Survey of
Mobile Wireless Ad-Hoc Networks

Stanley Kai Him Chiu
Department of Computer Science
University of British Columbia

December 9, 2005

Abstract

Mobile wireless ad-hoc networks have recently become a popular research topic. They have been successfully deployed by groups such as the military, in which agents are all cooperative. However, civilian devices that utilize this technology are now becoming affordable and are very common. Mobile wireless ad-hoc networks can be deployed amongst them, allowing these devices to communicate data with each other. Because civilians are agents who want to maximize their own utility, they can be thought of as non cooperative agents. Without some form of punishment mechanism, selfish agents will not help in the operation of such networks and will cause the network to be very inefficient. In mobile wireless ad-hoc networks (MANETs), nodes must relay packets for other nodes in order for the ad-hoc network to function properly, but selfish agents have no incentive to do so. This paper looks at mechanisms that have been designed to solve these problems with emphasis on how these systems have been modeled as multi agent systems and how game theory has been used in each mechanism.

1 Introduction

Mobile wireless ad-hoc networks (MANETs) have gained the interest of many researchers recently due to advances in technology. Researchers have shown that selfish agents in a MANET can greatly decrease the performance of the network, to the point where no messages can be sent at all [3]. Much of the recent literature in this field has been devoted to coming up with mechanisms and algorithms which alleviate this problem, allowing MANETs with non-cooperative agents to maintain high performance. The most researched problem is the forwarding problem. In MANETs, due to the low power and low range capabilities of the wireless transmitters, traffic often has to be relayed by at least one other node, but selfish agents will choose not to forward if no punishment mechanism is implemented. This can be represented as a multi agent system, as will be shown in later sections.

This paper discusses several mechanisms with emphasis on how multi agent systems have been used to model these problems and how game theory has been used to find ways to alleviate this problem. The following sections describe the relay problem (*section 2*), the solutions they have found with emphasis on multi agent systems and game theory (*section 3*), and a possible research idea (*section 4*).

2 Relay Problem

One of the most researched problems in this field is the relay problem. In MANETs, due to the low power and low range capabilities of the wireless transmitters, traffic often has to be relayed by at least one other node.

This can be viewed as a multi agent system in which each node tries to maximize its own utility. Because relaying other nodes' messages consumes energy, this situation is similar to the prisoner's dilemma. If all the nodes relay with a high probability, then the network functions well and nodes have high utility. However, if a node sets its own relay probability to a high probability, but other nodes do not, then other nodes have high utility, but the node itself has low utility. It should also be noted that without any sort of punishment mechanism, if a node sets its forwarding probability to 0, the utility of the node will always be higher than if it set its forwarding probability to anything higher. This is because other nodes do not change their forwarding rates and the node itself uses less energy. Therefore, a selfish node will never forward in this situation. If no agent forwards any message, then agents can only communicate directly with other agents within the range of their low power transmitter, and the network will not function properly.

3 Mechanisms to Alleviate the Problem

Much of the recent literature on MANETs have concentrated on designing mechanisms which solve this problem. The general approach in designing these mechanisms is to model the network as a multi agent system. A utility function is then defined, and mechanisms are designed so that agents who optimizing this utility function will follow a behavior that allows the network to function efficiently. The well referenced mechanisms tend to fall into one of three categories, which are simple punishment mechanisms, reputation mechanisms, and price mechanisms. The following subsections discuss these mechanisms and how game theory has been applied to help design mechanisms, evaluate mechanisms, and also help set variables within the mechanisms.

3.1 Simple Punishment Mechanisms

Using ideas from game theory, simple punishment mechanisms have been designed to solve this problem [1] [6] . Altman et al [1] and Srinivasan et al [6] attempt to solve this non cooperative game problem by looking at previous work done in game theory. Both papers come to the conclusion that tit-for-tat is a good algorithm that can be used to provide incentives for nodes to cooperate.

The simple mechanism devised in [1] formulates the utility as:

$$U_i(\gamma) = \sum_{n:(i,n)\in O} \lambda_{in} f_i(p(i,n;\gamma)) + \sum_{n:(n,i)\in O} \lambda_{ni} g_i(p(n,i;\gamma)) - aE_f\xi_i(\gamma) \quad (1)$$

where i is a node, $n : (i, n)$ is the set of n which i has a path to send a message to, γ is the probability that nodes will forward packets, λ is the flow rate, f_i is the utility of node i successfully sending a packet, g_i is the utility of i successfully receiving a packet, $p(s, d; \gamma)$ is the probability that node d receives a packet from node s, and E_f is the energy required to forward a packet. The main idea is that the first term on the right hand side represents the utility received for successfully sending messages. The second term represents the utility received for successfully receiving a message, and the last term is the disutility caused by the lose of energy due to message forwarding.

Using this utility function, a Nash equilibrium is a strategy where for all agents follow one of the strategies that satisfy

$$\gamma^* = argmax_{\gamma_i^{'}} U_i(\gamma_i^{'}, \gamma^{-i}) \quad (2)$$

Altman et al [1] defines a special tit-for-tat strategy in an attempt to get nodes to cooperate and finds properties of the Nash equilibriums under this strategy. The strategy is for all nodes to decrease their own forwarding probability to the minimum forwarding probability amongst all nodes. In other words, if all nodes originally forwarded packets at a rate of γ, and then one node decides to forward at $\gamma^{'} < \gamma$, then all the other nodes will react by setting their forwarding probability to $\gamma^{'}$ to punish that node.

In the original mechanism without punishment, $\gamma = 0$ is a dominant strategy because other agents' forwarding probabilities are independent of the node's own probability. In this new mechanism, $\gamma = 0$ is still always an equilibrium, but there are also other equilibria. It is still an equilibrium because if other nodes do not forward, then it is obviously in a node's best interest to also conserve its own energy and not forward. Note that γ can be much higher than 0 in the case where the amount of energy required to forward packets, aE_f, and the number of hops required for a message to be sent, L, are both small. Intuitively, this is because low aE_f and low L values mean forwarding messages is relatively cheap. Therefore, increasing the forwarding probability only increases the third term of equation 1 slightly but increases the first 2 terms greatly since f_i and g_i will both increase. However, as either L or aE_f increases, the cost of forwarding messages increase and the equilibrium γ will fall. γ approaches 0 as L or aE_f approach infinity, which degenerates it back to the case of not having a punishment mechanism.

The tit-for-tat mechanisms suffers from problems because it is overly simple and only sets a universal forwarding rate instead of setting one for each node. As shown in the previous section, a forwarding probability of 0 is always a equilibrium, which means that the mechanism does not always converge to a good point. It should also be noted that in real life, there are likely to be agents (nodes) who are either malicious or faulty. Under this mechanism, such nodes will bring equilibrium to 0 and thus make the network inefficent.

3.2 Reputation Mechanisms

Reputation mechanisms define a method for keeping track of a node's history by monitoring the node's past actions, and then uses this history in order to decide whether or not to forward packets for that node [2] [5]. This is basically a more complicated punishment system in which each node is tracked separately and punished only based on its own previous actions.

One of the most referenced reputation mechanisms is CORE [5], which stands for a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. The general idea behind CORE is that each node keeps track of a reputation table, in which the reputation of a node is decreased whenever a node does not forward a packet, and is increased whenever it does. When a request for relaying a packet arrives at a node, the node forwards if the requester has a positive reputation, and refuses otherwise.

There has been research on how to evaluate mechanisms such as this one using a game theoretical approach. Specifically, Michiardi et al [4] does this for CORE. In the non-cooperative case, [4] formulates a utility function for the nodes, which is

$$u_{n_i}(b_i, b_j) = E_{self} * (1 - b_i) * r_{n_i} - b_i * f * (E_R + E_{PF}) \tag{3}$$

where E_{self} is the amount of energy a node spends for its own communication, E_R is the amount of energy spent due to participation in the protocol, E_{PF} is the amount of energy spent relaying packets, b_i is the ratio of the energy node i spends for relaying and participation in the protocol to the total energy spent, f is just a multiplication factor, and r_{n_i} is the reputation of node i according to its neighbors. [4] put the r_{n_i} term on the second term in equation 3, but I believe it was a mistake and he actually meant to put it on the first term as shown. My reasoning is that reputation affects the amount of utility an agent gains by putting a certain amount of energy into sending messages, since high reputation means all the packets sent with that energy will get forwarded, while low reputation means only a small amount of packets will get to their destination.

Rather than using a mathematical proof, [4] graphs a selfish node's optimal behavior and shows that the equilibrium behavior approaches using half of its energy for its own communication and using the other half for relaying and supporting the protocol. The explanation for this behavior is that the selfish node starts off with neutral reputation. Thus, it can maximize its own utility by not relaying until its reputation has decreased enough to effect the forwarding rate for it greatly. From the formula, it can be seen that as r_{n_i} drops, the first term in equation 3 drops and thus the node starts cooperating by spending more than half of its energy forwarding and communicating in the protocol. When its reputation is high, it will spend more energy on its own communication. When b_i is graphed against time, this results in a sinusoid wave centered around $b_i = 0.5$ and decreasing in magnitude over time. In other words, this settles to the equilibrium previously stated over time.

3.3 Price Mechanisms

Another approach to solving this problem is the implementation of pricing mechanisms. In these mechanisms, money is charged to nodes that cause traffic and money is given to nodes that help forward packets. One well referenced mechanism that does this is SPRITE [7]. This mechanism not only uses game theory to describe node's behaviors, but it also models reporting payments as another multi agent game and uses game theory to help decide how to set the payments.

In SPIRTE, there is a central Credit Clearance Service (CCS) which the mobile nodes can send receipts to when they are connected to a fast network, such as through a broadband connection at home computer. In this system, the destination and any node that forwards a message keep receipts for the message, and give these receipts to the CCS in exchange for money. If a message gets from the sender to the destination, then all nodes that forwarded the packet receive a payment of α and the destination node receives β. The sender has to pay a total of $(n*\alpha+\beta)$, where n is the number of nodes from the source to the destination. The amount the sender pays is the amount that the other nodes receive, so this is budget balanced in the case where the message successfully arrives. However, if the message does not get to the destination, then each node that forwarded the message gets $\gamma*\alpha$, and the last node to receive the message (the last node to send a receipt) gets $\gamma*\beta$, and the sender has to pay $n*\alpha+\beta-2*\gamma*\beta)$. Note that γ is a number less than 1, and its purpose is to make nodes receive less money if the message does not get to the destination, therefore providing more incentive for nodes to forward and prevents cheating. Note also that β is less than α in order to make it more profitable for a node to forward a message and receive α instead of not forwarding and receiving β.

One important part of this mechanism is deciding on the right numbers. The paper analyzes whether nodes will tell the truth by modeling the decision as a game. Consider a game where there are d players, each of which is on the forwarding path of the message, and each player has two available actions, which is to either send the receipt, or not send the receipt. Consider a message sent by node s to node d that reaches only node e. According to the SPRITE algorithm, the payment function is

$$P_i = \begin{cases} \alpha & i < e = d \\ \beta & i = e = d \\ \gamma*\alpha & i < e < d \\ \gamma*\beta & i = e < d \end{cases} \tag{4}$$

If a node is e or is between node s and node e, then they receive P_i by telling the truth and sending the receipt, or receive 0 otherwise. Therefore, telling the truth is obviously the dominant strategy.

Consider the case where nodes collude. If a node c between e and the destination wants to receive a payment, a node in between nodes s and e must send the receipt to c. The receipt is smaller than the full message, but will still incur a forwarding cost δ to the forwarding node. Therefore, c must compensate the node forwarding the receipt by δ, so the total amount of money c will receive is $P_i - \delta$. Since it is the new last node, its P_i will be $\gamma * \beta$. Therefore, if $\delta \geq \gamma * \beta$, then $P_i - \delta$ will be negative. Therefore, truth telling is dominant, and this is the way the numbers in the mechanism should be set.

An interesting thing to note is that in this proof, the paper assumes that when agents collude, it will cost an agent δ to send the receipt to another agent. However, if agents need some external connection to connect to a central server periodically to send receipts, then I believe nodes can also set up some automatic mechanism to trade receipts at virtually no cost right before sending receipts to the central server. It will be in a node's best interest to do this when they are saving battery power and would not have forwarded anyways. This will modify the equilibrium reached under this mechanism in a negative way.

However, even with the truth telling properties of this algorithm, the amount of messages that get from source to destination still depends on the properties of the nodes themselves, such as whether or not they want to be credit balanced. Like the analysis of CORE presented in [6], Zhong et al [7] plots success rate, which is the ratio of messages that successfully reach destination to the total number of messages on a graph. The graph shows that for agents who want to maximize the total amount of messages sent while sending them as early as possible, the success rate approaches 1 as the battery life of the nodes increases.

4 Further Research

Attempts have been made to create game theoretic frameworks for designing and analyzing mechanisms, but they do not seem to fit much of the current research and have not gained popularity. I believe it would be very useful if researchers develop a standard set of tests using agents with different preferences for energy conservation and different preferences for money if applicable. These can be used to evaluate mechanisms to produce comparable results.

5 Conclusion

This paper has looked at MANETs in the non cooperative environment. It has described some recent research in this field and looked at how multi agent systems have been used to model these networks. It has also looked at how game theory has been used to help in both the design and analysis of these mechanisms.

References

[1] E. Altman, A. Kherani, P. Michiardi, and R. Molva. Non-cooperative forwarding in ad-hoc networks. *Technical Report INRIA Report No. RR-5116*, 2004.

[2] S. Buchegger and J. Le Boudec. Performance analysis of the CONFIDANT protocol: Cooperation of nodes - fairness in dynamic ad-hoc networks. *Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC), Lausanne, CH*, June 2002.

[3] Sergio Marti, T. J. Giuli, Kevin Lai, and Mary Baker. Mitigating routing misbehavior in mobile ad hoc networks. *Mobile Computing and Networking*, pages 255–265, 2000.

[4] P. Michiardi and R. Molva. A game theoretical approach to evaluate cooperation enforcement mechanisms in mobile ad-hoc networks. *In Proceedings of WiOpt'03, SophiaAntipolis, France*, March 2003.

[5] R. Molva P. Michiardi. Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. *Institut Eurecom Research Report*, 2001.

[6] V. Srinivasan, P. Nuggehalli, C. Chiasserini, and R. Rao. Cooperation in wireless ad hoc networks. *In Proceedings of IEEE Infocom*, 2003.

[7] S. Zhong, Y. Yang, and J. Chen. Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks. *Technical Report Yale/DCS/TR1235, Department of Computer Science, Yale University*, 2002.