Social Norm Incentives for Network Coding in MANETs
Chuchu Wu, Mario Gerla, Fellow, IEEE, and Mihaela van der Schaar, Fellow, IEEE

Abstract—The performance of mobile ad hoc network transmissions subject to disruption, loss, interference, and jamming can be significantly improved with the use of network coding (NC). However, NC requires extra work for forwarders, including additional bandwidth consumption due to transmitting overheads for redundant NC packets and additional processing due to generating the NC packets. Selfish forwarders may prefer to simply forward packets without coding them to avoid such overhead. This is especially true when network coding must be protected from pollution attacks, which involves additional, often processor intensive, pollution detection procedures. To drive selfish nodes to cooperate and encode the packets, this paper introduces social norm-based incentives. The social norm consists of a social strategy and a reputation system with reward and punishment connected with node behavior. Packet coding and forwarding are modeled and formalized as a repeated NC forwarding game. The conditions for the sustainability (or compliance) of the social norm are identified, and a sustainable social norm that maximizes the social utility is designed via selecting the optimal design parameters, including the social strategy, reputation threshold, reputation update frequency, and the generation size of network coding. For this game, the impacts of packet loss rate and transmission patterns on performance are evaluated, and their impacts on the decision of selecting the optimal social norm are discussed. Finally, practical issues, including distributed reputation dissemination and the existence of altruistic and malicious users, are discussed.

Index Terms—Game theory, network coding, incentive design, reputation scheme.

I. INTRODUCTION

MOBILE devices like smart phones and tablets are becoming increasingly powerful and capable to function not only as clients, but also as peers in a fully fledged ad hoc network. For instance, at a sports arena a spectator may capture a scene on video from a vantage point and peer-to-peer broadcast the video stream to other spectators with an obstructed view. Similarly, a mobile may propagate to neighbors in an ad hoc mode a stream that it is downloading from the Internet via WiFi or 3G. The mobile devices, however, have energy constraints. Since forwarding other devices’ packets provides no benefit to a mobile that is not an intended destination, rather, it consumes battery resources, a self-interested relay node chooses not to forward the packets. If every relay node drops others’ packets, the video never gets delivered to friends several hops away. This selfish behavior, however, can backfire. When the selfish node transmits its own video file, it will be treated the same way, i.e., its file will be dropped. This behavior is known as “tit for tat” in cooperative P2P distribution protocols (e.g. bit torrent) and is actually a primitive type of incentive to induce rational peers to cooperate and serve each other.

A similar situation occurs when the video stream is network coded. Network coding [1]–[3] has been shown to improve streaming performance dramatically in disruptive wireless networks with random loss, jamming or external interference. When the video originator streams the data to one or more receivers in a peer-to-peer ad hoc network, where the links between intermediate nodes can be stressed with high loss rate, network coding can be used to generate redundant packets and offset the loss. Without coding, the linear forwarding scheme fails. It does not retransmit lost packets, due to the broadcast MAC used for multicast streaming. Hence network coding is beneficial to improve packet delivery rate and stream quality.

While network coding has numerous benefits in these deployment scenarios, it also introduces extra costs for the interim forwarders. Comparing with simply forwarding the packets, when network coding is applied with redundancy introduced to offset the wireless loss, the intermediate nodes need to transmit more packets, which results in more bandwidth and power consumptions. Selfish intermediate nodes are motivated to avoid the extra costs and thus, will not perform NC.

Network coding is susceptible to pollution attacks. Upon receiving corrupted packets, the destination cannot decode and must throw away all the encoded packets in the generation. So network coding streams must be protected by special techniques that reveal the presence of an attack and allow to discard faulty packets. Such pollution protection mechanisms introduce even more costs for the intermediate forwarders, which motivates their selfish behavior. Even though there exist light-weight secure schemes, they have other issues, which we will discuss extensively in the related work.

Due to the potentially high cost of secure (i.e. pollution protected) network coding with respect to simple forwarding, selfish intermediate nodes are more likely to refuse to perform coding in order to save power for their own future transmissions. Hence, an incentive scheme is needed to encourage intermediate nodes to perform secure network coding. Incentive schemes have been proposed for conventional mobile networks before, e.g. [25]–[36]. In this paper we extend those...
schemes to the network coding scenario. We must qualify the definition of secure network coding to refer strictly to the pollution detection capability. Accordingly, only the cost of detection is considered in the game setting. There are in fact other costs that add to the basic detection, namely reliable reputation accounting, collusion prevention, malicious attacks to reputation, etc. These issues must be definitely addressed in the implementation and the related costs must be eventually included in the model. However, the inclusion of these factors, when known with precision, will not alter the basic structure and the main conclusions on this study, which rest on the assumption that “secure” network coding is more expensive than simple packet forwarding.

In summary, streaming in lossy ad hoc networks greatly benefits from network coding; network coding must be protected from pollution; pollution protection takes a heavy toll on processing overhead and battery consumption, inviting selfish behavior by intermediate nodes; this in turn defies network coding. Even worse, selfish nodes can drop the packets to save transmission costs, and thus cause more severe damage to network performance. It is thus apparent that to break this impasse and jump start network coding among coding capable nodes or at least forwarding among coding incapable nodes, an incentive mechanism is essential. The main purpose of this paper is to present an efficient incentive scheme for network coding.

To maintain the model analytically tractable, for the first part of this paper we assume a rather simple topology scenario, with unicast from a source to a destination via a single intermediate. The model can be extended to multi-hop and multicast scenarios. With the unicast session, the source node injects a network coded stream. The intermediate node may carry out network coding or may opt to simply forward, or even drop the packets, either because it cannot perform coding, or because it wants to avoid the network coding processing overhead and thus energy expenditure, an important consideration in mobile devices. The social norm based incentive scheme we propose prescribes the intermediate nodes to perform coding and inject redundant packets. If the receiver detects non-compliance by the intermediate node (i.e., the intermediate node refuses to inject redundant coded packets in presence of loss), it will broadcast the deviating action to a scoped neighborhood and the intermediate node’s reputation will be lowered. As a consequence, when a low reputation intermediate node in turn becomes source and sends its own stream, it will be punished; namely, the neighbors will refuse to code its packets according to social strategy. This “punishment” should be sufficient to persuade the nodes that are frequent video originators or receivers to encode the packets, if they can.

The above topology is extremely simple. However, the careful reader will notice that the method and the results of this simple topology can be extended to much more general and complex topologies with multiple intermediate hops, multiple paths and multiple receivers (i.e., broadcast), as discussed later in this paper.

When multiple streams are downloaded from different sources, the possibility of inter-flow network coding arises. Inter-flow network coding can offer additional throughput benefits above intra-flow network coding. It is however an orthogonal problem, for two reasons. First, inter-flow coding does not improve the reliability in conditions of heavy interference and connectivity intermittence (as is the case here). Secondly, inter-flow network coding introduces neither extra processing overhead nor extra pollution risks. So, nodes do not need extra incentives to implement it when it is cost-effective. For these reasons we do not consider inter-flow coding.

II. RELATED WORK

A. Secure Network Coding Schemes for Pollution Protection

Several methods have been proposed for NC pollution avoidance, but they all introduce extra overheads to the intermediate forwarders. The most popular method is the homomorphic hash/signature [4], [5]. This method however requires heavy processing overhead, up to 100 times the processing of conventional network coding [3]. There are other methods that are less processor consuming. For example, the anti pollution schemes DART [40], Null keys [41] and Split Null Keys [42] significantly reduce the NC security overhead with respect to Homomorphic signature/hashing. However, these schemes introduce either latency or processing overhead issues of their own that must be accounted for before adoption. DART requires node synchronization which is difficult to apply to a dynamic MANET where nodes join and leave over time. Also, there is a latency between the reception of a data block and the reception of the checksum that verifies it. For real-time applications that are sensitive to delays, this latency can cause unacceptable quality degradation. Talooki et al. also pointed out this issue in the survey that investigates existing secure network coding schemes [43]. Null Keys is an elegant scheme. However, it requires that for each generation the Null Key matrix is propagated to the network using a homomorphic scheme. Thus, the overhead is reduced with respect to the scheme that Homo-encodes all data packets. However, there is still significant overhead especially if generations are small, say 8 packets, as required for real time streaming. Moreover, there is non-negligible latency required to distribute the Null Key before beginning the transmission of generation packets, as analyzed in [43]. Split Null Key improves upon Null Key in the following way. The Null Key is the sum of two subkeys: a generation independent random subkey that must be delivered to trusted forwarders before the application starts. if it is captured by malicious nodes, the system fails. Then, the generation dependent subkey is broadcast to everyone (no need to secure it). This scheme highly relies on the trusted forwarders at the beginning, which however is not feasible for fully distributed and dynamic mobile ad hoc networks where no one is guaranteed to be trusted. This issue is also pointed out in [43]. In summary, DART introduces latency and thus is not feasible for time sensitive applications such as video streaming of live events such as sports events, concerts and vehicular applications. Null Keys introduces software complexity and overhead to the point that some node may also be motivated to avoid. Split Null Keys also implies the cost of secure distribution of random Split Key, suggesting the use of
a game to incentivize, and also has problem in finding trusted forwarders for the initial distribution of subkeys. Hence, given that the motivation of selfish nodes in avoiding secure NC is still valid, incentives are needed no matter what anti-pollution technique is used.

B. Game Theoretic Research for Network Coding

There are related literatures [7]–[11] using game theory as a tool to solve Network Coding problems, however those works are very different from ours in the following aspects:

- The nodes or players in their models are not truly “selfish”. Even if the word “selfish” is often used to describe the nodes or players in these literatures, nodes are only selfish in the sense that they take the action to maximize its individual utility function, but the utility function is designable by protocol designers. So the nodes are in fact puppets performing the predefined optimized actions. However in our scheme, the utility functions represent the users’ own benefits and costs that cannot be designed by protocol designers. Namely, nodes are truly selfish, and games here are used to model their behaviors, as opposed to an optimization method in these related works.
- The nodes in their model are obliged to perform Network Coding as a common bottom line. However in our scheme, nodes are completely selfish users who are able to decline Network Coding in order to avoid costs.
- Existing game theoretic research on network coding considers only inter-flow network coding but not intra-flow network coding. So the actions in their games are either bandwidth/rate allocation strategies [7]–[9] or route selections [10], [11]. Differently, we consider intra-flow network coding, so the actions in our game are decisions made by intermediate nodes on whether performing intra-flow NC or not, or even dropping the packets.

Essentially, existing game theoretic research on network coding solve optimization problems among altruistic nodes instead of handling selfish behavior.

C. Related Work in Incentive Design

There are two ways to deal with selfish behavior: detect and block the selfish users, or incentivize selfish users to cooperate. Schemes like watchdog and pathrater [13], as well as a reputation scheme [14], have been proposed to detect and isolate selfish users in wireless networks, however if selfish behavior becomes mainstream or pervasive, isolating selfish users cannot be the ultimate solution.

There are related works designing incentives for selfish users in different types of networks, which can be summarized as Table I. The earliest incentive schemes come from economics and are designed for human networks where users are naturally selfish, e.g. online trading systems like eBay [16]. Then incentives are introduced to peer-to-peer file sharing systems where selfish behavior of peers can seriously degrade the system performance [17]–[24]. Up to here, trading or sharing problems are dominantly modeled as simple binary action games, i.e., players have only two possible actions: \{Serve, NotServe\} or \{Share, NotShare\}. Such relatively simple games are rigorously formalized and thoroughly analyzed, various types of incentives (bartering, token, reputation) are explored, and solutions are optimized with the consideration of many practical factors.

Similarly, wireless networks can also be seriously affected by selfish behaviors, e.g. dropping others’ packets to save its own battery and bandwidth, and it can also be modeled as a simple binary action game for conventional wireless networks without Network Coding, i.e., nodes can only choose to either forward or drop the packets. Yet different from the sharing/trading systems where centralized credit/reputation schemes can be feasible, it is very hard to maintain a centralized control scheme in wireless ad hoc networks. A limited amount of work has been done to design incentives for conventional wireless networks [25]–[36]. Among them, only [26] aims at maximizing the overall utility of the network, whereas others each focus on a specific advantage, e.g. [31] focuses on designing a low cost credit system where virtual tokens are easily distributed and maintained; [30] focuses on designing a scheme where secure signatures are ensured to be checked before forwarding; [36] focuses on the robustness and false detection of the reputation system. Some of the above schemes use centralized authority to maintain the reputations/credits, and some propose fully distributed schemes.

Designing incentives for wireless networks with network coding is very different from the ones for conventional forwarding for two major reasons. First, games and incentives for forwarding have a predefined target action, i.e. nodes should forward instead of dropping under any condition. When the NC option is also considered, the utility of NC depends on network conditions (e.g., packet loss rate), so there is no a priory defined target action. In particular, when the network condition is not good but also not extremely bad, the nodes are incentivized to perform NC to improve the delivery rate. When the network condition is either perfect or very bad, NC is not needed or cannot help, respectively. Thus, nodes should not be incentivized to perform NC. In section V, numerical results on how the sustainability varies over the packet loss rate is presented, shown in Fig. 5. Another important difference is that adding more actions, i.e., adding the NC option in addition to forwarding or dropping, further increases the complexity. This leads to additional challenges in the design of the social strategy and reputation update rules. However so far very little work has been done to design incentives for networks with NC. The only schemes we know are a credit scheme [37] and a reputation scheme [38].

---

**TABLE I**

<table>
<thead>
<tr>
<th>Related Works on Incentive Design for Selfish Users</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Literatures</strong></td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>[15]</td>
</tr>
<tr>
<td>[16]</td>
</tr>
<tr>
<td>[17]–[18]</td>
</tr>
<tr>
<td>[19]–[24]</td>
</tr>
<tr>
<td>[25]</td>
</tr>
<tr>
<td>[26]–[36]</td>
</tr>
<tr>
<td>[37]</td>
</tr>
<tr>
<td>[38]</td>
</tr>
</tbody>
</table>

---

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.
Comparing between the two major types of incentive schemes, reputation schemes are advantageous over credit or token schemes in the following aspects: first, the tokens are very difficult to implement. If tokens are simply cryptographic packets, they can be easily duplicated and cannot be used as currency. So the implementation of tokens needs special hardware imbedded techniques. Second, the transactions of token before each transmission introduce a large overhead and serious delay. Due to the above reasons, we deploy reputation schemes instead of tokens.

The existing reputation scheme for network coding in wireless network [38] is far from complete due to the following limitations: first, the existing work does not have rigorous formalism for NC forwarding using games. Second, the incentive compatibility constraints are not considered in the existing work, i.e., the conditions under which the incentive scheme can be complied by selfish users, which is essential and important for any incentive design. Third, the reputation update rule is not optimized. Fourth, the existing work proposed a centralized reputation management scheme with the help of a central authority for wireless mesh network, however such scheme is not feasible for pure MANETs in most cases.

Our incentive scheme is also based on reputation, however, is significantly different from the previous one in these aspects:

- for the first time it models the (secure) network coding and forwarding in wireless networks as an infinite and discounted repeated game and provides rigorous game theory formalism.
- it provides a set of social norm based reputation schemes which are specifically designed for mobile networks with selfish users to encourage network coding.
- it formalizes the social norm selection as an optimization problem with incentive compatible (sustainability) constraints, and solves the problem with both theoretic and numerical analysis.
- it analyzes the impacts of network parameters to our scheme including packet loss rate, the costs of forwarding and coding, and the stability of the network (persistence factor).
- it discusses some practical issues including the implementation details and costs of the distributed reputation scheme, transmission sessions with multiple hops, the existence of malicious and altruistic nodes in the network.

III. System Model

A. Network Model and Network Coding Scenario

We consider a wireless ad hoc network, where mobile devices can act as source, sink or relay nodes at different times or simultaneously in different sessions, as shown in Fig.1.

Intra-flow network coding is used to increase transmission reliability on lossy wireless channels. The source generates a stream that is segmented generation by generation, where generation size = $K$ packets. The source then injects in the network $K$ packets that are linear network coded, i.e., a linear random combination of the initial generation packets. Redundancy is required to overcome losses on link (R,D) with a packet loss rate denoted as $p$. When an intermediate node R receives $K$ encoded packets from its predecessor node (in our case, source node S), it generates and forwards to the next hop (i.e., sink node D) $n$ random linear coded packets[3]. The ratio $n/K$ (where $n \geq K$) is defined as redundancy rate. If a secure NC scheme such as homomorphic signature is used to protect packets from pollutions, the procedure is different from random linear network coding except for the extra processing overhead.

The probability of delivering $K$ packets with $n$ transmissions on a faulty link follows a negative binomial distribution with parameter $1-p$. So the expected number of transmissions $E(n) = K/(1-p)$, which means the redundancy rate should be set to $1/(1-p)$ to achieve full delivery. If less than $K$ linearly independent packets are received, the entire generation is lost because the receiver cannot decode. “Progressive coding” is used here to overcome this problem, where the intermediate node forwards the full generation without coding but only encodes the redundant packets. The advantage of progressive coding is a lower decoding delay and also the ability to receive some packets at the destination even if the full rank is never received. This scheme is clearly attractive if some of the intermediate nodes in the path are not capable to code. In particular, if the non coding node is in the critical section, the destination receives all (and only) the packets in the generation that survive link loss along the path.

To simplify the model, without loss of generality, we assume that the intermediate node knows a priori loss rate $p$ and introduces redundancy accordingly, i.e., it injects $K/(1-p)$ packets, of which $K$ are the original packets and the rest are coded packets.

The nodes in our model are mobiles with the following characteristics: having limited power resources, following selfish algorithms and being randomly matched as session origin, relay and destination. Here selfishness means taking an action to maximize its own utility. The possible action of an intermediate node includes performing NC, simple forwarding or dropping the packets totally. Relatively high mobility forces the devices to have different neighbors throughout time. Hence, we use a global distributed reputation system (say, reputation reports are signed and periodically broadcast to a scoped neighborhood) to keep track of the reputation value of each node. This way, each node has a consistent view of the reputations of the other nodes. More detailed discussion on such distributed reputation maintenance system is presented in Section VII.

The model described above is rather simple (unicast session on a single path), yet it captures the worst damage a node R
A node D; (2) simply forward (F) the original K packets to D;

c in the single path case because of path redundancy. The single

WU et al.

the session. So

are two players: player 1 is the intermediate

forwarding game

= \{(1−p) \}.

K, c_1, c_2, p, K.

in every three sessions, a certain node in the network will act as relay, source and sink respectively. Namely, we assume the nodes in the network are uniform, i.e., having the same chance of matching and sharing the same parameters of δ, B, c_1, c_2, p, K, etc. Expectedly, in every three sessions, a certain node in the network will act as relay, source and sink respectively. The long-term utility of a node is:

\[ u^\infty = \sum_{t=1}^{\infty} \delta^{t-1}(u^t_R + u^t_{SD}) \] (2)

If every node has fixed neighbors and plays the repeated game with the same opponents over time, simply adopting a tit-for-tat strategy will be enough to enforce the nodes to perform network coding. However, mobiles usually change neighbors throughout the experiment, so they do not have knowledge of their new neighbor’s behavior history. Under such circumstance, tagging each node with a reputation value is efficient to record its behavior in the past. By introducing the idea of social norm from economics to wireless networks, we design a novel incentive scheme to encourage nodes to perform network coding despite of the expensive power consumption.

C. Social Norm

The social norm, denoted by κ, is composed of a social strategy σ and a reputation scheme τ.

In our reputation system, every node in the network is tagged with a reputation θ. Reputations are updated periodically, so θ is a constant within each period. We consider the case where reputations take values from a finite set. Without loss of generality, we assume that θ is an integer from the finite set \( \Theta = \{0, 1, \ldots, L\} \) for some L. A high θ means that the node performed well (e.g. coding and/or forwarding) in history, otherwise the node did not perform well in the past.

The reputation system has the following structure: (1) every node maintains a table that stores all its neighbors’ reputation values; (2) nodes broadcast deviating reports to a scoped neighborhood periodically (3) each node updates the reputations of its neighbors according to the deviating reports received within the period.

σ is a reputation-based behavioral strategy, which is represented by a mapping \( \sigma : \Theta \times \Theta^2 \rightarrow A \), where the first \( \Theta \) is the reputation of the intermediate node R, and \( \Theta^2 \) represents the reputations of the S-D pair. A = \{NCF, F, Drop\}. So σ specifies the action \( \sigma(\theta_R, \theta_S, \theta_D) \in A \) which node R with reputation \( \theta_R \) should select when faced with a unicast session consisting of a source node with reputation \( \theta_S \) and a sink node with reputation \( \theta_D \).
Generally, the social strategy rewards high reputation nodes, or highly cooperative nodes, by prescribing intermediate nodes to perform coding and forwarding for them; also, it punishes low reputation nodes, or deviating nodes, by prescribing intermediate nodes to drop their packets. We restrict our attention to a set of threshold-based strategies $\Gamma$. Every strategy $\sigma \in \Gamma$ can be characterized by a threshold $k(\sigma) \in \{0, 1, \ldots, L\}$. Since there are at least three nodes in session $(S,D,R)$, and so three reputations, there are many possible alternative social strategies. The social strategies in (3)(4) are two examples. Among all possible strategies, $\sigma_2$ is the strictest (i.e., node R performs NCF only if both source and sink nodes have reputations higher than $L$). In this paper, we mainly use $\sigma_1$ as our proposed social strategy (here R performs NCF if at least one of the source and sink has higher reputation than $L$).

The reputation update in our scheme lowers the relay node’s reputation only when misbehavior (i.e., performing a worse action than the prescribed action) is detected, whereas performing F/NCF when the prescribed action is “drop” does not count for such misbehavior. So prescribing “drop” when the relay node’s reputation is low simply allows low reputation nodes to automatically recover their reputation without extra efforts. This strategy is only one among many valid strategies; however, various strategies result in different performances under different network conditions. There is no optimal strategy for all network settings and set-ups. We will also compare $\sigma_1$ with other alternatives and analyze the impacts of choosing different strategies in part A of Section IV.

$$\sigma_1(\theta_R, \theta_S, \theta_D) = \begin{cases} 
\text{NCF} & \text{if } \theta_R = L \land \min\{\theta_S, \theta_D\} = L \\
& \land \min\{\theta_S, \theta_D\} \geq k \\
\text{Drop} & \text{if } \min\{\theta_R, \theta_S, \theta_D\} < k \\
\text{F} & \text{Otherwise} 
\end{cases}$$ (3)

$$\sigma_2(\theta_R, \theta_S, \theta_D) = \begin{cases} 
\text{NCF} & \text{if } \theta_R = L \land \min\{\theta_S, \theta_D\} = L \\
\text{Drop} & \text{if } \min\{\theta_R, \theta_S, \theta_D\} < k \\
\text{F} & \text{Otherwise} 
\end{cases}$$ (4)

Consistent with our threshold-based social strategy $\sigma_1$, we propose a “multi-stage punishment reputation scheme” denoted as $\tau$, expressed in formula (5), where $x$ denotes the node’s behavior. Generally, if an intermediate node always complies with strategy $\sigma_1$, represented as $x = 1$, its reputation will increase by 1 until it reaches the maximum $L$; if it deviates from $\sigma_1$ for at least once within the period, its reputation will decrease to either $k$ or 0, depending on its deviating action — if it drops packet totally when it should not (i.e. $x = 2$), its reputation will decrease to 0; otherwise, only decreases to $k$ (i.e. $x = 1$). The “multi-stage punishment reputation scheme” is shown in Fig. 2, where $\alpha_1', \alpha_2'$ are transition probabilities caused by approved behavior and $\varepsilon_{11}', \varepsilon_{12}', \varepsilon_{21}', \varepsilon_{22}'$ are transition probabilities caused by evil behavior.

$$\tau(\theta, x) = \begin{cases} 
\min\{L, \theta + 1\} & x = 0 \\
k & x = 1 \\
0 & x = 2. 
\end{cases}$$ (5)

### IV. GAME OPTIMIZATION

#### A. Metrics

We design social norm-based incentives to encourage nodes in the network to cooperate, aiming at maximizing the transmission performance (packet delivery rate) of the network. We use the total number of packets delivered in all sessions in the network within one time period as the metric of transmission performance. Taking the costs (the power consumptions) into account, we use social utility as a metric of the system. Social utility is defined to be the total utility of all nodes in the network after one time period.

Another important metric for an incentive scheme is the obedience of the nodes, i.e., sustainability of the social norm. A social norm will be sustained, i.e., can persist for ever leading to a stable solution, when selfish nodes choose to comply with the social strategy $\sigma$ instead of deviating from it. In our case sustainability is guaranteed when the proper punishment is enforced, namely when an intermediate node will have a long-term utility decrease if it does not perform coding and/or forwarding when it is prescribed to do so. Sustainability reflects the probability that a selfish node will choose to cooperate and help others. The more intermediate nodes perform coding, the higher delivery rate the network will achieve; so the sustainability of the incentive scheme also affects the transmission performance.

System designers have to consider both aspects, i.e., social utility and sustainability, when selecting parameters of the social norm-based protocol. We wish to maximize both aspects, however we note that very often there is a tradeoff between the two aspects.

#### B. System Convergence

Given the social norm represented by social strategy $\sigma_1$ in (3) and reputation scheme $\tau$ in (5), we derive the stationary distribution of reputation in this subsection. Initially every node is assigned with the maximum reputation value $L$. After interacting with other nodes for some time, the distribution of reputation will converge to a fixed distribution $\eta(\theta)$. 

Fig. 2. Multi-stage punishment reputation scheme $\tau$. 

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.
We assume that each node engages in \( M \) sessions acting as a relay node within one time period. We use \( m_1 \) and \( m_2 \) to denote the number of NC and F sessions for a high reputation intermediate node (an intermediate node with \( \theta_R = L \)). We use \( l \) to denote the number of F sessions for a medium reputation intermediate node (an intermediate node with \( \theta_R \in [k, L) \)). The reputation update probabilities shown in Fig. 2, including \( \alpha_1, \alpha_2, \epsilon_1, \epsilon_2, \epsilon_{21}, \epsilon_{22} \), are all functions of \( \{\text{packet loss rate } p, \text{ packet delivery rate } q(=1-p), \text{ NC generation size } K, m_1, m_2, l\} \). Specific definitions are in Appendix.

**Proposition 1:** When all nodes follow the previously defined social norm, the reputation distribution of nodes in the network converges to a unique point \( \{\eta(\theta)\} \) in (6).

\[
\eta(L)^{-1} = \frac{(1 + \epsilon_{21} + \epsilon_{22} - k \epsilon_{21}')(1 - \alpha_1')(1 - \alpha_2'^{L-k})}{(1 - \alpha_2')\alpha_2'^{L-k} - \epsilon_{11}'} + (k + 1) \left( \frac{1 - \alpha_1'}{\alpha_2'^{L-k} - \epsilon_{11}'} \right) + 1 + \epsilon_{12}'
\]

\[
\eta(\theta) = \frac{1 - \alpha_1'}{\alpha_2'^{L-k} - \epsilon_{11}'} \eta(L), \quad \text{if } k \leq \theta \leq L - 1
\]

\[
\eta(\theta) = \frac{1 - \alpha_1'}{\alpha_2'^{L-k} - \epsilon_{11}'} \eta(L) - \eta(0), \quad \text{if } 1 \leq \theta < k - 1
\]

\[
\eta(0) = \epsilon_{11}' \mu_1 + \epsilon_{21}' \mu_2
\]

\[
\mu_1 = \eta(L)
\]

\[
\mu_2 = \sum_{\theta=k}^{L-1} \eta(\theta) = \frac{(1 - \alpha_1')(1 - \alpha_2'^{L-k})}{(1 - \alpha_2')\alpha_2'^{L-k} - \epsilon_{11}'} \eta(L)
\]

(6)

Where \( \mu_1, \mu_2 \) and \( \mu_3 = 1 - \mu_1 - \mu_2 \) are respectively the proportions of nodes with a high, medium and low reputation at the steady state, as illustrated in Fig. 2. Since we assume that the nodes in a unicast session are randomly matched, then each node has a probability of \( \mu_1 \) to have \( \theta = L \), a probability of \( \mu_2 \) to have \( k \leq \theta < L \) and a probability of \( \mu_3 \) to have \( \theta < k \).

**C. Social Utility at Steady State**

In this subsection we determine the social utility (total utility of all nodes in the network) at steady state for social norm \( \kappa \) with reputation distribution \( \eta(\theta) \) in (6). According to the social strategy \( \sigma_1 \), a session containing low reputation nodes (either S, D or R) always has a utility of 0, because low reputation nodes always drop others’ packets and their own packets are also dropped by others as punishment. So we only consider the sessions where all nodes have reputations \( \theta \geq k \). As defined in part B, \( m_1 \) and \( m_2 \) are the numbers of NC and F sessions with a high reputation intermediate node in one period; \( l \) is the number of F sessions for a medium reputation intermediate node in one period. Since all nodes are considered uniform, the expected utility of any node in the network per one-hop session is:

\[
U = \sum_{\theta} \eta(\theta) v(\theta)
\]

\[
= \frac{1}{3} \mu_1 \left[ \frac{m_1}{M} (B - c_1 + c_2) + \frac{m_2}{M} (Bq - c_2) \right] + \frac{1}{3} \mu_2 \left[ \frac{l}{M} (Bq - c_2) \right]
\]

(7)

Where

\[
m_1 = M \left( \mu_1^2 + 2 \mu_1 \mu_2 \right)
\]

\[
m_2 = M \mu_2^2
\]

\[
l = M (\mu_1 + \mu_2)^2
\]

(8)

We now extend the results to a multi-hop session with \( H \) hops, i.e., \( H \) intermediate nodes, expectedly there are \( \mu_1 H \) high reputation intermediate node which can perform NCF, and \( \mu_2 H \) medium reputation intermediate node which can only do F. In practice, the fraction of nodes with low reputation, i.e., \( \mu_3 \), is very close to 0. In other words, we can take an approximation as \( \mu_1 + \mu_2 = 1 \). So the expected utility of a session with \( H \) hops is approximately:

\[
U^H = \frac{m_1}{M} \left( Bq^{H-1} - c_2 H - \frac{c_1 + c_2}{q} \mu_1 H \right) + \frac{m_2}{M} \left( Bq^H - c_2 H \right).
\]

(9)

**D. Sustainability Conditions**

In this subsection, we determine the sustainability conditions, under which all nodes in the network will rationally choose to comply with the social strategy \( \sigma_1 \) instead of deviating from it. Based on the assumption that all nodes in the network take the action to maximize their own long-term utility, we have the following lemma:

**Lemma 1:** When a social norm \( \kappa = (\sigma, \tau) \) is applied, if the long-term utility, shown in (2), of any node in the network by complying with the social strategy is no smaller than the long-term utility by deviating from the prescribed actions in the social strategy, social norm \( \kappa \) is sustainable, i.e., for any reputation \( \theta \in \Theta \) and any selfish strategy \( \sigma' \neq \sigma \), we must have:

\[
v^\infty(\sigma, \tau) \geq v^\infty(\sigma', \tau)
\]

where,

\[
v^\infty(\theta) = E \left[ \sum_{t=1}^{\infty} \delta^{t-1} v(\theta_t) \right] = v(\theta_{t=1}) + \delta \sum_{\theta'} p_{\tau}(\theta') v^\infty(\theta')
\]

(11)

Initially, all nodes have \( \theta = L \). So we have \( \mu_1 = 1 \) and \( \mu_2 = \mu_3 = 0 \). According to (8), we have \( m_1 = M \) and \( m_2 = l = 0 \). Here \( l = 0 \) is because there is no medium reputation node. In order to make sure all nodes comply with \( \sigma_1 \) initially, all parameters must satisfy the conditions in (12) and (13). If (12) is satisfied, nodes will have a larger (or equal) long-term utility by performing NCF instead of F; if (13) is satisfied, nodes will have a larger (or equal) long-term utility by performing NCF rather than Drop.

\[
\left( B - \frac{c_1 + c_2}{q} \right) \sum_{t=1}^{\infty} \delta^{t-1} \geq B - c_2 + \left( Bq - c_2 \right) \sum_{t=2}^{\infty} \delta^{t-1}
\]

(12)

\[
\left( B - \frac{c_1 + c_2}{q} \right) \sum_{t=1}^{\infty} \delta^{t-1} \geq B - 0 + 0 \cdot \sum_{t=2}^{\infty} \delta^{t-1} = B
\]

(13)
Conclusion 1: If discount factor $\delta$ satisfies the condition in (14), all nodes will perform NC as prescribed in $\sigma_1$, and the social norm will be sustained initially.

$$\delta \geq \frac{c_1 + c_2 - qc_2}{qB(1-q)}$$ (14)

At any time after the initial point until the steady state, nodes will always comply with the social strategy iff. the long-term utility will decrease if they deviate. By comparing the long-term utilities of all possible deviations motivated by selfish intention, i.e., NCF $\rightarrow$ F, NCF $\rightarrow$ Drop and F $\rightarrow$ Drop, we obtain the sustainability conditions in (15)(16)(17).

$$\frac{\delta}{1-\delta} \left[ B - Bq - \left( \frac{\mu_1}{\mu_2} + 2 \right) \frac{(c_1 + c_2) - c_2}{q} \right]$$

$$\cdot \mu_1\mu_2 \geq \frac{c_1 + c_2}{q(1-q)}$$ (15)

$$\frac{\delta}{1-\delta} \left[ B + Bq \left( \frac{\mu_2}{\mu_1} + 2 \right) - \left( 2 + \frac{\mu_1}{\mu_2} \right) \frac{c_1 + c_2}{q} - c_2 \frac{\mu_2}{\mu_1} \right]$$

$$\cdot \mu_1\mu_2 \geq \frac{c_1 + c_2}{q}$$ (16)

$$\frac{\delta}{1-\delta} \left[ \frac{\mu_1}{\mu_2} B + Bq \left( \frac{\mu_2}{\mu_1} + 2 \right) - \left( \frac{\mu_1}{\mu_2} + 2 + \frac{\mu_2}{\mu_1} c_2 \right) \right]$$

$$\cdot \mu_1\mu_2 \geq c_2$$ (17)

For steady state, all parameters must also satisfy the above three sustainability conditions.

$$(14)(15)(16)(17)$$ can also be written as the form of $\delta \geq \delta_1, \delta \geq \delta_2, \delta \geq \delta_3, \delta \geq \delta_4$ or $\delta \geq \max \{ \delta_1, \delta_2, \delta_3, \delta_4 \}$. So, if $\max \{ \delta_1, \delta_2, \delta_3, \delta_4 \} < 1$, the sustainability conditions can always be satisfied if the actual $\delta$ of the nodes in the network is close enough to 1, which indicates that the nodes will stay in the system forever. By computing inequalities (15)(16)(17), we obtain a threshold for $B$ in (18), above which we have $\max \{ \delta_1, \delta_2, \delta_3, \delta_4 \} < 1$.

**Proposition 2:** If benefit $B$ is large enough or at least satisfies:

$$B \geq \frac{c_1 + c_2 - qc_2}{q(1-q)} \left( \frac{\mu_1}{\mu_2} + 2 \right)$$ (18)

there always exists a $\delta \in [0,1]$, s.t. when $\delta \geq \delta$, the incentive scheme can be sustained by all nodes in the network. Here, $\delta = \max \{ \delta_1, \delta_2, \delta_3, \delta_4 \}$ (19)

Since $\delta \in [0,1]$ is decided by the nodes themselves and not tunable, whether the social norm based incentive scheme can be sustained is greatly decided by the system itself. So we use the interval between $\delta$ and 1 as the metric to reflect the sustainability of the scheme.

**Definition 2:** The Sustainable Interval for an incentive scheme is defined as: $SI = 1 - \delta$. In any game theoretic model based on discounted repeated games, SI can be used to represent the sustainability of the incentive scheme. The larger SI, the more compliant and better performing the scheme is.

Nodes with discount factor $\delta$ falling in the interval (i.e., $\delta \geq \delta$) will comply to the social strategy (do network coding/forwarding). A larger SI indicates that more nodes will cooperate and do network coding, so the network will achieve a larger throughput.

**E. Optimization Problem With Constraints**

We formalize the parameter selection as an optimization problem with constraints, i.e., we aim at maximizing the social utility $U$ in (7) and (9) with the constraint of sustainability condition, $\delta \geq \delta$ in (19), for any $(\mu_1, \mu_2)$.

The parameters can be classified into two categories: one category reflects the nature of the network, and is fixed (not tunable by designers); the other reflects the characteristics of the social norm based scheme we propose, and can be tuned by the protocol designer.

In summary, fixed parameters include: packet loss rate $p$, benefit-cost ratio $B : c_1 : c_2$, discount factor $\delta$, number of hops $H$; tunable parameters include: social strategy $\sigma$, reputation threshold $k$, number of sessions $M$ within one reputation update period and NC generation size $K$.

**V. PERFORMANCE EVALUATION - NUMERICAL ANALYSIS**

In this section, we evaluate the performance of the NC forwarding game in the previously introduced scenario using numerical techniques. We compute social utility $U$ and sustainability SI results as a function of system variables and design parameters. In particular we define the optimal selection of reputation thresholds and other design parameters. Here, the evaluation is still under the assumption that nodes are randomly distributed and have equal chance to become source/relay/destination nodes in a session, as described in section III.A.

**A. Selecting the Social Strategy $\sigma$**

In part C of section II, we pointed out that there are many design choices for $\sigma$, and we presented two examples in (3)(4). In this subsection, we show that there exists a tradeoff between social utility and sustainability when designing the social strategy. Specifically, a stricter social strategy (e.g. a strategy that requires high reputation of both source and destination for coding at the intermediate node) will lead to a higher sustainability (i.e., higher compliance ratio) at the price of a lower social utility. We discover this tradeoff by comparing the social utility $U$ and the sustainability interval SI of $\sigma_1$ in (3) and $\sigma_2$ in (4), where $\sigma_2$ is stricter than $\sigma_1$. In part (c) of Appendix, We prove that:

$$U_1 \geq U_2$$ (20)

and

$$SI_1 \leq SI_2$$ (21)

This happens because in $\sigma_2$, the intermediate node performs network coding only when all the nodes in the session have maximum reputation of $L$. Naturally it harms the overall packet delivery rate since it reduces the chance of coding. On the other hand, it enforces the nodes to be more compliant in order to keep a high reputation and avoid severe utility degradation.

**B. Selecting the Design Parameters $L$, $k$, and $M$**

Fig.3 shows the numerical results of stationary social utilities $U$ under different settings of threshold intervals $L - k$ and number of sessions in one reputation update period, $M$. 
In practice, when the social norm is applied, the selfish nodes are incentivized to behave well, so there are very few nodes with reputation lower than \( k \), since the probability that all packets are unexpected lost is tiny. Hence the value of \( k \) almost has no effect on network performance. Here we only consider the setting of the interval between \( L \) and \( k \). According to Fig.3, \( U \) is high when \( L - k \) is set small. On the contrary, when \( L - k \) is set very large, \( U \) approaches the utility value that corresponds to all nodes only performing F. The intuition is that if \( L - k \) is very large, medium reputation nodes have to keep performing well for many periods of time in order to reach \( L \). During this long journey from \( k \) to \( L \), they are still medium reputation nodes which only perform (and are only served with) F. Therefore the network has a lower overall packet delivery rate. We also prove mathematically that \( U \) is a strictly decreasing function of \( L - k \) in Appendix.

Fig.3 also shows that \( U \) increases when \( M \) gets smaller. It means that if the system updates or refreshes reputations frequently, i.e., there are fewer sessions in an update period, the network will have a better packet delivery rate. Intuitively, if refreshing rate is higher, the nodes’ reputations more accurately reflect their behavior history. And they will be rewarded or punished in time. We also prove mathematically that \( U \) is a strictly decreasing function of \( M \) in Appendix.

Fig.4 shows the numerical results of sustainable interval (SI) under different settings of \( L - k \) and \( M \).

According to Fig.4, SI is generally higher when \( L - k \) and \( M \) are set smaller. When \( L - k \) and \( M \) are set very large, SI approaches 0, i.e., nodes will not comply with the social strategy \( \sigma_1 \). The reason is similar to \( U \), i.e., if \( L - k \) and \( M \) are large, a medium reputation node will take too long to become a high reputation node, and it probably will leave the system before its reputation reaches \( L \). Considering its own long-term utility, it naturally has less incentives to perform well.

### C. Impact of Non Configurable System Variables

\( (p, B : c_1 : c_2, \delta) \) on Performance

In the above subsections, we did not discuss the setting of NC generation size \( K \). According to numerical tests, changing \( K \) has little impact on system performance. Intuitively, for both NCF and F, we mainly care about the proportion of packets delivered, so the value of \( K \) does not affect results. This is verified by the simulation result shown in Section VII.B.

In part D of section III, we have thoroughly discussed the impacts of discount factor \( \delta \) (i.e., the probability of a node staying in the system for the next period) on the sustainability of our scheme. The higher \( \delta \), the more sustainable the system is. There exists a threshold of \( \delta \), i.e., \( \delta_0 \), below which nodes will not comply or perform network coding, above which they will. This indicates that a node that will stay in the system for a long time is more likely to help others than the node that will leave the system soon.

Also in part D of section III, we show that there exists a \( B \), below which the scheme can never be sustained. Benefit \( B \) is a relative parameter evaluated by each node itself. If an intermediate node highly values the benefit of its own future transmissions, it has more incentives to serve others and pursue a high reputation. If \( \delta \) is low, a high \( B \) is still insufficient to encourage nodes to do network coding, hence the packet delivery performance does not increase.

A low packet loss rate \( p \), or a high packet delivery rate \( q \), will lead to a higher throughput and social utility, which is obvious. Fig.5 shows how the sustainability of the scheme varies as a function of packet delivery rate. When the packet loss rate is very low, the network condition is good, and nobody does expensive secure NC because the transmission...
result of doing NC is hardly different from doing F. When the packet loss rate is very high, the network condition is very bad, and nobody does expensive secure NC either, because even if they do NC, nobody notices and they will be punished anyway due to the severe packet loss. In realistic scenarios, the packet loss rate is generally well below 0.5. With this condition, we may conclude that our incentive scheme is more sustainable (i.e., more nodes are induced to use Network Coding) when packet loss increases, just as NC is beneficial when packet loss is serious.

VI. PERFORMANCE EVALUATION - SIMULATIONS

We extended our work by performing simulations on a mobile ad hoc network consisting of 30 nodes that are randomly located within 1 km² area and slowly moving towards random directions over time. Each node starts a unicast transmission session at a random time within 60s and lasts for a random period of time. The maximum reputation is set as $L = 10$ and the reputation threshold $k = 6$.

A. Packet Delivery Rate

Fig. 6 shows the simulation results of overall packet delivery rate under different wireless link loss rate. Here we set NC generation size $K = 32$. The red line shows the performance of the scenario where all the relay nodes only perform forwarding - it is intuitive that the overall packet delivery rate is close to $(1 - p)$, where $p$ indicates the link loss rate. The blue line shows the performance of the scenario where the social norm is applied - it significant slows down the performance degradation caused by increased wireless losses.

B. NC Generation Size

To learn the impact of NC generation size to the transmission performance in our scheme, we vary the generation size in the simulation, and examine its impact on the delivery rate. Fig. 7 shows the overall packet delivery rate under different NC generation size. Here link loss rate $p = 0.3$. We see that the generation size does not show a clear positive or negative impact on the delivery rate. As discussed in section III.A, in this paper, we use “progressive coding”, where only the redundant packets are encoded. In this way, we can avoid the total generation loss caused by not receiving enough independent NC packets. Progressive coding should be the major reason why generation size does not significantly affect packet delivery rate or social utility.

VII. PRACTICAL ISSUES

In this section, we consider some practical issues of applying the social norm based protocol to mobile ad hoc networks.

A. Distributed Reputation System

As mentioned before, centralized or partially centralized reputation management with a reputation administrator or multiple reputation managers is not feasible for wireless ad hoc networks. We adopt a fully distributed reputation management system where nodes disseminate reputation information to scoped neighbors of limited hops periodically. Research has been done to minimize the cost of reputation data dissemination by encoding the reputation packets[21], as well as to recognize and prevent fake reputation reports[36]. We try to optimize the reputation system by solving the following three major problems: (1) what’s the simplest message format to broadcast and update reputation information; (2) what’s the best reputation update frequency if taking the cost of reputation dissemination into account; (3) to what scope of neighbors do nodes disseminate the reputation information.

According to the reputation update rule $\tau$ explained in part C of section III, a node’s new reputation is decided by its old reputation and its behavior in the last period. Initially nodes have the uniform reputation information: all nodes are tagged with initial reputation $\theta_0$. If all nodes in the network have consistent reputation information at the beginning of period $i$, as long as they receive consistent information about the behavior of others in period $i$, and calculate the new reputations according to $\tau$ at the end of period $i$, then they will have consistent updated reputation information at the beginning of period $i + 1$. Hence, disseminating reputation values and distinguishing the updated info from out-of-date info is not necessary and should be replaced by disseminating nodes’ behavior instead. So problem (1) is reduced to: in what format do nodes broadcast the behavior of their neighbors. Since reputations are updated only at the end of each period, nodes only need to broadcast a single message, which is
The table shows a summary of its neighbors' behaviors during the period. There are two possible types of deviating actions: perform Drop when it should have performed F or NCF (case 0) and perform F when it should have performed NCF (case 1). The deviating node's reputation should be decreased to 0 for case 0 and decreased to \( k \) for case 1. Let \( x \) denote deviating actions. If a node has \( j \) deviating actions within one period, the worst behavior is considered, i.e., \( x = x_1 \cap x_2 \cap \ldots \cap x_j \). So the deviating action can be represented as a one-bit message: 0 indicating reputation down to 0 and 1 indicating down to \( k \). If Node \( i \) has \( n_i \) deviating neighbors and their Node IDs are \( \{ d_{i0}, d_{i1}, \ldots, d_{in_i-1} \} \), its reputation update message should be in the format shown in Table III. For a network graph \( G(\mathcal{N}, \mathcal{E}) \) with maximum degree \( k(G) \), if it has no more than 256 nodes (\( |\mathcal{N}| \leq 256 \)), then the size of Node ID is 1 byte, and the largest message has size of \( 9 \times k(G) \) bits. Even if \( G \) is one big cluster (i.e., \( k(G) \equiv |\mathcal{N}| - 1 \)), which is not quite possible for ad hoc networks but is the worst-case with the most information redundancy, the message size is approximately \( 1/4 \) of a regular packet size. The reputation information can also be combined as a header of regular video packets when the message size is very small.

If the nodes are static, i.e., every node has fixed neighbors, then it would be sufficient to broadcast reputation update messages within 2 hops. If the network structure and neighbors are changing all the time, broadcasting to farther neighborhood is necessary. Although the scope of reputation broadcasting greatly depends on the mobility of the nodes, however nodes can always broadcast queries and retrieve reputation information from others upon need, and neighbors are relatively fixed within each period due to high reputation update frequency. So, in order to save cost of message dissemination, the reputation message broadcasting can be restricted to 2 hops.

As concluded in part C of section IV, the more frequently reputations are updated, the better system performance can be achieved. However, updating reputation more frequently leads to higher message dissemination costs as well. Let \( c_3 \) denotes the cost of broadcasting one reputation update message to two-hop neighbors. If the ratio between the sizes of a reputation update message and a regular packet is \( f \) (0 < \( f < 1 \)), then \( c_3 \leq (k(G) + 1) \times f \times c_2 \). The cost is estimated to be fairly insignificant comparing to regular packets. Even when \( M = 1 \), i.e., one update for each generation transmission, it would not flood the network. Besides, many routing protocols, such as OLSR, have routing information updates periodically. Hence, the reputation information can be attached to the routing information updates.

### B. Extension to Multi-Hop, Multi-Path and Multicast Scenarios

For multi-hop scenario, as the number of hops \( H \) in a session increases, the chance of packet loss increases, therefore the social utility \( U_H \) and the throughput decrease. In a typical stable MANET, the number of hops needed to reach any node in the network should not be very large, and when the number of hops is small, there is limited degradation of performance. When the number of hops is less than 4, the utility is still better than the case where no network coding is applied. On the other hand, \( H \) does not affect the sustainability of the incentive scheme, nor the parameter selections, because each intermediate node within a session makes forwarding decisions independently, based on the knowledge of the reputation values of itself, the source and the sink node only. Fig. 8 shows that the social utility decreases if the number of hops (or number of intermediate nodes) in the session increases. Here \( U_H \) is an approximate value where we assume every node in the session at least does F and nobody totally drops the packets, because there are a tiny fraction of nodes with reputations lower than \( k \). That is also why the value of \( k \) does not affect system performance. When \( H = 1 \), \( U_1 \sim U \), which in turn testifies our equations in (7) and (9).

In multi-path scenario there exist redundant paths. However, the intermediate nodes along the redundant paths still follow the same social strategy. Hence multi-path scenario is equivalent to multiple unicast sessions over the same pair of source and destination nodes.

In the multicast case, the destinations submit their requests to the source thus forming a multicast tree. The internal tree nodes prune the leaves with low reputation until the reduced tree has only high/medium reputation leaves. The source then multicasts the file over the branches of the tree as shown in Fig. 9. Each downstream node reports the misbehavior of the upstream node according to its action (NCF, F or Drop). The careful reader will notice that some internal tree nodes or leaf node that can overhear will get the NCF delivery even though they have low reputations, thus, they get a “free ride”. A possible solution is for each source to issue periodically a symmetric crypto key to high reputation nodes, and to use that key to distribute the file on the multicast tree. In this case, low reputation nodes that serve as relays are properly penalized as they cannot read the encrypted file. Yet they can still perform network coding and thus improve their reputations. There is clearly a trade off between the benefit of complete enforcement of the rule (i.e., elimination of free rides) and the performance...
C. Networks With Altruistic, Malicious and Intermittent Nodes

So far we assume that all the nodes are selfish and rational, i.e., choose the action that maximizes its own utility. Usually such users are called “reciprocative” users. Now we consider the following cases where some users are:

- altruistic, i.e., always perform NC and forward;
- malicious, i.e., drop packets intentionally and upload bogus packets.

Altruistic users themselves do not need to be incentivized, however the existence of altruistic users will encourage selfish behavior of other selfish nodes — some selfish behavior is not punished. Our social strategy prescribes that a relay should take actions corresponding to the reputation of source and destination nodes. In this sense, the altruistic users deviate from the prescribed strategy — they are supposed to drop the packets from low reputation nodes, but they choose to network code / forward the packets. In order to avoid the negative impacts of altruistic nodes, a modification for the reputation update rule should be enough, i.e., include the improper altruistic behavior as a type of deviating action, and punish the altruistic nodes by reducing their reputation.

Malicious nodes that intentionally drop packets will receive low reputation immediately according to our current reputation scheme. So neighbors will refuse to forward the packets coming from malicious nodes, which effectively stops the spread of bogus/spam information. Hence, our incentive scheme has a by-product of protecting the network from malicious attacks.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we present a social norm based incentive scheme for mobile ad hoc networks, where nodes are selfish, strategic users. For simplicity, we consider a single-path unicast session, where the intermediate node can rationally choose to do secure network coding and forward packets with redundancy (NCF), simply forward the original packets (F) or totally drop the packets (Drop). We model the unicast session as a repeated gift-giving game with persistence factor $\delta$, where nodes are symmetric and randomly matched. Our incentive scheme consists of a social strategy $\sigma$ and a reputation scheme $\tau$, which reward cooperative nodes by increasing their reputation and promising better transmission quality in the future, and punish uncooperative nodes by decreasing their reputations and refusing coding/forwarding their packets in the future. Most existing incentive schemes only propose a single protocol and prove it can incentivize nodes to perform coding/forwarding. Differently from previous schemes, we propose a set of social norm based incentive schemes with parameters that we adjust to optimize the social utility and sustainability. We first prove that the reputation distribution will converge to a unique steady state, corresponding to a fixed setting of the parameters. Then we present a series of results for parameter selection that are supported by mathematical derivations and numerical analysis: (1) there exists a tradeoff between the social utility and sustainability — a stricter social strategy will increase the sustainability but decrease the utility; (2) both utility and sustainability will be optimized if the reputation threshold $k$ is set close to the maximum reputation $L$, and reputations are updated or refreshed frequently; (3) NC generation size does not affect the performance of our incentive scheme, which is also verified by the simulation result. We extend our work with simulations and show that our social norm significantly improves the network’s overall packet delivery rate over the scenario where nodes only perform forwarding. Additionally we show how the network condition or the game itself impacts our incentive scheme. Moreover, practical issues for the proposed reputation system are discussed, including the distributed reputation system, as well as the existence of altruistic and malicious users.

The following summarizes a few limitations of our current scheme and directions for future work.

First, our design did not consider the costs associated with implementing pollution protection mechanisms for network coding such as homomorphic signatures or file reassembly and mixing with new signatures at intermediate nodes [6]. We note that different pollution protection schemes have different benefits, performance costs and limitations. They affect the design social norm design in different ways. Because of these reasons we have opted to present a general social norm framework. This general framework can be easily customized to the various deployment scenarios and their specific characteristics. Future work will consider the impact of specific pollution protection mechanisms and their associated costs and benefits on social norm design and performance.

Second, the social norm reputation scheme is vulnerable to attacks such as faking reputation and collusion, just like any reputation-based system is. Since this paper deals with the problem of selfish nodes that are reluctant to perform network coding because of the costs, and with the design of incentives to encourage them to comply, we focus on the effectiveness of
TABLE IV
NOTATION LIST

<table>
<thead>
<tr>
<th>Notation</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>Relay node</td>
</tr>
<tr>
<td>S</td>
<td>Source node</td>
</tr>
<tr>
<td>D</td>
<td>Destination node</td>
</tr>
<tr>
<td>N</td>
<td>Generation size</td>
</tr>
<tr>
<td>p</td>
<td>Packet loss rate</td>
</tr>
<tr>
<td>B</td>
<td>Benefit for delivering each packet</td>
</tr>
<tr>
<td>c1</td>
<td>Cost of NC encoding each packet</td>
</tr>
<tr>
<td>c2</td>
<td>Cost of forwarding each packet</td>
</tr>
<tr>
<td>δ</td>
<td>Persistence factor</td>
</tr>
<tr>
<td>σ</td>
<td>Social norm</td>
</tr>
<tr>
<td>Σ</td>
<td>Social strategy</td>
</tr>
<tr>
<td>Ξ</td>
<td>Threshold-based strategy set</td>
</tr>
<tr>
<td>α</td>
<td>Observed action</td>
</tr>
<tr>
<td>τ</td>
<td>Reputation update rule</td>
</tr>
<tr>
<td>θ</td>
<td>Reputation of a certain node</td>
</tr>
<tr>
<td>Ω</td>
<td>Reputation set</td>
</tr>
<tr>
<td>L</td>
<td>Maximum reputation value</td>
</tr>
<tr>
<td>k</td>
<td>Reputation threshold</td>
</tr>
<tr>
<td>μ1</td>
<td>Fraction of nodes with high reputation</td>
</tr>
<tr>
<td>μ2</td>
<td>Fraction of nodes with medium reputation</td>
</tr>
<tr>
<td>μ3</td>
<td>Fraction of nodes with low reputation</td>
</tr>
<tr>
<td>α′</td>
<td>Reputation transition probability caused by approved behavior</td>
</tr>
<tr>
<td>ε</td>
<td>Reputation transition probability caused by evildoer behavior</td>
</tr>
<tr>
<td>M</td>
<td>Number of generations in each time period</td>
</tr>
<tr>
<td>η</td>
<td>Reputation distribution</td>
</tr>
<tr>
<td>U</td>
<td>Social utility</td>
</tr>
<tr>
<td>H</td>
<td>Number of hops in a session</td>
</tr>
<tr>
<td>SI</td>
<td>Sustainability interval (1 − q)</td>
</tr>
</tbody>
</table>

APPENDIX A

see notation list in Table IV.

APPENDIX B

PROOF OF PROPOSITION 1

\[
\mu_1 = \frac{\alpha_2^{L-k} \eta(k)}{1 - \alpha_1} + \frac{1 - \delta}{1 - \alpha_1} = \frac{\alpha_2^{L-k} \eta(k)}{1 - \alpha_1} + \frac{1 - \alpha_2^{L-k}}{1 - \alpha_2} = \mu_2
\]

At the steady state, we have \(\eta^t(\theta) = \eta^{t+1}(\theta) = \eta(\theta)\) for any \(\theta \in [0, L]\). By computing the set of equations, we obtain the stationary distribution in (6). Since we assume that the nodes in a unicast session are randomly matched, then each node has a probability of \(\mu_1\) to have \(\theta = L\), a probability of \(\mu_2\) to have \(k \leq \theta \leq L\), and a probability of \(\mu_3\) to have \(\theta < k\). In typical pathologic situations, \(\mu_3\) is very close to zero.

APPENDIX C

PROOF OF EQUATION (20) AND (21)

The social utility \(U\) of any \(\sigma\) is shown in (7). Parameters \(m_1, m_2, l\) corresponding to \(\sigma_1\) are shown in (8). \(m_1', m_2', l'\) corresponding to \(\sigma_2\) are as follows:

\[
m_1' = M\mu_1^2
\]

\[
m_2' = M(2\mu_1\mu_2 + \mu_2^2)
\]

\[
l' = M(\mu_1 + \mu_2)^2
\]

Based on (1) (i.e., \((B - \frac{c_1 + c_2}{q}) \geq (Bq - c_2))\), we have:

\[
U_1 - U_2 = 2\mu_1\mu_2 \left[B - \frac{c_1 + c_2}{q} - (Bq - c_2)\right] \times \frac{1}{3} \mu_1 \geq 0
\]

So we prove that \(U_1 \geq U_2\). According to \(\sigma_2\), the long-term utility in (2) and the sustainability condition in (10), in order to make sure that the nodes will have long-term utility decrease if they deviate, we must have:

\[
\mu_1^2(B - Bq - \frac{c_1 + c_2}{q} + c_2) \geq \frac{c_1 + c_2(1 - q)}{q}
\]

Comparing with the sustainability conditions for \(\sigma_1\) in (15), we have \(\eta_1 \geq \eta_2\) unless \(\mu_1 \ll \mu_2\), which will not happen as long as we choose \(L - k\) and \(M\) properly. So we prove that \(SI_1 \leq SI_2\).

APPENDIX D

PROOF OF MONOTONITY

According to the stationary distribution of reputation \(\{\eta(\theta)\}\) in (6), we have:

\[
\frac{\mu_1}{\mu_2} = \frac{(1 - \alpha_2^L)\alpha_2^L}{(1 - \alpha_1^L)(1 - \alpha_2^L)}
\]

Where \(\alpha_1', \alpha_2' \in [0, 1]\) decrease when \(M\) increases. It is easy to find out that \(\frac{\mu_1}{\mu_2}\) decreases monotonically with \(L - k\) and \(M\). According to the definition of social utility, \(U\) increases monotonically with \(\frac{\mu_1}{\mu_2}\) (a better throughput when a larger fraction of nodes do network coding). So \(U\) decreases monotonically with \(L - k\) and \(M\).

REFERENCES


