model the social dilemma game, we select only two nodes among the n nodes and allow them to play the two player message forwarding game. In an instance, the game allows each node to send a message and then following their own strategies they decide whether to forward or drop.

Consider the node X, Y, Z, and K. Assume that node X would send a message to Z through Y. Node Y decides whether to forward (F) or drop (D) the message from X based on the reputation of the node X. If node Y forwards the message of X, node X gains a utility of r units while node Y would consume its resource of b units. If node Y transmits the message through X to K, Y will gain a utility of r units while X will lose a utility of b units. Table 1 represents the game so played.

Table 1: Payoff matrix for message transmission between two nodes.

		Node Y: F	Node Y: D
Node X	F	$r-b, r-b$	$-r-b, r$
	D	$r, -r-b$	$-r, -r$

To simplify the representation, we rewrite the payoff matrix after using the following formula $y = \frac{x+r}{2r-b}$ where x is the entry in Table 1, and y is the respective entry in Table 2. We also fix the relay cost $b = 1$.

Table 2: Simplified Payoff matrix after transformation with $b = 1$

		Node Y: F	Node Y: D
Node X	F	$1, 1$	$\frac{-1}{2r-1}\ \frac{2r}{2r-1}$
	D	$\frac{2r}{2r-1}, \frac{-1}{2r-1}$	$0, 0$

2.1. PROBLEM FORMULATION

The communication model between two nodes comprises an enormous amount of message transmission. So, the communication between the two nodes seems straightforward to be modeled as an iterative Prisoner's Dilemma game. However, this is not always possible. The message forwarding game has to work in an uncertain environment where a node may misinterpret the message dropping by its neighbor nodes due to an error in the communication channel as a refusal by its neighbors to forward the messages.

Here, as defined in [17] we denote ψ as the probability of error (noise) while perceiving the strategies of other nodes. Recall that at any stage of the game we have two nodes X and Y. Each node $i \in \{X, Y\}$ takes an action $a_i \in \{F, D\}$. We assume that a node cannot see the actions of other nodes, but has a perception $\varphi = \{\varphi_X, \varphi_Y\} \in \{c, d\} \times \{c, d\}$ when the other node choose an action. Here c and d denote good and bad perception by the node.

We will define $\varphi = \{\varphi_X, \varphi_Y\}$ as the signal profile of two nodes. We also assume that $\varphi_i, \quad i \in \{X, Y\}$ is a stochastic variable where its distribution depends on the action $\{F, D\}$ of nodes and the noise ψ of the environment. Based on the perception profile φ of a node and the action profile a $=\{a_X, a_Y\}$ we define the error probability as δ $(\varphi_X, \varphi_Y|a)$.

According to the error probability, errors occurs if in every stage of the game if node X plays $a_X = F$ (or $a_X = D$) but Y perceived $\varphi_Y = d$ (or $\varphi_Y = c$). Consider nodes X and Y that choose an action F, the error probability of the node will be $\delta(c, c|(F, F))= 1- \psi$, $\delta(c, d|(F, F))= \psi$ for some $\psi \in [0, 1]$. So, based on the above discussion we define the game of each stage as $(a_i, \varphi_i) \in \{F, D\} \times \{c, d\}$. The realized stage payoff $u_i(a_i, \varphi_i)$ for $i \in \{X, Y\}$ for each node based on the action profile and the perception c, d is defined as $u_X(F, c)$ $= u_Y(F, c) = 1$, $u_X(F, d) = u_Y(F, d) = \frac{-1}{2r-1}$, $u_X(D, c) = \frac{2r}{2r-1} = u_Y(D, c)$, $u_X(D, d) = u_Y(D, d) = 0$.

According to the general structure [14], the expected stage payoff for each node in a noisy signal is defined as:

$$f_i(a) = \sum_{\varphi} u_i(a_i, \varphi_i)\delta(\varphi|a) \tag{1}$$

The expected payoff under different action profiles (F, F), (F, D), (D, F), and (D, D) are denoted by R, S, T, P, receptively. The expected stage payoff is then calculated using Eq. (1). For example, the value of R according to the equation is calculated as

$$\begin{aligned} R &= u_i(F, c)\delta(\varphi|F, F) + u_i(F, d)\delta(\varphi|F, F) \\ &= u_i(F, c)\delta(c, c|F, F) + u_i(F, d)\delta(c, d|F, F) \\ &= (1-\psi) + \psi\frac{-1}{2r-1} \\ &= \frac{1}{2r-1}(-\psi + (2r-1)(-\psi + 1)) \end{aligned}$$

We derive the values of S, T, and P as follows.

$$\begin{cases} S = \psi - \frac{-\psi+1}{2r-1} \\ T = \frac{2r(-\psi+1)}{2r-1} \\ P = \frac{2r\psi}{2r-1} \end{cases} \tag{2}$$

The expected stage payoff vector for node X is denoted by $S_X(t) = (R, S, T, P)$ and node Y's payoff is denoted by the vector $S_Y(t) = (R, T, S, P)$. When $\psi \to 0$, S_X and S_Y are ordered as in Prisoner's dilemma.

$$T > R > P > S$$
$$2R > T + S$$

Lets $s_X(t)$ be the payoff received by the node X at time t. The expected payoff of the repeated social dilemma game given the strategy of all other nodes is

$$U(X) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} s_X(t) \tag{3}$$

Every node on the social dilemma game will choose the strategy to maximize U. We use reciprocity for maximizing the utility. A node does this by keeping track of the other node's behavior and reactions. Researchers group the interactions between rational agents into two types, i.e., direct and indirect reciprocity. In the direct reciprocity, a node keeps track of the node she interacts with and therefore does not get enough motivation to relay messages as her judgment is made only by her interaction with that node. Thus, we rule out the use of direct reciprocity. In the indirect reciprocity, a subset of the population observes an action between one node and its opponent. Thus nodes have to be careful in deciding strategies while interacting with others. This concept of indirect reciprocity is based on the philosophy: "I help you, and somebody else helps me" and thus it stimulates the cooperation. A crucial problem in the indirect reciprocity mechanism is to establish the reputation system. So, we establish the reputation system based on the history of the node's actions such that we record 1 and 0 to the forwarder and the dropper, respectively.

2.2. MESSAGE FORWARDING ALGORITHM

We use indirect reciprocity to model the iterative noisy social dilemma game for the reasons as stated above. The proposed social dilemma game is a temporal game. The interacting node changes its partner with time. We model such a game using Markov Process ([17]). We divide the interaction between two nodes into time slots t. For every time interval of t we define the strategy of a node by a set of four parameters p = (p_1, p_2, p_3, p_4). Consider two interacting nodes X and Y at interval time t, then the parameter p of X for forwarding relayed messages is given as follows.

$$p_1 = P(F|(F_X(t-1), F_Y(t-1)))$$
$$p_2 = P(F|(F_X(t-1), D_Y(t-1)))$$
$$p_3 = P(F|(D_X(t-1), F_Y(t-1)))$$
$$p_4 = P(F|(D_X(t-1), D_Y(t-1)))$$

where $F_X(t-1)$, and $D_X(t-1)$ are the forwarding and dropping strategies adopted by node X at time $t-1$; $F_Y(t-1)$ and $D_Y(t-1)$ are the forwarding and dropping strategies of Y perceived by X till time $t-1$.

The function $P(.)$ is the condition of following forward strategy at a time interval t when nodes X and Y choose different strategies at a time interval of $t-1$. The strategy of Y is also defined as a set of four parameters, $q = (q_1, q_2, q_3, q_4)$ where

$$q_1 = P(F|(F_Y(t-1), F_X(t-1)))$$
$$q_2 = P(F|(F_Y(t-1), D_X(t-1)))$$
$$q_3 = P(F|(D_Y(t-1), F_X(t-1)))$$
$$q_4 = P(F|(D_Y(t-1), D_X(t-1)))$$

Following the Markov model, we represent the interaction between the nodes X and Y by a matrix M. We consider the entry of the matrix as a collection of chances of nodes X and Y to forward or drop the messages. For example, the entry in $(F_X(t-1), F_Y(t-1))^{th}$ row and the $(D_X(t), F_Y(t))^{th}$ column is a collection of chances of X and Y to forward or drop the message. The chance of dropping by node X after correctly perceiving Y's earlier strategy and the chance of forwarding by Y after correctly perceiving X's previous move is $(1-3\psi)(1-p_1)q_1$; the chance of dropping by node X after wrongly perceiving Y's strategy and the chance of forwarding after Y correctly perceives X's strategy is $\psi(1-p_2)q_1$. The entry also includes the chance of forwarding by Y when it wrongly perceives X's strategy and the chance of dropping by X by correctly perceiving Y's strategy i.e., $\psi(1-p_1)q_2$; and finally, when both X and Y perceive each other's previous strategies wrongly i.e., $\psi(1-p_2)q_2$. Similarly, for other entries also we calculate the influence of ψ during the interaction of nodes X and Y. Let $\zeta = M - I_4$ whose determinant is 0.

Let $adj(\zeta)$ be the adjoint of ζ such that $adj(\zeta)\zeta = 0_{4,4}$ (by Cramer's rule). Let π be the limiting distribution of M and let w be the limiting row vector, without normalization such that w is a vector satisfying $wM = w$ i.e., $w\zeta = 0$ (since ζ is a singular matrix) and $\pi = \frac{w}{w\cdot 1}$.

Changing the last column of of the singular matrix ζ into player X's stage payoff vector $S_X \in \{R, S, T, P\}$, we can get a new matrix ζ'. Adding the first column into the second and the third column of ζ' gives us a new form of the determinant.

The resultant determinant has a peculiar feature as follows. Its second column $\hat{p} = (-2p_1\psi + p_1 + 2p_2\psi - 1, 2p_1\psi - 2p_2\psi + p_2 - 1, -2p_3\psi + p_3 + 2p_4\psi, 2p_3\psi - 2p_4\psi + p_4)$ depends only on X's strategy p and the noise ψ; Similarly its third column $\hat{q} = (-2\psi q_1 + 2\psi q_2 + q_1 - 1, -2\psi q_3 + 2\psi q_4 + q_3, 2\psi q_1 - 2\psi q_2 + q_2 - 1, 2\psi q_3 - 2\psi q_4 + q_4)$ depends only on Y's strategy and the noise ψ. Since $adj(\zeta)\zeta = 0_{4,4}$ and $w\zeta = 0$, we can prove (using Laplace expansion) that for any four-dimensional vector S_X there exists some $\eta \neq 0$, such that the following equation holds.

$$w.S_X = \eta D(p, q, S_X), \text{ where } D(p, q, S_X) = det(\zeta')$$

$$M = \begin{array}{cc}
 & \begin{array}{cccc} (F_X(t),F_Y(t)) & (F_X(t),D_Y(t)) & (D_X(t),F_Y(t)) & (D_X(t),D_Y(t)) \end{array} \\
\begin{array}{c} (F_X(t-1),F_Y(t-1)) \\ (F_X(t-1),D_Y(t-1)) \\ (D_X(t-1),F_Y(t-1)) \\ (D_X(t-1),D_Y(t-1)) \end{array} &
\begin{pmatrix}
p_1\psi q_2 + p_1 q_1(-3\psi+1) + p_2\psi q_1 + p_2\psi q_2 & p_1\psi(-q_2+1) + p_1(-3\psi+1)(-q_1+1) + p_2\psi(-q_1+1) + p_2\psi(-q_2+1) & \psi q_1(-p_2+1) + \psi q_2(-p_1+1) + \psi q_2(-p_2+1) + q_1(-p_1+1)(-3\psi+1) & \psi(-p_1+1)(-q_2+1) + \psi(-p_2+1)(-q_1+1) + \psi(-p_2+1)(-q_2+1) + (-p_1+1)(-3\psi+1)(-q_1+1) \\
p_1\psi q_3 + p_1\psi q_4 + p_2\psi q_4 + p_2 q_3(-3\psi+1) & p_1\psi(-q_3+1) + p_1\psi(-q_4+1) + p_2\psi(-q_4+1) + p_2(-3\psi+1)(-q_3+1) & \psi q_3(-p_1+1) + \psi q_4(-p_1+1) + \psi q_4(-p_2+1) + q_3(-p_2+1)(-3\psi+1) & \psi(-p_1+1)(-q_3+1) + \psi(-p_1+1)(-q_4+1) + \psi(-p_2+1)(-q_4+1) + (-p_2+1)(-3\psi+1)(-q_3+1) \\
p_3\psi q_1 + p_3 q_2(-3\psi+1) + p_4\psi q_1 + p_4\psi q_2 & p_3\psi(-q_1+1) + p_3(-3\psi+1)(-q_2+1) + p_4\psi(-q_1+1) + p_4\psi(-q_2+1) & \psi q_1(-p_3+1) + \psi q_1(-p_4+1) + \psi q_2(-p_4+1) + q_2(-p_3+1)(-3\psi+1) & \psi(-p_3+1)(-q_1+1) + \psi(-p_4+1)(-q_1+1) + \psi(-p_4+1)(-q_2+1) + (-p_3+1)(-3\psi+1)(-q_2+1) \\
p_3\psi q_3 + p_3\psi q_4 + p_4\psi q_3 + p_4 q_4(-3\psi+1) & p_3\psi(-q_3+1) + p_3\psi(-q_4+1) + p_4\psi(-q_3+1) + p_4(-3\psi+1)(-q_4+1) & \psi q_3(-p_3+1) + \psi q_3(-p_4+1) + \psi q_4(-p_3+1) + q_4(-p_4+1)(-3\psi+1) & \psi(-p_3+1)(-q_3+1) + \psi(-p_3+1)(-q_4+1) + \psi(-p_4+1)(-q_3+1) + (-p_4+1)(-3\psi+1)(-q_4+1)
\end{pmatrix}
\end{array}$$

$$det(\zeta') = \begin{bmatrix} \cdots & -2p_1\psi + p_1 + 2p_2\psi - 1 & -2\psi q_1 + 2q_2\psi + q_1 - 1 & R \\ \cdots & 2p_1\psi - 2p_2\psi + p_2 - 1 & -2\psi q_3 + q_3 + 2q_4\psi & S \\ \cdots & -2p_3\psi + p_3 + 2\psi p_4 & q_2 + 2\psi q_1 - 2\psi q_2 - 1 & T \\ \cdots & 2p_3\psi - 2p_4\psi + p_4 & 2\psi q_3 - 2\psi q_4 + q_4 & P \end{bmatrix}$$

Given π, the long run expected payoff of player X is obtained as follows.

$$s_X = \pi \cdot S_X = \frac{w \cdot S_X}{w \cdot 1} = \frac{D(p,q,S_X)}{D(p,q,1)} \tag{4}$$

Similarly for Y, we can have,

$$s_Y = \pi \cdot S_Y = \frac{w \cdot S_Y}{w \cdot 1} = \frac{D(p,q,S_Y)}{D(p,q,1)} \tag{5}$$

A linear combination of the two scores s_X and s_Y with some coefficients α, β,and γ gives us

$$\alpha s_X + \beta s_Y + \gamma = \frac{D(p,q,\alpha S_X + \beta S_Y + \gamma)}{D(p,q,1)} \tag{6}$$

In Eq. 6, node X (or Y) can unilaterally form a linear relationship between its score s_X (or s_Y) and Y's (or X's) score s_Y (or s_X) i.e.,

$$\alpha s_X + \beta s_Y + \gamma = 0 \tag{7}$$

by setting

$$D(p,q,\alpha s_X + \beta s_Y + \gamma) = 0$$

To do so, Eq. 8 given in the following should have a feasible solution.

$$\hat{p} = \alpha S_x + \beta S_y + \gamma \tag{8}$$

This is because if two columns of ζ' are equal, then its determinant will be equal to zero. If this equation has a workable solution, then it will be possible for node X to adjust $p = (p_1, p_2, p_3, p_4)$ to form a linear relationship between her and her opponent's payoff. Since X's strategy is realized by setting the determinant to zero, it is a generalized framework of Zero-determinant (ZD) strategy [13] in the noisy environment. Now node X can choose the strategy such that it controls the payoff of Y. This is done by setting $\hat{p} = \beta s_Y + \gamma$ such that $\alpha = 0$, and

$$s_Y = -\frac{\gamma}{\beta} \tag{9}$$

Thus we have,

$$\begin{cases} 2p_1\psi + p_1 + 2p_2\psi - 1 = \beta R + \gamma \\ 2p_1\psi - 2p_2\psi + p_2 - 1 = \beta T + \gamma \\ -2p_3\psi + p_3 + 2p_4\psi = \beta S + \gamma \\ 2p_3\psi - 2p_4\psi + p_4 = \beta P + \gamma \end{cases} \tag{10}$$

From Eq. 10, we derive the values of α, γ, p_2, and p_3 such that these variables depend on the p_1 and p_4 of X. Replacing the value of R, S, T and P from Eq. 2 and solving Eq. 10 in term's of p_1 and p_2 we get,

$$\beta = \frac{3}{32\psi^2 - 24\psi + 3}\big(-4p_1\psi + p_1 - 8p_4\psi^2 + 4p_4\psi - p_4 + 4\psi - 1\big)$$

$$\gamma = \frac{1}{128\psi^3 - 128\psi^2 + 36\psi - 3}\big(64p_1\psi^3 - 40p_1\psi^2 + 6p_1\psi - 128p_4\psi^4 + 208p_4\psi^3 - 112p_4\psi^2 + 30p_4\psi - 3p_4 - 64\psi^3 + 40\psi^2 - 6\psi\big)$$

$$p_2 = \frac{1}{128\psi^3 - 128\psi^2 + 36\psi - 3}\big(128p_1\psi^3 - 128p_1\psi^2 + 40p_1\psi - 4p_1 + 8p_4\psi^2 - 4p_4\psi + p_4 - 4\psi + 1\big)$$

$$p_3 = \frac{1}{128\psi^3 - 128\psi^2 + 36\psi - 3}\big(-4p_1\psi + p_1 - 40p_4\psi^2 + 28p_4\psi - 4p_4 + 4\psi - 1\big)$$

So, node X's strategies (i.e. p_1 and p_2) and node Y's expected payoff are related as given by Eq. 9. This states that X component's p_1, p_4 and the noise ψ control Y's payoff.

Following the Zero-determinant strategy (ZD), X can also get a pre-specified extortionate share of the total mutual defection or cooperation payoff. In the long run, depends on the value of K as explained below. This is by setting a unilateral relationship between her payoff and the opponent's payoff. We express such a linear relationship following [13] as:

$$S_X - K = \chi(S_Y - K) \tag{11}$$

where $\chi \geq 1$ is called extortion factor and K is either P or R such that

$$\hat{p} = \phi((S_X - K) - \chi(S_Y - K)$$

When K is P, the extortion strategies ensure that either X receives a higher payoff than Y or otherwise, both nodes receive the payoff for mutual dropping i.e., $s_X = s_Y = P$. Whereas when K is R, strategies ensure that both nodes receive the payoff for mutual forwarding or otherwise X receives a lower payoff than Y. To enforce such relation, node X needs to choose the following strategies.

$$p_1 = \frac{1}{4\psi^2 - 1}\left(2K\chi\phi\psi - K\chi\phi - 2K\phi\psi + K\phi + R\chi\phi - R\phi + 2S\phi\psi - 2T\chi\phi\psi + 2\psi - 1\right)$$

$$p_2 = \frac{1}{4\psi^2 - 1}\left(2K\chi\phi\psi - K\chi\phi - 2K\phi\psi + K\phi - 2R\chi\phi\psi + 2R\phi\psi - S\phi + T\chi\phi + 2\psi - 1\right)$$

$$p_3 = \frac{\phi}{4\psi^2 - 1}\left(2K\chi\psi - K\chi - 2K\psi + K - 2P\chi\psi + 2P\psi + S\chi - T\right)$$

$$p_4 = -\frac{\phi}{4\psi^2 - 1}\left(2K\chi\psi - K\chi - 2K\psi + K + P\chi - P - 2S\chi\psi + 2T\psi\right)$$

where value of ϕ is small and chosen in such a way that the parameters p_1, p_2, p_3, and p_4 have valid values.

Based on the above discussion, we propose a message forwarding algorithm (Algorithm 1).

Algorithm 1: Message Forwarding Algorithms

```
Output: p1, p2, p3, p4
parameter: FORWARD, DROP
while True do
    (p1, p2, p3, p4) ← calculate the probability of
      forwarding the message;
    OStrategy ← get the previous strategy of the other
      mobile node ;
    MyStrategy ← get the previous strategy of the current
      mobile node;
    if OStartegy == FORWARD and MyStrategy ==
      FORWARD then
        return p1;
    else
        if OStrategy == DROP and MyStrategy
          ==FORWARD then
            return p2;
        else
            if OStrategy ==FORWARD and MyStrategy
              ==DROP then
                return p3;
            else
                return p4;
            end
        end
    end
end
```

Table 3: Variables use in Algorithm (1) Pseudo Code

Variable	Description
FORWARD	Forward Strategy
DROP	Dropping Strategy
(p_1, p_2, p_3, p_4)	Stored the probability of forwarding message
OStrategy	Store the previous strategy of the other mobile node
MyStrategy	Store the previous strategy of the current mobile node

3. NUMERICAL ANALYSIS

Observe that Eq. 9 shows that X can pin down Y's outcome. For that X has to choose p_1 and p_4. In the absence of noise, expected payoff s_Y is directly influenced by p_1 and p_4, and so there is a significant value for p_1 and p_4 to pin down Y. However, as the error (noise) increases, the area of p_1 and p_4 to control Y decreases. Simulation has shown that beyond 12%

of noise, X can not pin down Y, i.e., there is no valid value for $p1$ and $p4$. The simulation is done by taking benefit of relay cost as 2, i.e., $r = 2$. From Figure 1, we conclude that when the node plays strategy such that p_1 is set close to 1, the opponent expects payoff to rise to the maximum value. In contrast, if the node sets p_1 close to zero, the expected payoff value of the opponents is pinned down to the minimum value. The above mentioned observation is valid for any condition either in noiseless or noisy environment.

The performance of Algorithm 1 is compared using the two cases in the Axelrod-Python library [9]. In the first case, we compared the performance of our algorithm with GTFT, GenZD, TFT and original Zero-determinant (ZD) strategy. Figure (2) shows that, except for GenZD, and ZD all memory-one strategies performed almost the same. The proposed algorithm, TFT, and GTFT have performed well as compared to other strategies. This is because of the increase in the chance of cooperation of these strategies though the opponent has the wrong impression of dropping messages in the previous round. So, the generosity of our algorithm, GTFT and TFT handle the noise that exists within the environment. In the second case, we added selfish nodes (i.e. alld strategy) along with nodes using memory-one strategies. Figure (3) shows that our algorithm and TFT strategies have outperformed other strategies. Again, another interesting observation is made when we simulate the individual group following TFT and Algorithm 1. We observe that in the presence of noise, TFT's strategy exploits each other. As a result, the group following TFT's strategy have the lowest expected payoff as compared to the group following our algorithm. The measure of the performance is shown by the lower median value of TFT strategy in figure (4). The differences arising between our algorithm and other strategies are due to the consideration of noise in the decision making. Meanwhile the TFT and the other memory-one strategies subsume as if it is in a noiseless environment.

Again, to check the moral judgment of the proposed algorithm, we next considered morality matrices [16]. Accordingly, we check Eigenmoses Rating, Eigenjesus Rating, Good-Partner Rating, and Cooperation Rating matrices of different strategies. In Eigenmoses rating, the strategy requires dropping of messages from nodes who often drop other's messages. It demands justice. In Eigenjessu rating, it maintains that kindness is always better towards those who are themselves kind and thus deserves receiving kindness from other nodes as well. Good-Partner Rating maintains to cooperate as much as the opponent. Lastly, Cooperation Rating defines a clear relationship between the success at the game and higher cooperation rate, i.e., the higher cooperating rate leads to exploitation by the selfish nodes. In the presence of noise, Table 4 shows different matrices of Algorithm 1 and other strategies. As compared to GTFT and GenZD, the proposed algorithm has a higher matrix value of Eigenmoses.

It implies that the proposed algorithm will not compromise with selfish node even in the noisy environment. We also conclude that due to higher cooperation rating, GTFT and GenZD perform worst and is exploited by the selfish nodes.

4. SIMULATION ANALYSIS

We have used the One simulator [8] to test the performance of the algorithm. Altogether 120 mobile nodes were engaged in communication using the epidemic routing algorithm. All the nodes were distributed randomly in a simulation world of 4500×3500 meter^2. 40 out of 120 mobile nodes traveled in the cars inside the simulation world with a speed of 2.5 to 13 km/s. Remaining 80 mobile nodes ware pedestrian. All the 120 nodes had a single short-range single interface system (maximum range of 10 m with the transmission speed of 2Mbps). Besides 120 nodes, there were three routers with two interfaces one having long-range, high-speed signal interface with a range of 1000m (with the transmission speed of 10Mbps) and another having a short range signal interface with the signal interface of 10m (with the transmission speed of 2Mbps). The routers followed the path defined by the map-based information moving at the rate of 7 to 10 km/s. Table 5 gives details of the parameters set for the simulation.

In the simulation, we assigned the node with different strategies, i.e., All Forward (allf), All Drop (alld), Generous Zero-determinant (GenZD), Generous Tit for Tat (GTFT), and Tit for Tat (TFT) strategies. We simulated the different combination of these policies to study the behavior of nodes. We also change the ONE simulator such that nodes learned the policy of the interacting nodes. Node stored the learned policy in a history table. Every time a node interacted with other nodes its history table got updated so it only saved the latest strategy followed. We implemented the proposed Algorithm 1 with $r = 2$. The proposed algorithm used the history table to decide the policy for the future course of action. We also flipped the learned strategy with probability equal to the amount of noise to simulate the noise environment.

To check the exploitation by selfish nodes, we simulated the SCN network taking two groups of mobile nodes, i.e., one selfish group and another following memory-one strategy, i.e., a combination of (a) TFT and Selfish Strategy, (b) GTFT and Selfish Strategy, (c) GenZD and Selfish Strategy and (d) Proposed algorithm and Selfish strategy for 12 hours. For every simulation, we collected traces of messages originating from the groups for the selfish group and the memory-one group. We plotted the summary of the collected statistics in Figure 5. So, as compared to other strategies, our proposed algorithm performs better in controlling free rider selfish nodes.

From the simulation statistic, we observe that if one group followed the TFT's strategy, the number of the message relayed by the network originating from the selfish group declined to the lowest value of 2.815% as compared to 19.34% of the proposed algorithm and 45% of GenZD. The reason for this situation corresponds to the numerical analysis of different strategies. We also observed that the TFT's nodes exploited each other in the same group forcing to decrease the number of messages relayed from the group i.e., reducing to relaying of 3.98% of the total messages generated as compare to 31% of the proposed algorithm. This also corresponds to the numerical analysis of the strategies as predicted by analyzing the moral judgment.

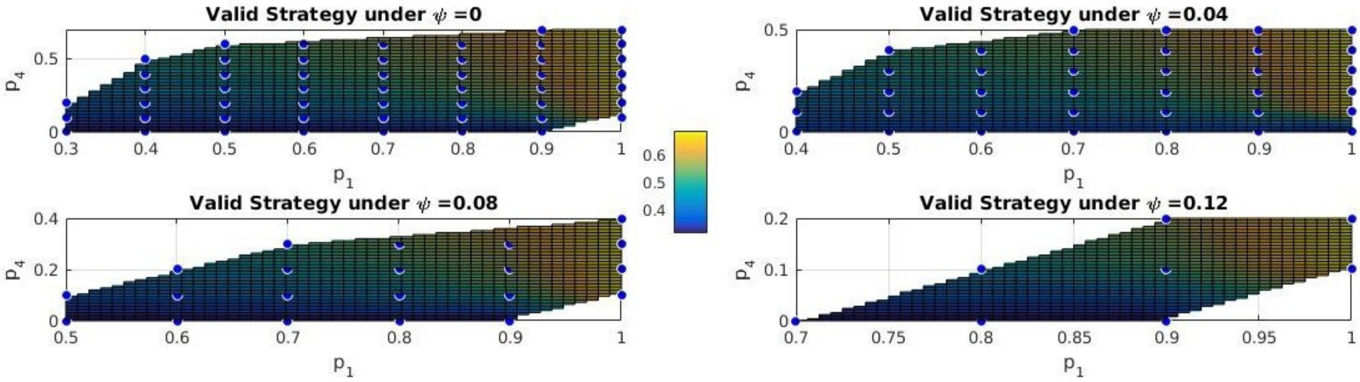

Figure 1: The valid value of p_1 and p_4 that will control Y payoff. The color-bar shows expected payoff value for Y. The study has shown that as noise increases, it reduces the area of X's strategy to control Y payoff. We used $r = 2$ for the simulation

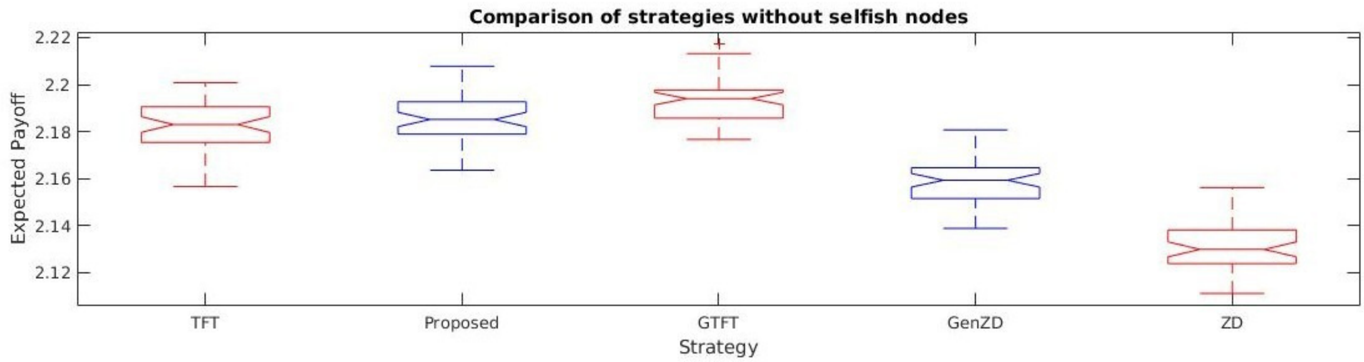

Figure 2: Comparison between the proposed algorithm with GTFT, GenZD, and Zero-determinant (ZD) strategy when the noise is 0.1, i.e.($\psi = 0.1$). The expected payoff is shown when the simulation is iterated for 10,000 times for 50 rounds. We took the value of (R, S, T, P) as $(3, 0, 5, 1)$. The parameters for the proposed algorithm are $\chi = 2$, $\phi = 0.1$

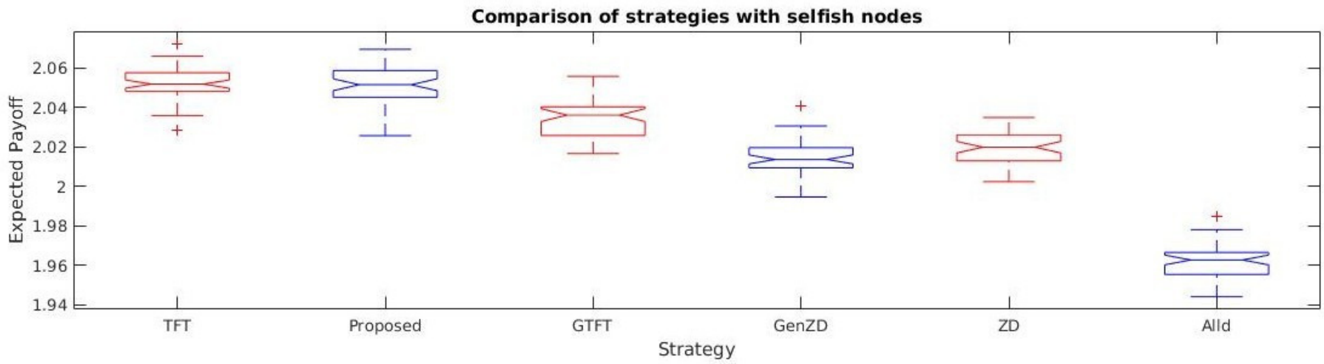

Figure 3: Comparison between the proposed algorithm with GTFT, GenZD, Zero-determinant (ZD) and alld strategy when the noise is 0.1, i.e.($\psi = 0.1$). The expected payoff is shown when the simulation is iterated for 10,000 times for 50 rounds. We took the value of (R, S, T, P) as $(3, 0, 5, 1)$. The parameters for the proposed algorithm are $\chi = 2$, $\phi = 0.1$

Table 4: Moralities value of Iterated Game Strategies with $\psi = 0.1$.

Morality	**Proposed Algorithm**	**GTFT**	**GenZD**	**ZD**	**TFT**
Eigenmoses Rating	0.33	-0.12	0.04	0.44	0.33
Eigenjesus Rating	0.4	0.57	0.49	0.3	0.4
Good-Partner Rating	0.4	1.0	0.8	0.2	0.6
Cooperation Rating	0.45	0.64	0.56	0.35	0.4

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2698-2708

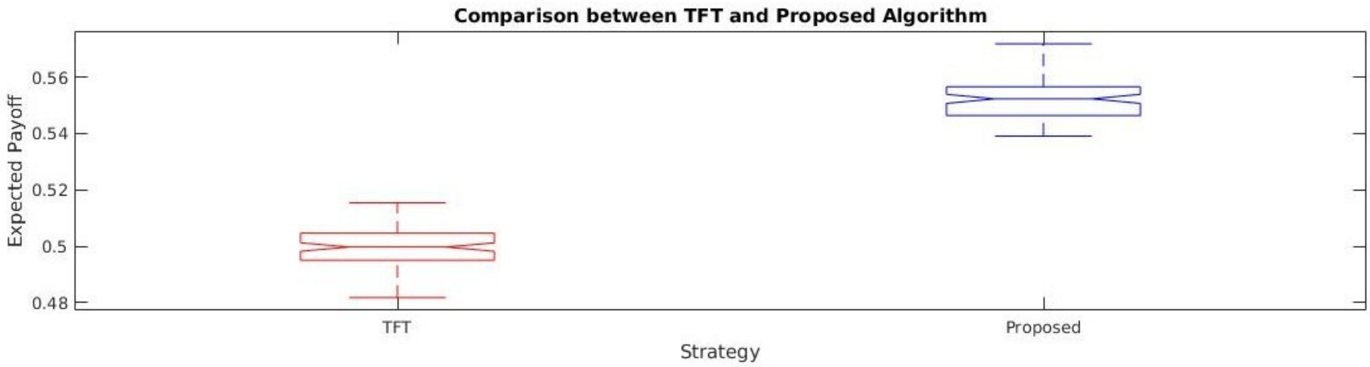

Figure 4: Comparison between TFT and the proposed algorithm. We simulate two cases wherein the first case we have only TFT's nodes, and in the second case, we have only nodes following the proposed algorithm. We used payoff specified in Eq. (2) with $r = 2$, $\chi = 2$, $\psi = 0.1$ and $\phi = 0.2$

Table 5: Different Parameters used for the simulation.

Parameters	**Value**
Buffer Size of Nodes	5 MB
Buffer Size of Router	50 MB
Wait Time of the Group	0 - 120 sec
Message TTL	300 sec
Walking Speed	0.5 -1.5 km/s
χ, ϕ	2 , 0.2

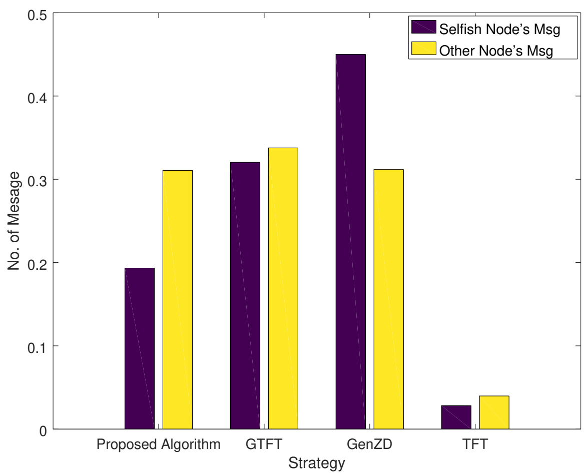

Figure 5: Comparison of rate of messages relayed from different sources.

5. CONCLUSION AND FUTURE WORKS

Every node is a potential router in SCN. Lack of resources makes nodes reluctant to relay other node's messages exploiting other network resources for the benefit of its own. TFT and its variants like GTFT or GenZD are known to control the exploitation by self-centered rational agents. However, uncertainty in the communication channel reduces the performance of TFT, GTFT, and GenZD. To address this uncertainty, the proposed algorithm considers network error (noise) in the decision making. We showed through extensive analysis that a node if wishes would pin down a selfish node. We have shown through rigorous analysis and simulations that our algorithm works well compared to other memory-one strategies. We defined the morality matrices for our algorithm. It is clear from the matrices that our algorithm shows altruistic behaviour to other nodes while it demands justice against the selfish nodes.

In this work, we have prefixed the value of expectation against different strategies. However, if all the nodes have learnt its expected payoff on-line in advance and evolve to meet the changing environment, the whole scenario could have changed. Analysis of such behavior will be the future course of our research.

6. ACKNOWLEDGMENTS

This work is funded partly by UKIERI [184-15/2017(IC)].

7. CONFLICT OF INTEREST STATEMENT

The authors confirmed that the received funding does not lead to any conflicts of interest.

8. DATA AVAILABILITY STATEMENT

The network trace data used to support the findings of this study are available from the corresponding author upon request.

REFERENCES

[1] Axelrod, R. and Dion, D. (1988). The further evolution of cooperation. *Science*, 242(4884):1385–1390.

[2] Chakrabarti, C., Chakrabarti, S., and Banerjee, A. (2015). A dynamic two hops reputation assignment scheme for selfish node detection and avoidance in delay tolerant network. In *2015 IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pages 345–350.

[3] Chakrabarti, C., Roy, S., and Basu, S. (2018). Intention aware misbehavior detection for post-disaster opportunistic communication over peer-to-peer dtn. *Peer-to-Peer Networking and Applications*.

[4] Demmer, M., Brewer, E., Fall, K., Ho, M., Patra, R., Demmer, M., Brewer, E., Fall, K., Jain, S., Ho, M., and Patra, R. (2003). Implementing delay tolerant networking. Technical report.

[5] Dias, J. A. F. F., Rodrigues, J. J. P. C., Kumar, N., and Mavromoustakis, C. X. (2015). A hybrid system to stimulate selfish nodes to cooperate in vehicular delay-tolerant networks. In *2015 IEEE International Conference on Communications (ICC)*, pages 5910–5915.

[6] Gale, D. and Kariv, S. (2003). Bayesian learning in social networks. *Games and Economic Behavior*, 45(2):329 – 346. Special Issue in Honor of Robert W. Rosenthal.

[7] Kadadha, M., Al-Ali, H., Mufti, M. A., Al-Aamri, A., and Mizouni, R. (2016). Opportunistic mobile social networks: Challenges survey and application in smart campus. *2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 1–8.

[8] Keränen, A., Ott, J., and Kärkkäinen, T. (2009). The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA. ICST.

[9] Knight, V., Campbell, O., Harper, M., eric-s s, Janga, V. R., Campbell, J., Langner, K. M., Nikoleta, Rymer, J., Campbell, T., Young, J., MHakem, Palmer, G., Glass, K., Vlašiček, D., edouardArgenson, Mancia, D., Jones, M., gaffney2010, Davidson-Pilon, C., alajara, Das, R., Zoulias, M., Kratz, A., Standen, T., Slavin, P., Pohl, A., Müller, J., Koutsovoulos, G., and Ahmed, A. (2017). Axelrod-python/axelrod: v3.9.0.

[10] Li, Y., Su, G., Wu, D. O., Jin, D., Su, L., and Zeng, L. (2011). The impact of node selfishness on multicasting in delay tolerant networks. *IEEE Transactions on Vehicular Technology*, 60(5):2224–2238.

[11] Misra, S., Pal, S., and Saha, B. K. (2015). Distributed information-based cooperative strategy adaptationin opportunistic mobile networks. *IEEE Transactions on Parallel and Distributed Systems*, 26(3):724–737.

[12] Oualhaj, O. A., Kobbane, A., Elmachkour, M., Sabir, E., and Ben-Othman, J. (2015). A coalitional-game-based incentive mechanism for content caching in heterogeneous delay tolerant networks. In *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 987–992.

[13] Press, W. H. and Dyson, F. J. (2012). Iterated prisoners dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences*, 109(26):10409 – 10413.

[14] Sekiguchi, T. (1997). Efficiency in repeated prisoner's dilemma with private monitoring. *Journal of Economic Theory*, 76(2):345 – 361.

[15] Shevade, U., Song, H. H., Qiu, L., and Zhang, Y. (2008). Incentive-aware routing in dtns. In *2008 IEEE International Conference on Network Protocols*, pages 238–247.

[16] Singer-Clark, T. (2014). Morality metrics on iterated prisoner's dilemma players.

[17] Singh, M. T. and Borkotokey, S. (2018). Selfish controlled scheme in opportunistic mobile communication network. In *Proceedings of the International Conference on Computing and Communications*, Recent Developments in Machine Learning and Data Analytics. Springer Singapore.

[18] Sobin, C. C., Raychoudhury, V., and Saha, S. (2017). An incentive-based scheme for mitigating node selfishness in smart opportunistic mobile networks. *Wireless Personal Communications*, 96(3):3533–3551.

[19] Socievole, A., Rango, F. D., Caputo, A., and Marano, S. (2016). Simulating node selfishness in opportunistic networks. In *2016 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, pages 1–6.

[20] Stewart, A. J. and Plotkin, J. B. (2012). Extortion and cooperation in the prisoners dilemma. *Proceedings of the National Academy of Sciences*, 109(26):10134–10135.

[21] Xie, Y. and Zhang, Y. (2015). A secure, service priority-based incentive scheme for delay tolerant networks. *Security and Communication Networks*, 9(1):5–18.

[22] Xu, B., Yang, D., Liu, Q., and Xiao, Y. (2014). *Social Communication Network, Case Study*, pages 1771–1777. Springer New York, New York, NY.

[23] Yin, L., m. Lu, H., d. Cao, Y., and m. Gao, J. (2010). Cooperation in delay tolerant networks. In *2010 2nd International Conference on Signal Processing Systems*, volume 1, pages V1–202–V1–205.

[24] Zhang, X., Wang, X., Liu, A., Zhang, Q., and Tang, C. (2011). Reputation-based scheme for delay tolerant networks. *Proceedings of 2011 International Conference on Computer Science and Network Technology*.

[25] Zhu, H., Lin, X., Lu, R., Fan, Y., and Shen, X. (2009). Smart: A secure multilayer credit-based incentive scheme for delay-tolerant networks. *IEEE Transactions on Vehicular Technology*, 58(8):4628–4639.