Cooperation in Wireless Ad Hoc Networks

Vikram Srinivasan[†], Pavan Nuggehalli[†], Carla F. Chiasserini[‡], Ramesh R. Rao[†]

[†] Department of Electrical and Computer Engineering, University of California at San Diego
email: {vikram,pavan,rao}@cwc.ucsd.edu
[‡] Dipartimento di Elettronica, Politecnico di Torino, Italy
email: chiasserini@polito.it

***Abstract*— In wireless ad hoc networks, nodes communicate with far off destinations using intermediate nodes as relays. Since wireless nodes are energy constrained, it may not be in the best interest of a node to always accept relay requests. On the other hand, if all nodes decide not to expend energy in relaying, then network throughput will drop dramatically. Both these extreme scenarios (complete cooperation and complete non-cooperation) are inimical to the interests of a user. In this paper we address the issue of user cooperation in ad hoc networks. We assume that nodes are rational, i.e., their actions are strictly determined by self interest, and that each node is associated with a minimum lifetime constraint. Given these lifetime constraints and the assumption of rational behavior, we are able to determine the optimal throughput that each node should receive. We define this to be the rational Pareto optimal operating point. We then propose a distributed and scalable acceptance algorithm called Generous TIT-FOR-TAT (GTFT). The acceptance algorithm is used by the nodes to decide whether to accept or reject a relay request. We show that GTFT results in a Nash equilibrium and prove that the system converges to the rational and optimal operating point.**

Methods keywords – Economics (game theory), System Design.

I. Introduction

Wireless ad hoc networks have matured as a viable means to provide ubiquitous untethered communication. In order to enhance network connectivity, a source communicates with far off destinations by using intermediate nodes as relays [1], [2], [3], [4]. However, the limitation of finite energy supply raises concerns about the traditional belief that nodes in ad hoc networks will always relay packets for each other. Consider a user in a campus environment equipped with a laptop. As part of his daily activity, the user may participate in different ad hoc networks in classrooms, the library and coffee shops. He might expect that his battery-powered laptop will last without recharging until the end of the day. When he participates in these different ad hoc networks, he will be expected to relay traffic for other users. If he accepts all relay requests, he might run out of energy prematurely. Therefore, to extend his lifetime, he might decide to reject all relay requests. If every user argues in this fashion, then the throughput that each user receives will drop dramatically. We can see that there is a trade-off between an individual user's lifetime and throughput.

Cooperation among nodes in an ad hoc network has been previously addressed in [5], [6], [7], [8], [9]. In [5], nodes, which agree to relay traffic but do not, are termed as misbehaving. Clever means to identify misbehaving users and avoid routing through these nodes are proposed. Their approach consists of two applications: *Watchdog* and *Pathrater*. The former runs on every node keeping track of how the other nodes behave; the latter uses this information to calculate the route with the highest reliability. In [6], [7], [8], a secure mechanism to stimulate nodes to cooperate and to prevent them from overloading the network is presented. The key idea is that nodes providing a service should be remunerated, while nodes receiving a service should be charged. Based on this concept, an acceptance algorithm is proposed. The acceptance algorithm is used to decide whether to accept or reject a packet relay request. The acceptance algorithm at each node attempts to balance the number of packets it has relayed with the number of its packets that have been relayed by others. The drawback of this scheme is that it involves per packet processing which results in large overheads. In [9], two acceptance algorithms are proposed, which are used by the network nodes to decide whether to relay traffic on a per session basis. The goal of these algorithms is to balance the energy consumed by a node in relaying traffic for others with energy consumed by other nodes in relaying traffic and to find an optimal trade-off between energy consumption and session blocking probability. By taking decisions on a per session basis, the per packet processing overhead of previous schemes is eliminated. We emphasize, however, that all the above algorithms are based on heuristics and lack a formal framework to analyze the optimal trade-off between lifetime and throughput.

In this paper, we consider a finite population of N nodes (e.g., students on a campus). Each node, depending on its type (e.g., laptop, PDA, cell phone), is associated with an average power constraint. This constraint can be derived by dividing its initial energy allocation by its lifetime expectation. We assume that time is slotted and that each session lasts for one slot. We deal with connection-oriented traffic. At the beginning of each slot, a source, destination and several relays are randomly chosen out of the N nodes to form an ad hoc network (e.g., students in a coffee shop). The source requests the relay nodes in the route to forward its traffic to the destination. If any of the relay nodes rejects the request, the traffic connection is blocked.

For each node, we define the *Normalized Acceptance Rate* (NAR) as the ratio of the number of successful relay requests generated by the node, to the number of relay requests made by the node. This quantity is an indication of the throughput experienced by the node. Then, we study the optimal trade-off between the lifetime and NARs of the nodes. In particular, given the energy constraints and the lifetime expectation of the nodes, we identify the feasible set of NARs. This provides us with a set of Pareto optimal values, i.e., values of NAR such that a node cannot improve its NAR without decreasing some other node's NAR. By assuming the nodes to be rational, i.e., that their actions are strictly determined by self interest, we are able to identify a unique set of *rational and Pareto optimal* NARs for each user.

Since users are self-interested and rational, there is no guarantee that they will follow a particular strategy unless they are convinced that they cannot do better by following some other strategy. In game theoretic terms [10], we need to identify a set of strategies which constitute a Nash equilibrium[1]. Ideally, we would like the Nash Equilibrium to result in the rational and Pareto optimal operating point. We achieve this by proposing a distributed and scalable acceptance algorithm, called Generous TIT-FOR-TAT (GTFT). We prove that GTFT is a Nash Equilibrium which converges to the rational and Pareto optimal NARs.

To the best of our knowledge, this is the first paper applying game theory to the problem of cooperation among nodes in an ad hoc network for relaying traffic.

The remainder of the paper is organized as follows. We describe the system scenario and introduce some notations and definitions in Section II. In Section III we use rationality arguments to derive the rational Pareto optimal values of NAR. In Section IV we present the GTFT algorithm that leads the nodes to operate at the rational optimal operating point. Section V shows that the GTFT algorithm constitutes a Nash Equilibrium and that the NARs of the nodes converge to the rational and Pareto Optimal operating point. Numerical results are shown and discussed in Section VI. Section VII discusses some implementation issues of the GTFT algorithm. Finally, Section VIII concludes the paper and points to some aspects that will be the subject of future research.

II. System Model

We consider a finite population of N nodes distributed among K classes. Let n_i be the number of nodes in class i ($i = 1, \ldots K$). All nodes in class i are associated with an energy constraint, denoted by E_i, and an expectation of lifetime, denoted by L_i. Based on these requirements, we contend that nodes in class i have an average power constraint of $\rho_i = E_i/L_i$. We assume that $\rho_1 > \rho_2 > \ldots > \rho_K$. The system operates in discrete time. In each slot, any one of the N nodes can be chosen as a source with equal probability. M is the maximum number of relays that the source can use to reach its destination. The probability that the source requires $l \leq M$ relays is given by $q(l)$. For the sake of simplicity, in our study we assume $q(0) = 0$, i.e., there is at least one relay in each session. This assumption can be easily relaxed by subtracting the energy spent in direct transmissions from the total energy budget of each node. The l relays are chosen with equal probability from the remaining $N-1$ nodes. We assume that each session lasts for one slot. In this time interval, the source along with the l relays forms an ad hoc network that remains unchanged for the duration of the slot.

The source requests the relay nodes to forward its traffic to the destination. A relay node has the option to either accept or refuse the request. We assume that a relay node communicates its decision to the source by transmitting either a positive or a negative acknowledgment. If a negative acknowledgment is sent, the traffic session is blocked. A session is said to belong to type j, if at least one of the nodes involved belongs to class j and the class of any other node is less than or equal to j[2]. As an example, consider a session with two relays. Let the source belong to class 1, the first relay to class 2 and the second to class 1. Then, the session is of type 2. It will become clear later in the paper that the interaction between nodes in a session is dominated by the node with the smallest power constraint.

A node spends energy in transmitting, receiving and processing traffic. We assume that energy spent in transmit mode is the dominant source of energy consumption; thus, in this paper we consider only energy spent in transmitting traffic[3] This allows us to ignore the destination node in our model. The energy consumed by the nodes in transmitting a session will depend on several factors like the channel conditions, the file size, and the modulation scheme. Here, we assume that the energy required to relay a session is constant and equal to 1. While this is not a very reasonable assumption, it allows us to capture the salient aspects of the problem. We believe that the ideas presented in this paper can be extended to more realistic settings.

Finally, for a generic node h, we denote by $B_h^j(k)$ the number of relay requests made by node h for a session of type j till time k, and by $A_h^j(k)$ the number of relay requests generated by node h for a session of type j which have been accepted till time k. Equivalently, we denote by $D_h^j(k)$ the number of relay requests made to node h for a session of type j till time k, and by $C_h^j(k)$ the number of relay requests made to node h for a session of type j which have been accepted by node h till time k.

For $1 \leq j \leq K$ and $1 \leq h \leq N$, we define: $\phi_h^j(k) = A_h^j(k)/B_h^j(k)$, and $\psi_h^j(k) = C_h^j(k)/D_h^j(k)$. Observe that ϕ_h^j is the ratio of the number of relay requests for type j sessions made by h which have been accepted, to the number of requests for type j sessions made by h; thus, ϕ_h^j is an indication of the throughput experienced by h, with respect to type j sessions. The Normalized Acceptance Rate (NAR)

[1]A Nash equilibrium is a strategy profile having the property that no player can benefit from unilaterally deviating from his strategy.

[2]The nodes involved in the session include the source and the relays; the destination node is not considered.

[3]We ignore the energy spent by a source in requesting nodes to relay traffic and the energy spent by a relay in communicating its decision

is defined as NAR $= lim_{k\to\infty}\phi_h^j(k)^4$. Note that the NAR is defined for each node and session type, however, we have suppressed the indices for the sake of simplicity. From the above definitions it is clear that the throughput of a node is determined by its values of NAR. In the following we will equivalently refer to NARs and throughput.

III. Rational and Pareto Optimal Operating Point

The set of NAR values which users receive is a function of the acceptance algorithm executed at the relays. As mentioned above, we assume that the nodes are rational, i.e., their actions are strictly determined by self interest. Given this assumption, we can identify a set of NAR values such that: *(i)* they meet the energy constraints of the nodes; *(ii)* they are Pareto optimal values, i.e., values of NAR such that a node cannot improve its NAR without decreasing some other node's NAR; *(iii)* all rational users will find the allocation fair to themselves and hence will accept it.

In order to derive the feasible region of operation, we assume that the nodes adopt a stationary policy, i.e., a node in class i in a session of type j accepts a relay request with probability τ_{ij}. Given this stationary policy, we first write the constraints on the energy consumption rate of the nodes, from which we can derive the feasible set of τ_{ij}s. Consider a node p participating in a type j session ($1 \le j \le K$). The average energy per slot spent by the node as a source, $e_{pj}^{(s)}$, can be written as

$$\begin{aligned} e_{pj}^{(s)} &= \frac{1}{N} \times \text{NAR} \\ &= \frac{1}{N}\sum_{l=1}^{M}\sum_{h_1,\dots h_j} q(l)\Gamma(l;h_1,\dots h_j)\tau_{1j}^{h_1}\dots\tau_{jj}^{h_j} \end{aligned} \tag{1}$$

where:

- $1/N$ is the probability that node p is the source;
- $\Gamma(l;h_1,\dots h_K)$ is a multivariate probability function conditioned on the fact that the session belongs to type j with l relays. h_i refers to the number of relays of class i participating in the session;
- $\tau_{1j}^{h_1}\dots\tau_{jj}^{h_j}$ represent the probability that all the relay nodes accept the request.

Similarly, the average energy per slot spent by the node as a relay, $e_{pj}^{(r)}$, is given by

$$\begin{aligned} e_{pj}^{(r)} &= \frac{1}{N}\sum_{l=1}^{M} lq(l)\sum_{h_1,\dots h_j}\Gamma(l-1;h_1,\dots h_j) \\ &\qquad \tau_{1j}^{h_1}\dots\tau_{jj}^{h_j}\tau_{class(p)j} \end{aligned} \tag{2}$$

with l/N being the probability that node p is chosen as one of the l relays. The feasible region for the τ_{ij}s is then defined by the following set of inequalities,

$$\begin{aligned} \sum_{j=1}^{K}(e_{pj}^{(s)} + e_{pj}^{(r)}) \le \rho_{class(p)} \quad 1 \le p \le N \\ \tau_{class(p)j} \in [0,1] \quad 1 \le j \le K; 1 \le p \le N, \end{aligned} \tag{3}$$

where $class(p)$ is the class to which node p belongs. For a feasible set of τ_{ij}s, the corresponding feasible set of NARs can be directly computed from (1). The Pareto optimal values of the τ_{ij}s can be derived by imposing the equality relation in (3).

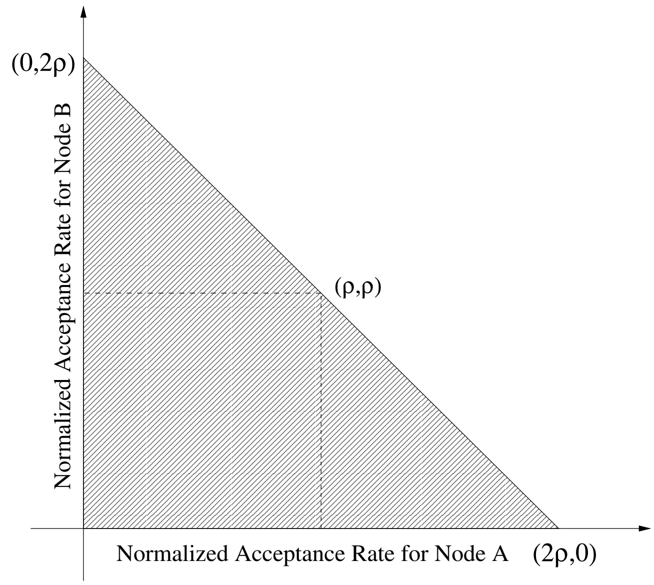

Fig. 1. Feasible Region for $N = 2$, $K = 1$, $\rho = 0.5$.

As an example, consider a system with two nodes, say A and B, belonging to the same class and with a power constraint ρ. Assume that both nodes want to transmit to an Internet access point, and $q(1) = 1, M = 1$. In this case, the feasible region for the NARs is shown in Fig. 1. The Pareto optimal values of the NARs are given by the line segment joining $(0, 2\rho)$ with $(2\rho, 0)$. In fact, while operating at any of these points, both nodes are consuming energy at the maximum allowable rate. Therefore a node cannot increase its NAR without decreasing the other node's NAR.

We now show how rationality can be used to derive the unique operating point from the set of feasible points. Rationality implies that each user wants to maximize his benefit by expending least amount of effort (i.e., energy). In the example in Fig. 1, it is straightforward to see that the only Pareto optimal operating point acceptable to both rational users is (ρ, ρ). In the case of multiple classes, nodes belonging to different classes will have different NARs. The notion of rationality can be extended to this case as follows. First, consider a system with N nodes, all belonging to the same class. By rationality, each node must possess the same value of NAR; thus, it is a simple matter to derive the maximal value of τ which satisfies the energy constraint as in (3). Then,

[4] We don't define this as an acceptance probability, since we don't restrict attention to the class of stationary acceptance algorithms

consider a system with n_1 nodes in class 1 and n_2 nodes in class 2. Suppose $n_1 = 1$; by rationality, the lone node in class 1 will not expend more energy than the remaining nodes in class 2. This is because the node in class 1 will not receive higher throughput if it is more generous to users in class 2 than users in class 2 are to it. Indeed, self interest dictates that the lone node behaves as though he belongs to class 2. Suppose now, that there are two nodes in class 1. When the nodes in class 1 are involved in type 2 sessions, they have no incentive to behave any differently than as if they were class 2 nodes. While, when they are involved in type 1 sessions, they can utilize their excess energy to their mutual benefit. Thus the rationality argument leads us to the following lemma.

Lemma 1: For a set of self-interested nodes, the rational values of τ_{ij} have the following property.

$$\tau_{ij} = \tau_{jj} \quad 1 \le i \le j \le K. \tag{4}$$

Henceforth, we shall denote τ_{jj} by τ_j.

Given Lemma 1, the rational Pareto optimal values of the τ_js and hence the NARs can be determined by recursively solving the energy constraints in (3) and by using (1) and (2). Also, for a node h belonging to class i involved in type j sessions, we define

$$L_{ij} = \frac{\text{Prob}(h \text{ is served in a type } j \text{ session})}{\text{Prob}(h \text{ accepts to relay a type } j \text{ session})}. \tag{5}$$

L_{ij} is the ratio of the rational Pareto optimal NAR for type j session to τ_j. Below some examples are provided.

A. Example 1

Consider K classes and N nodes with n_i nodes in class i, and $q(1) = 1, M = 1$, i.e., the route between any source-destination pair consists of exactly one relay node. In this case, the session type is the maximum of the source class and the relay class. Consider a node in class i. The average energy per slot spent by the node as a source is as follows

$$e_i^{(s)} = \frac{1}{N(N-1)} \left[\sum_{k=1}^{i-1} n_k \tau_i + (n_i - 1)\tau_i + \sum_{l=i+1}^{K} n_l \tau_l \right].$$

When the relay belongs to a class lower than i, the session is of type i and if the relay belongs to a class higher than i, the session type is the same as the class of the relay. The same expression holds for the average energy per slot, $e_i^{(r)}$, spent by the node as a relay. The rational Pareto optimal τ_i can be derived from the set of equations below

$$\begin{aligned} e_i^{(s)} + e_i^{(r)} &= \rho_i \quad 1 \le i \le K \\ \tau_i &\in [0,1] \quad 1 \le i \le K. \end{aligned}$$

In particular, for $K = 1$, the rational and Pareto optimal τ is equal to $N\rho/2$, and the rational Pareto optimal NAR is equal to τ.

B. Example 2

Consider a system with two classes. For simplicity assume that no more than 2 relays are ever involved ($M = 2$). Consider a node in class 2. The energy spent by this node as a source, $e_2^{(s)}$, and as a relay, $e_2^{(r)}$, are given by

$$\begin{aligned} e_2^{(s)} &= \frac{1}{N} \sum_{l=1}^{M} q(l) \tau_2^l \\ e_2^{(r)} &= \frac{1}{N} \sum_{l=1}^{M} l q(l) \tau_2^l. \end{aligned}$$

The optimal τ_2 can be found by solving the quadratic equation $e_2^{(s)} + e_2^{(r)} = \rho_2$.

Now consider a node in class 1. The energy spent by this node as a source, $e_1^{(s)}$, and as a relay, $e_1^{(r)}$, are given by

$$\begin{aligned} e_1^{(s)} &= \frac{1}{N} \Bigg[q(1) \left\{ \frac{n_2}{N-1}\tau_2 + \left(1 - \frac{n_2}{N-1}\right)\tau_1 \right\} \\ &+ \quad q(2) \left\{ \frac{(n_1-1)(n_1-2)}{(N-1)(N-2)} \tau_1^2 \right. \\ &+ \quad \left. \left(1 - \frac{(n_1-1)(n_1-2)}{(N-1)(N-2)}\right) \tau_2^2 \right\} \Bigg] \\ e_1^{(r)} &= \frac{1}{N} \Bigg[q(1) \left\{ \frac{n_2}{N-1}\tau_2 + \left(1 - \frac{n_2}{N-1}\right)\tau_1 \right\} \\ &+ \quad 2q(2) \left\{ \frac{(n_1-1)(n_1-2)}{(N-1)(N-2)} \tau_1^2 \right. \\ &+ \quad \left. \left(1 - \frac{(n_1-1)(n_1-2)}{(N-1)(N-2)}\right) \tau_2^2 \right\} \Bigg]. \end{aligned} \tag{6}$$

Since we know τ_2, we can obtain τ_1 by solving the quadratic equation $e_1^{(s)} + e_1^{(r)} = \rho_1$.

Note that the method presented in these examples can be easily extended to multiple classes and relays.

IV. The GTFT Algorithm

In this section, we present a distributed and scalable acceptance algorithm which propels the nodes to operate at the rational Pareto optimal NARs. We call this algorithm the Generous TIT-FOR-TAT (GTFT) algorithm.

In a network of self-interested nodes, each node will decide on those actions which will provide it maximum benefit. Any strategy that leads such users to the rational optimal NARs should possess certain features. Firstly, it cannot be a randomized stationary policy. If a node in class i gets a request for a type j session, then a possible course of action would be to accept that request with probability τ_j. If all nodes were to use this policy, then the rational optimal τs described in Section III can be used to achieve the optimal operating point. However, a rational selfish node will exploit the naivete of other nodes by always denying their relay requests thereby increasing its lifetime, while keeping its NAR constant. In other words, in our system, any stationary strategy is dominated by the always deny behavior. Hence, stationary strategies are not sustainable, and behavioral strategies are required in order to stimulate cooperation. By behavioral

TABLE I

PUNISHMENT MATRIX FOR THE PRISONERS DILEMMA. THE FIRST ENTRY REFERS TO PRISONER P1'S PRISON TERM AND THE SECOND ONE TO PRISONER P2'S PRISON TERM.

P1 \ P2	Confess	Not Confess
Confess	(5,5)	(0,10)
Not Confess	(10,0)	(1,1)

strategies, we mean that a node bases its decision on the past behavior of the nodes in the system. The second feature, which we would like an acceptance algorithm to have, is protection from exploitation. Finally, the algorithm must be scalable.

Our problem falls in the framework of Non-Cooperative Game Theory [10]. There, the canonical example is that of the Prisoners Dilemma. In this example, two people are accused of a crime. The prosecution promises that, if exactly one confesses, the confessor goes free, while the other goes to prison for ten years. If both confess, then they both go to prison for five years. If neither confesses, they both go to prison for just a year. Table I presents the punishment matrix showing the years of prison that the players get depending on the decision they make. Clearly, the mutually beneficial strategy would be for both not to confess. However, from the perspective of the first prisoner, P1, his punishment is minimized if he confesses, irrespective of what the other prisoner, P2, does. Since the other prisoner argues similarly, the unique Nash Equilibrium is the confess strategy for both prisoners. Nevertheless, if this game were played repeatedly (Iterated Prisoners Dilemma), it has been shown that cooperative behavior can emerge as a Nash equilibrium. By employing behavioral strategies, a user can base his decision on the outcomes of previous games. This allows the emergence of cooperative equilibrium. A well known strategy to achieve this desirable state of affairs is the Generous TIT-FOR-TAT (GTFT) strategy [11]. In the Generous TIT-FOR-TAT strategy, each player mimics the action of the other player in the previous game. Each player, however, is slightly generous and on occasion cooperates by not confessing even if the other player had confessed in the previous game. We have adapted the GTFT algorithm to our problem.

In our algorithm, each node maintains a record of its past experience by using the two variables $\psi_h^{(j)}$ and $\phi_h^{(j)}$, $h = 1, \ldots N$, $j = 1, \ldots K$, defined in Section II. Each node therefore maintains only information per session type and does not maintain individual records of its experience with every node in the network.

The decisions are always taken by the relay nodes based only on their $\psi_h^{(j)}$ and $\phi_h^{(j)}$ values. First, consider the case with N nodes, K classes, $q(1) = 1$ and $M = 1$, i.e., each session uses only one relay. Assume that a generic node h receives a relay request for a type j session. Let ϵ be a small positive number. The acceptance algorithm, which we call the GTFT algorithm is as follows.

- If $\psi_h^{(j)}(k) > \tau_j$ or $\phi_h^{(j)}(k) < \psi_h^{(j)}(k) - \epsilon$ Reject
- Else Accept .

Thus, a request for a type j session is refused if either *(i)* $\psi_h^{(j)}(k) > \tau_j$, i.e., node h has relayed more traffic for type j sessions than what it should, or *(ii)* $\phi_h^{(j)}(k) < \psi_h^{(j)}(k) - \epsilon$, i.e., the amount of traffic relayed by node h in sessions of type j is greater than the amount of traffic relayed for node h by others in type j sessions. Since ϵ is positive, nodes are a little generous by agreeing to relay traffic for others even if they have not received a reciprocal amount of help. The GTFT algorithm has the following desirable properties. *(i)* It is not a stationary strategy. *(ii)* Each node takes its action based solely on locally gathered information; this prevents a node from being exploited. *(iii)* Only $4K$ variables need to be stored at each node, independently of N, and this makes GTFT scalable.

Let us now consider the multiple relay case. While for the single relay case, GTFT attempts to equalize the amount of cooperation a node provides with the amount of cooperation it receives, when multiple relays are used, the amount of help rendered is always more than the amount of help received. This is because a node is a relay more often than it is a source. We therefore modify the GTFT algorithm as follows, and call this version of the algorithm m-GTFT. Assume that a relay request for a type j session arrives at node h belonging to class i. The acceptance algorithm becomes,

- If $\psi_h^{(j)}(k) > \tau_j$ or $\phi_h^{(j)}(k) < L_{ij}\psi_h^{(j)}(k) - \epsilon$ Reject
- Else Accept

where L_{ij} is defined as in (5).

V. NASH EQUILIBRIUM OF THE GTFT ALGORITHM

We now prove that the GTFT algorithm constitutes a Nash Equilibrium and show that similar arguments can be extended to prove the convergence of the m-GTFT algorithm.

We first consider the case where all nodes belong to the same class and routes include one relay only (i.e., $q(1) = 1$, $M = 1$). For the sake of simplicity, we drop the session type index in the following theorem.

Theorem 1: Consider a system of N nodes, with all nodes belonging to the same class and having energy constraint ρ. Assume $q(1) = 1$ and $M = 1$. Then,

1) If all nodes except node h are employing GTFT, then $\limsup_{k\to\infty} \phi_h(k) \leq N\frac{\rho}{2}$
2) If all nodes employ GTFT, then all $\phi_h(k)$ $(h = 1, \ldots N)$ converge to $\tau = \frac{N\rho}{2}$.

Proof: See the appendix. ∎

The first part of Theorem 1 shows that if node h tries to deviate from the GTFT strategy, then it cannot achieve throughput greater than the rational Pareto optimal value. The second part of the theorem shows that GTFT results in the rational Pareto optimal point.

We can now extend the proof to the case with multiple classes and a single relay, i.e., $K > 1$, $q(1) = 1$ and $M = 1$.

TABLE II

RATIONAL AND PARETO OPTIMAL VALUES OF THE NARS.

	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	0.84	0.49	0.30	0.20	0.12
Class 2	0.49	0.49	0.30	0.20	0.12
Class 3	0.30	0.30	0.30	0.20	0.12
Class 4	0.20	0.20	0.20	0.20	0.12
Class 5	0.12	0.12	0.12	0.12	0.12

***Theorem** 2:* Consider a system of N nodes with K classes, $q(1) = 1$, $M = 1$, n_i nodes in class i, $i = 1, \dots K$, and energy constraints $\rho_1 > \rho_2 > \dots > \rho_K$. Then,

1) If all nodes except node h are employing GTFT, then $\limsup_{k\to\infty} \phi_h^{(j)}(k) \leq \tau_j$
2) If all nodes employ GTFT, then all $\phi_h^{(j)}(k)$ converge to τ_j $(h = 1, \dots N;\ i, j = 1, \dots K)$.

Proof: See the appendix. ■

From Theorems 1 and 2, it is easy to show, by using randomizing arguments, that m-GTFT also constitutes a Nash Equilibrium and converges to the rational and Pareto optimal operating point.

***Theorem** 3:* Consider a system with N nodes, K classes, $M > 1$, $q(l) > 0$, $l = 1, \dots M$, n_i nodes in class i, $i = 1, \dots K$, and energy constraints $\rho_1 > \rho_2 > \dots > \rho_K$. Then,

1) If all nodes except node h are employing m-GTFT, then $\limsup_{k\to\infty} \phi_h^{(j)}(k) \leq \tau_j$
2) If all nodes employ m-GTFT, then all $\phi_h^{(j)}(k)$ converge to τ_j $(h = 1, \dots N;\ i, j = 1, \dots K)$.

Proof: See the appendix. ■

***Corollary** 1:* It follows from parts 1) and 2) of Theorem 3 that all nodes employing m-GTFT constitutes a Nash Equilibrium.

VI. RESULTS

In this section, we investigate the behavior of the GTFT and m-GTFT algorithms by simulation.

First, we focus on the single relay case. We consider a system with five classes, and five nodes in each class (N = 25). The energy constraints are given by ρ_1 = 0.03, ρ_2 = 0.025, ρ_3 = 0.02, ρ_4 = 0.015 and ρ_5 = 0.01. Also, we assume $q(1) = 1$ and $M = 1$, i.e., the route between the source and the destination node includes exactly one relay. The rational and Pareto optimal values of NARs are shown in Table II, where the entry corresponding to the i^{th} row and j^{th} column equals the rational optimal NAR that we obtain when the source belongs to class i and the relay to class j, i.e., the session type is equal to $\max(i, j)$. These values were derived by solving the system of linear equations as in Example 1 in Section III.

We study convergence of the proposed strategy by assuming that all nodes employ GTFT as their acceptance algorithm. The results show that the NAR values converge to the desired rational Pareto optimal values. The NARs associated with the different session types are presented in Fig. 2, as a function of time. For the sake of simplicity, in the plot, the evolution of the NARs is shown for just one node per each session type. We note that all NARs converge to the values reported in Table II, i.e., to the rational optimal values.

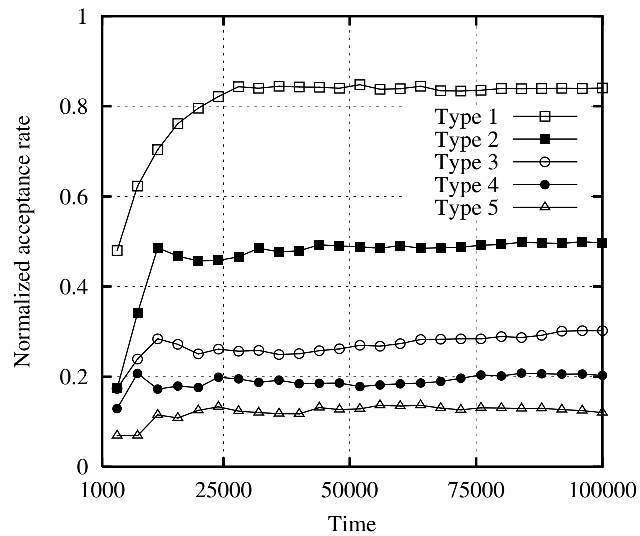

Fig. 2. NAR versus time when $N = 25$, $K = 5$, $q(1) = 1$, $M = 1$, and all nodes employ GTFT. NAR values converge to the optimal operating point.

In Fig. 3, we show that it is critically important that the parameter ϵ, introduced in Section IV, be positive. In other words, nodes should always be slightly generous for the NARs to achieve the rational optimal values. The results presented in Fig. 3 were obtained by setting ϵ to -0.01. In this case the NAR values converge to 0, or equivalently the system throughput goes to 0, under scoring the importance of being generous. When ϵ is equal to 0, the nodes behavior depends on the initial value of ψ and ϕ.

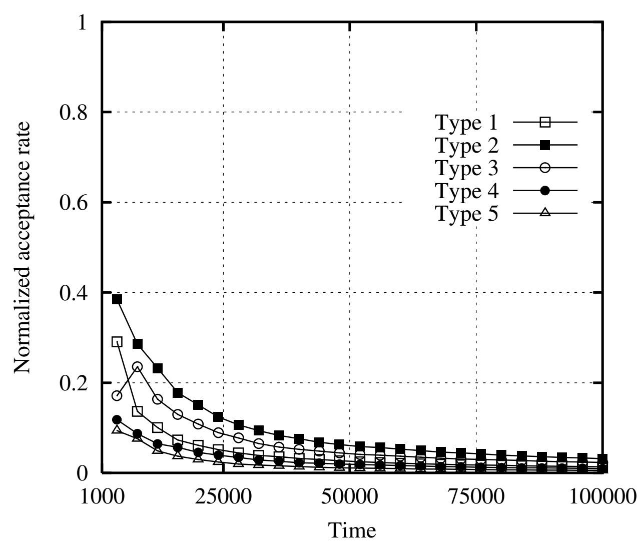

Fig. 3. NAR versus time when $N = 25$, $K = 5$, $q(1) = 1, M = 1$, all nodes employ GTFT, and $\epsilon < 0$. If nodes are not slightly generous ($\epsilon > 0$), GTFT fails to reach the optimal operating point.

Next, we study the robustness of the GTFT algorithm in the presence of parasites. We assume that a node in class 2 and a node in class 4 are parasitic, i.e., these nodes never relay traffic. Figure 4 shows the NAR as a function of time, for the different session types in the system. We see that the performance of type 2 and type 4 sessions degrade severely while performance for other types of sessions remain unaffected. This implies that, since nodes are self-interested and rational, they have no motivation to behave in a parasitic manner. Notice that if some node adopts a strategy such that it relays less traffic than it should, then its throughput suffers. This is because the GTFT is a Nash equilibrium.

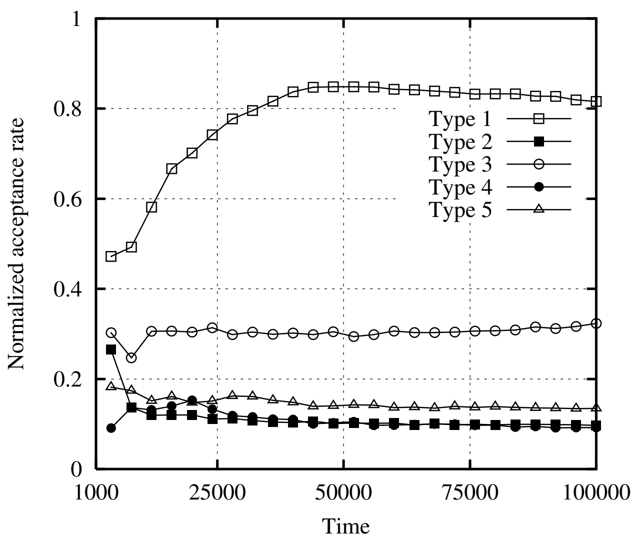

Fig. 4. NAR versus time when $N = 25$, $K = 5$, $q(1) = 1$, $M = 1$, and one node in class 2 and one node in class 4 are parasites while all other nodes employ GTFT. Performance of nodes in type 2 and type 4 sessions degrade showing that GTFT prevents parasitic behavior in rational users.

We mention in passing that similar results were obtained when the energy consumed per session was assumed to be an i.i.d. random variable with unit mean.

We now focus our attention on the case of multiple relays and study the system performance when all the network nodes adopt the m-GTFT algorithm. We consider a system with two classes, and six nodes in each class. We assume $q(1) = q(2) = 0.5$, and $M = 2$. The energy constraint for nodes in class 1 is equal to 0.03 and for those in class 2 is equal to 0.015. The optimal NAR values are obtained as described in Example 2 in Section III. Figure 5 shows the evolution in time of the NAR for the two types of sessions. We see that in this case too, the NARs converge to their optimal values.

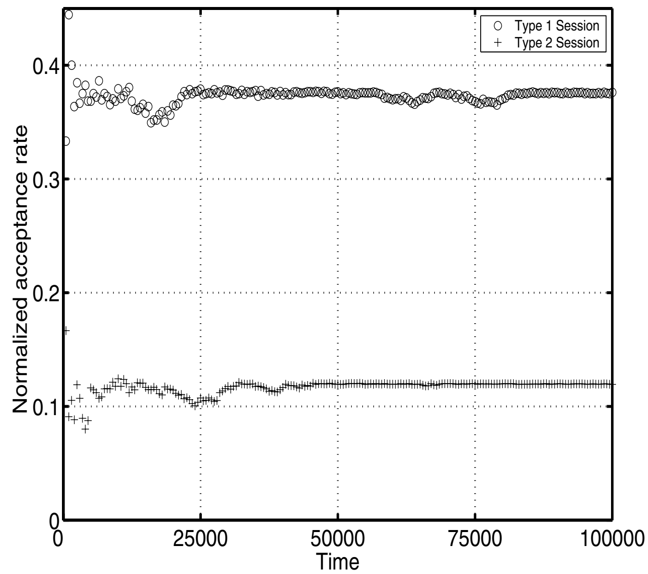

Fig. 5. Convergence of m-GTFT for $N = 12$, $K = 2$, $\rho_1 = 0.03$, $\rho_2 = 0.015$, $q(1) = q(2) = 0.5$, and $M = 2$.

VII. Discussion

In this work, our objective is to provide a mathematical framework for studying user cooperation in ad hoc networks and to define behavioral strategies that lead the system to the optimal operating point. Several implementation aspects however need to be addressed. In this section, we briefly discuss some of these issues.

So far we have assumed that each user possesses sufficient information about the system in order to calculate the optimal values of NARs. This requires each user in the system to be aware of the number of users in each energy class and the energy constraint for each class. Since, by their very nature, ad hoc networks should not rely on a centralized database, we need to devise a distributed mechanism to acquire and disseminate the necessary information to all users. For example, users can exchange their view of the system whenever they interact. However, the algorithm should be sufficiently robust to prevent users from propagating incorrect information to serve their own needs.

In our model, we have made the critical assumption that users are only rational and selfish, but are not malicious. A malicious user, as opposed to a selfish user, is willing to wreak havoc in the network even at the expense of his own throughput. For instance, a malicious user may always deny relay requests. Such a user can rapidly deteriorate the performance of the nodes belonging to the same class, as shown in Fig. 4. A watchdog like mechanism, as proposed in [5], may be employed to identify such users and a Pathrater like mechanism can be adopted to avoid relaying through such users.

Finally, we propose that the m-GTFT algorithm can be implemented by modifying the current AODV routing algorithm [13]. In the AODV algorithm, when a source needs a route to a destination node, it sends a route request (RREQ) packet to its neighbors. As the RREQ propagates to the destination, every intermediate node can append to the packet its class identifier, along with its address. Once the destination receives a RREQ, it sends back a route reply (RREP) packet over the same path followed by the RREQ it received. Since the type of session is determined by the nodes on the route, the destination can add the session type tag to the RREP message. As the RREP propagates back to the source, the intermediate relay nodes can easily implement m-GTFT.

VIII. CONCLUSIONS

Ad hoc networks hold the key to the future of wireless communication, promising adaptive connectivity without the need for expensive infrastructure. In ad hoc networks, the lack of centralized control implies that the behavior of individual users has a profound effect on network performance. For example, by choosing to leave a network or refusing to honor relay requests, a user can severely inhibit communication between other users. This is a stark contrast with fixed wireless systems where a single user has much less influence on other users. The influence of user behavior on network performance, in combination with the fact that nodes in an ad hoc network are constrained by their finite energy capacity, motivates the need for a rational and efficient resource allocation scheme.

In this paper, we addressed the problem of cooperation among energy constrained nodes in wireless ad hoc networks. We assumed that users are rational and showed that as a consequence users will not always be willing to expend their energy resources to relay traffic generated by other users. By using elementary game theory, we were able to show the existence of an operating point which optimally trades off throughput with lifetime. We devised simple and scalable behavioral strategies namely, GTFT and m-GTFT, which were shown to constitute a Nash equilibrium. We also proved that these algorithms lead the system towards the optimal operating point.

We would like to emphasize that the aim of this work was to provide a mathematical framework for studying user cooperation in ad hoc networks, and to define strategies leading to an optimal user behavior. Further research is required to devise an algorithm that enables the nodes to accrue over time the system information needed to implement the proposed strategies.

REFERENCES

[1] V. Rodoplu and T.H. Meng, "Minimum Energy Mobile Wireless Networks," *IEEE Journal on Selected Areas in Communications,* Vol. 17, No. 8, August 1999, pp. 1333–1344.

[2] J.-H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad Hoc Networks," *Proc. of INFOCOM 2000,* Tel Aviv, Israel, March 2000.

[3] A. Michail and A. Ephremides, "Energy Efficient Routing for Connection-Oriented Traffic in Ad Hoc Wireless Networks," *Proc. of the 11th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC),* 2000.

[4] V. Srinivasan, P. Nuggehalli, C-F. Chiasserini, and R. R. Rao, "Optimal Rate Allocation and Traffic Splits for Energy Efficient Routing in Ad Hoc Networks," *Proc. of Infocom 2001,* New York City, June 2001.

[5] S. Marti, T.J. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," *Proc. of MobiCom 2000,* Boston, August 2000.

[6] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J.P. Hubaux, and J.Y. Le Boudec, "Self-Organization in Mobile Ad-Hoc Networks: the Approach of Terminodes," *IEEE Communications Magazine*, Vol. 39 No. 6, June 2001.

[7] L. Buttyan and J.P. Hubaux, "Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks," *Technical Report No. DSC/2001/046,* August 2001.

[8] L. Buttyan and J.P. Hubaux, "Enforcing Service Availability in Mobile Ad-Hoc WANs," *Proc. of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC),* Boston, MA, USA, August 2000.

[9] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini, and R.R. Rao,"Energy Efficiency of Ad Hoc Wireless Networks with Selfish Users," *European Wireless Conference 2002 (EW2002)*, Florence, Italy, February 2002.

[10] R.B. Myerson, *Game Theory: Analysis of Conflict,* Harvard University Press, Cambridge, Mass., 1991.

[11] R. Axelrod, *The Evolution of Cooperation,* Basic Books, New York, 1984.

[12] D.P. Bertsekas and J.N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, Belmont, 1996.

[13] C.E. Perkins, E.M. Royer, "Ad-Hoc On-Demand Distance Vector Routing," *Proc. of Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, 1999, pp. 90–100.

APPENDIX

Here, we prove the theorems presented in Section V.

***Theorem** 1:* Consider a system of N nodes, with all nodes belonging to the same class and having energy constraint ρ. Assume $q(1) = 1$ and $M = 1$. Then,

1) If all nodes except node h are employing GTFT, then $\limsup_{k\to\infty} \phi_h(k) \leq N\frac{\rho}{2}$
2) If all nodes employ GTFT, then all $\phi_h(k)$ $(h = 1, \ldots N)$ converge to $\tau = \frac{N\rho}{2}$.

Proof: The first part of the theorem follows from the fact that $N-1$ users excluding h are employing GTFT. We know that a node u employing the GTFT scheme rejects a relay request whenever $\psi_u(k) > \frac{N\rho}{2}$; thus we have: $\limsup_{k\to\infty} \psi_u(k) \leq \frac{N\rho}{2}, u \neq h$. Since the acceptance mechanism in GTFT is independent of the source identity, each user receives the same amount of help (ϕ). Hence $\limsup_{k\to\infty} \phi_u(k) \leq N\frac{\rho}{2}, u = 1, \ldots N$.

We now prove that $\psi_h(k)$ and $\phi_h(k)$ converge to the same value. In order to do so, for the generic node h we define

$$\alpha_h(k) = \frac{\text{number of successful relay requests made by } h \text{ till } k}{k}$$
$$\beta_h(k) = \frac{\text{number of sessions relayed by } h \text{ till } k}{k}. \quad \text{(A-1)}$$

We call $\alpha_h(k)$ the node traffic flow, and write the average traffic flow as $\alpha_h = \lim_{k\to\infty} \alpha_h(k)$. Henceforth, we shall assume that this limit exists. Recall that the source is chosen randomly from the N nodes and the relay is chosen randomly from the remaining $N-1$ nodes. We can derive the following correspondence between average flow and NAR,

$$\alpha_h = \lim_{k\to\infty} \frac{\text{no. of successful relay requests made by } h}{k} \cdot \frac{\text{no. of sessions relayed by } h}{\text{no. of sessions relayed by } h}$$
$$= \frac{\text{NAR}}{N(N-1)}. \quad \text{(A-2)}$$

Due to the linear relationship between the flows and NARs shown in (A-2), it is easy to see that the NARs converge *iff* the flows converge. Moreover, since the total number of successful requests made by all nodes must equal the total number of requests relayed by all nodes, we see that flows are conserved at any time step k. We summarize this as the following Lemma.

***Lemma** 1:*

$$\sum_{i=1}^{N}(\alpha_i(k)-\beta_i(k)) \quad = \quad 0 \qquad \text{(A-3)}$$

Consider node h at time k. As a first step, we prove that $\alpha_h(k)-\beta_h(k)$ converges to 0.

We track the evolution of $\alpha_h(k)$ and $\beta_h(k)$ with the following recursions,

$$\begin{aligned}\alpha_h(k+1) &= \frac{k\alpha_h(k)+1_A}{k+1} \\ \beta_h(k+1) &= \frac{k\beta_h(k)+1_R}{k+1} \qquad \text{(A-4)}\end{aligned}$$

where we have

$$1_A = \begin{cases} 1 \text{ if } h \text{ is a source and its relay request is accepted} \\ 0 \text{ else} \end{cases}$$

$$1_R = \begin{cases} 1 \text{ if } h \text{ is a relay and accepts a relay request} \\ 0 \text{ else.} \end{cases}$$

Then, we define

$$r(k) = [\alpha_1(k)-\beta_1(k), \dots \alpha_N(k)-\beta_N(k)]^T, \qquad \text{(A-5)}$$

thus the recursion on $r(k)$ is as follows,

$$r_h(k+1) = \begin{cases} r_h(k)+\frac{1}{k+1}(-r_h(k)), \\ \text{if } h \text{ is neither a source nor a relay, or} \\ h \text{ is source and its request is rejected, or} \\ h \text{ is a relay and rejects a request} \\ r_h(k)+\frac{1}{k+1}(-r_h(k)+1), \\ \text{if } h \text{ is a source and its request is accepted} \\ r_h(k)+\frac{1}{k+1}(-r_h(k)-1), \\ \text{if } h \text{ is a relay and accepts a request.} \end{cases}$$

We can re-write the recursion on $r(k)$ as,

$$r(k+1) = r(k) + \frac{1}{k+1}(-r(k)+w(k)) \qquad \text{(A-6)}$$

where, $w(k)$ is a random variable taking values in $\{-1,0,1\}$. We would like to show that the sequence $\{r_k\}$ converges to point $r^\star = [0,\dots 0]^T$ when the GTFT algorithm is used. This will imply that $\alpha_h(k)$ - $\beta_h(k)$ converges to 0, i.e., $\psi_h(k)$ - $\phi_h(k)$ converges to 0. To prove this, we use the following corollary [12].

***Corollary** 2:* Consider a sequence $\{q(k)\}$, such that

$$\begin{aligned} q(k+1) &= q(k)+\gamma(k)s(w(k),q(k)) \\ \sum_{k=1}^{\infty}\gamma(k) &= \infty \\ \sum_{k=1}^{\infty}\gamma^2(k) &< \infty. \qquad \text{(A-7)}\end{aligned}$$

Define $\overline{s}(q) = E[s(q,w)]$. Then, if,

(a) $(q^\star - q)^T\overline{s}(q) \geq C_1\|q^\star - q\|^2$ for some $C_1 > 0$,

(b) $E[\|s(q,w)\|^2] \leq C_2[\|q^\star - q\|^2 + 1]$ for some $C_2 > 0$,

we have: $lim_{k\to\infty}q(k) = q^\star$ with probability 1.

We need to show that $r(k)$ converges to $r^\star$. By considering: $\gamma(k) = 1/(k+1)$ and $s(q,w) = -r(k)+w(k)$, we see that (A-6) satisfies (A-7). It is easy to verify that condition *(b)* of Corollary 2 is satisfied for sufficiently large C_2. We need to show that condition *(a)* holds, i.e.,

$$(-r)^T\overline{s}(r) \geq C_1\|-r\|^2. \qquad \text{(A-8)}$$

At time step k, assume that m out of the N nodes are accepting relay requests, and $N-m$ nodes are rejecting requests. In other words, $\phi_h(k)-\psi_h(k) > \epsilon, h = 1,\dots m$, and $\phi_h(k)-\psi_h(k) < \epsilon, h = m+1,\dots N$. Correspondingly for the flows, for some $\delta > 0$, $\alpha_h(k)-\beta_h(k) > \delta, h = 1,\dots m$ and $\alpha_h(k)-\beta_h(k) < \delta, h = m+1,\dots N$.

Also, recall that the probability that a node generates a relay request in a time step is equal to $1/N$. Then, the event that a node belonging to the set of the m accepting nodes makes a request and that its request is accepted occurs with probability $(m-1)/[N(N-1)]$. While, the node will receive a relay request, that it will accept, with probability $(N-1)/[N(N-1)]$. Likewise, the event that a node in the set of the rejecting nodes generates a relay request and that its request is accepted has probability $m/[N(N-1)]$. While, the probability that it will accept a request is equal to 0. From the above considerations, it is easy to see that

$$\overline{s}_h(r(k),w(k)) = \begin{cases} -r_h(k)+\frac{m-N}{N(N-1)} \\ \text{if } h = 1,\dots m \\ -r_h(k)+\frac{m}{N(N-1)} \\ \text{if } h = m+1,\dots N. \end{cases} \qquad \text{(A-9)}$$

We obtain,

$$(-r_k)^T\overline{s}(r_k) = \|r(k)\|^2 - \frac{1}{N(N-1)}\sum_{h=1}^{m}(m-N)r_h(k) - \frac{1}{N(N-1)}\sum_{h=m+1}^{N}mr_h(k). \qquad \text{(A-10)}$$

By using Lemma 1, we have

$$\begin{aligned}(-r_k)^T\overline{s}(r_k) &= \|r(k)\|^2 + \frac{1}{N-1}\sum_{h=1}^{m}r_h(k) \\ &> \|r(k)\|^2 + \frac{m}{N-1}\delta \\ &> \|r(k)\|^2. \qquad \text{(A-11)}\end{aligned}$$

Therefore, *(a)* is satisfied for $C_1 = 1$ and Corollary 2 can be applied. We have: $lim_{k\to\infty}r(k) = r^\star$ with probability 1, i.e., $\alpha_h(k)-\beta_h(k)$ and, hence, $\phi_h(k)-\psi_h(k)$ converge to zero for each h.

We know that for a node h employing the GTFT scheme $\limsup_{k\to\infty}\psi_h(k) \leq \frac{N\rho}{2}$. We also know that, since $lim_{k\to\infty}\psi_h(k)-\phi_h(k) = 0$, $\liminf_{k\to\infty}\psi_h(k) \geq \frac{N\rho}{2}$. This is because, if a node h uses the GTFT algorithm and $\psi_h(k) \leq \frac{N\rho}{2}$, it will always accept a relay request when $\psi_h(k) - \phi_h(k) = 0$, thereby increasing $\psi_h(k)$. It follows that: $\liminf_{k\to\infty} \psi_h(k) \not< \frac{N\rho}{2}$. We can conclude that: $lim_{k\to\infty}\psi_h(k) = \frac{N\rho}{2}$. Since $\psi_h(k) - \phi_h(k)$ goes to zero, $lim_{k\to\infty}\phi_h(k) = \frac{N\rho}{2}$. ■

***Theorem** 2:* Consider a system of N nodes with K classes, $q(1) = 1$, $M = 1$, n_i nodes in class i, $i = 1, \ldots K$, and energy constraints $\rho_1 > \rho_2 > \ldots > \rho_K$. Then,

1) If all nodes except node h are employing GTFT, then $\limsup_{k\to\infty} \phi_h^{(j)}(k) \leq \tau_j$
2) If all nodes employ GTFT, then all $\phi_h^{(j)}(k)$ converge to τ_j $(h = 1, \ldots N;\ i, j = 1, \ldots K)$.

Proof: Here, we are essentially randomizing between the K types of GTFT. If we consider all sessions of type j, then all nodes involved in sessions of type j, behave as if they had the same energy constraint ρ_j. From Theorem 1, we see that if we consider sessions of type j, alone, then $\psi_h^{(j)}(k)$ and $\phi_h^{(j)}(k)$ will converge. Hence these values will converge for all the session types eventually. ■

***Theorem** 3:* Consider a system with N nodes, K classes, $M > 1$, $q(l) > 0, l = 1, \ldots M$, n_i nodes in class i, $i = 1, \ldots K$, and energy constraints $\rho_1 > \rho_2 > \ldots > \rho_K$. Then,

1) If all nodes except node h are employing m-GTFT, then $\limsup_{k\to\infty} \phi_h^{(j)}(k) \leq \tau_j$
2) If all nodes employ m-GTFT, then all $\phi_h^{(j)}(k)$ converge to τ_j $(h = 1, \ldots N;\ i, j = 1, \ldots K)$.

Proof: For the sake of brevity, we provide a rough sketch of the proof. We can classify each session based on the number of relays used. We say that the session employing l relays is a l-relay session. For a fixed l, we can show that $\psi_h^{(j)}(k)$ and $\phi_h^{(j)}(k)$ converge, by using the same arguments as in Theorem 2 and by appropriately scaling Lemma 1. By adding these variables with the appropriate weights (i.e., $q(l)$, $l = 1 \ldots M$), the theorem is proved. ■