P2P Overlay Topology Control in MANETs

Afzal Mawji and Hossam Hassanein
Telecommunications Research Lab, School of Computing, Queen's University
Kingston, Ontario, Canada, K7L 3N6
{mawji, hossam}@cs.queensu.ca

***Abstract*—P2P applications are enormously popular on the Internet and their uses vary from file sharing to Voice–over–IP to gaming and more. Increasingly, users are moving toward wireless networked devices and wish to continue using P2P applications in these new environments. MANETs are expected to grow in use as wireless mesh and 4G networks increase in popularity. P2P and MANETs share some similarities, such as self-organization, dynamism, and resilience to failure, but it is necessary that P2P algorithms should take advantage of the realities of MANETs. In P2P networks, the overlay peers must form a topology of connections between themselves and this topology should reflect the underlying network in order to reduce delay and energy consumption. We study the results of a game–theoretic topology control algorithm which considers energy and distance between nodes in a P2P network running atop a MANET. We find that the minimum cost topologies are very highly connected and thus resilient, but in most cases the topologies do not stabilize even without peer mobility or churn.**

I. Introduction

Peer–to–peer (P2P) networks are popular among users and they are used for many different applications, such as file sharing, Voice–over–IP, gaming and instant messaging. P2P networks are usually implemented in the form of overlay networks, which consist of higher–layer connections between peers that are independent of the underlying or substrate network. This allows overlay networks to abstract their connectivity to a higher–level view of the peers that make up the network. The overlay is formed by the amalgamation of the logical connections that peers form with one another. Peers communicate with one another via the overlay, with queries and responses possibly being routed over multiple hops. The peers self–organize and must be able to handle "churn", the term used for the constant connections and disconnections of peers as they join and leave the network, meaning P2P networks must be resilient. On the Internet, peers exist on the access links, or "edge" of the network, and many P2P networks in use today scale to millions of simultaneous users.

Mobile ad hoc networks (MANETs) are multi–hop, variable topology networks in which a pre–existing infrastructure is not needed. Instead, devices form a self–organizing network with their peers and any node may act as a client and/or server at any time, so both clients and servers are mobile and resource–constrained. Devices are heterogeneous, resulting in a potentially large variation in link and node capabilities, and due to mobility, the network topology is constantly changing. These autonomous and infrastructureless networks pose particularly challenging design problems, especially with regard to lifetime and scalability.

It is clear that users want to run P2P applications, regardless of what type of network they are using, be it a wired network, a MANET, or a hybrid infrastructure–infrastructureless network such as a wireless mesh or 4G network. Therefore, it is important to consider how to effectively run a P2P overlay in an infrastructureless network such as a cooperative MANET. If the user is connected to a partly–infrastructured network, avoiding the base station and instead communicating with nearby nodes may reduce delay and energy use since transmissions travel a shorter distance and require less power. Furthermore, avoiding communication with the base station may avoid charges from the network operator. Finally, if there are multiple infrastructured networks in an area, such as an airport, it may be possible for users of these different networks to join together to form large, cooperative MANET.

It is a natural evolution for MANETs and P2P networks to be combined together so that a P2P overlay runs on a cooperative MANET (we use the term P2P–MANET to refer to such networks), since both types of networks have many similarities. Both are decentralized networks, both must dynamically organize themselves, both must deal with frequent topology changes, both attempt to be resilient to failure, and both perform the routing function.

Despite the similarities between P2P networks and MANETs, it is unclear that simply adopting existing P2P overlay techniques and using them in MANETs is desirable, since there are also differences. P2P networks tend to be very large–scale with millions of simultaneous users, and are designed as overlays for deployment on the "edge" of the Internet, where the nodes generally do not move about. On the other hand, MANETs tend to have far fewer nodes, the devices are severely resource–constrained in comparison, and the links between nodes usually have higher delay. Energy consumption is of great concern and users are also geographically nearby one another.

Figure 1 illustrates an example of a peer–to–peer overlay on a MANET. The light circles represent nodes participating in the overlay network, while the dark circles are not part of the overlay. The solid lines show how the overlay network is connected, with potentially multiple hops of the underlying MANET providing the direct overlay connections. The dashed lines represent the MANET links.

We expect that in the future, as wireless networks with mobile ad hoc components such as mesh and 4G networks begin to see large increases in the number of users, those users will continue to show a preference for sharing information in the form of P2P networks. However, for the most part, existing

978-1-4244-7265-9/10/$26.00 ©2010 IEEE

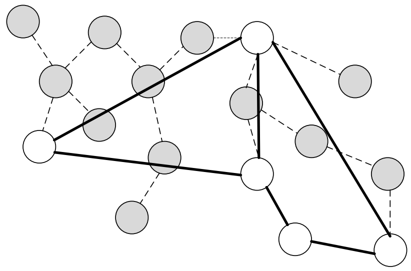


Fig. 1: An example of a peer–to–peer overlay running on a MANET

P2P algorithms cannot be directly applied to MANETs due to the lack of infrastructure, node mobility and energy issues.

This paper studies a game–theoretic topology control algorithm for P2P–MANETs, inspired by [1]. In our non–cooperative game, each node chooses which links to form in the overlay, based on the energy of the potential neighbour, the cost of maintaining links, and the total resulting distance to all other nodes. We discuss both the minimum cost topology and the Nash equilibrium topology. We show that topologies formed by the game are resilient due to their highly interconnected nature, and that the Nash equilibrium is often unable to be determined.

The rest of the paper is organized as follows. Section II provides some background on P2P overlay topologies. Section III discusses the P2P–MANET creation game in detail. In Section IV we analyze the topologies formed by the game. Section V provides some conclusions and directions for future work.

II. Background

P2P overlays may be separated into two classes: *structured* and *unstructured*. In a structured overlay, the topology is strictly controlled and data are kept with certain, specific peers, allowing for more efficient search and routing. Structured overlays use a Distributed Hash Table (DHT) which allows the location of data objects to be efficiently and deterministically found. Unstructured overlays do not implement a DHT or even require any specific topology. Peers join without knowledge of the topology, using relatively loose rules of connectivity.

Structured overlay peers hash data objects, which provides a key that is used to later search for and locate the object. They maintain a routing table that consists of their neighbours' NodeIDs and IP addresses. Search queries are forwarded to neighbours with NodeIDs that are "closer" to the key in the identifier space, with the definition of distance and the details of routing depending on the specific structured algorithm being used. DHT–based algorithms guarantee an upper bound on the number of overlay hops for lookup and routing peers, usually $O(\log N)$ for N peers. A downside to DHTs is that slight differences in data will generate different key values. For example, if there exist multiple copies of the same file with slightly different names, these copies would be mapped differently if the hash takes place over the file name. Also, keyword searching, wildcard, and boolean search operations cannot supported in DHTs.

There are many different structured overlays proposed, such as Chord [2], Pastry [3], and CAN [4]. Due to the manner in which structured overlay algorithms construct the topology, the overlay network may not be efficiently constructed in terms of the number of underlay hops, resulting in highly variable performance. In addition, the tightly controlled topologies of structured overlays are typically undesirable in MANETs because node mobility would hamper their performance.

Many different unstructured overlays exist, such as Gnutella [5] and FastTrack [6]. In unstructured overlays, search queries are flooded to peers, generally with a limited scope to prevent overloading the network. Flooding–based techniques work well in locating readily available objects and are resilient to churn, however they may not locate rare objects that exist in the network. They also face difficulty scaling, as is typical of flooding-based techniques. Despite these disadvantages, unstructured overlays are more commonly used today. They also provide support for keyword, wildcard, and boolean search operations. Hora *et al.* [7] have found that in MANETs, unstructured protocols are more resilient than structured ones at the cost of higher energy and delay.

In constructing an overlay topology for P2P–MANETs, it is important to consider the topology of the underlying physical links. Liu [8] showed that in an unstructured overlay, more than 70 percent of the links do not reflect the underlay topology due to the selection of random neighbours. Tamura *et al.* [9] show that even in wired networks, end–to–end file transfer delay is smaller when the overlay and underlay topologies are the same. Unnecessary energy consumption will also occur when MANET nodes are involved in routing overlay traffic due to a mismatch in the overlay and underlay topologies.

Most attempts to combine P2P networks and MANETs have used cross–layer techniques to merge existing structured algorithms with MANET routing protocols. They focus on improving the lookup and routing performance in MANETs by exploiting P2P DHT techniques. In doing so, they often force a specific routing protocol and structured overlay topology to be used, they focus on node mobility only tangentially, and they often ignore consideration of energy consumption.

Delmastro [10] examined the performance of a Pastry–like algorithm over a MANET using the AODV routing protocol and determined that the performance was poor due to the heavy overhead needed to maintain the large number of connections. It was recommended that the overlay and underlay networks should more closely correspond in order to improve performance.

Cramer and Fuhrmann [11] simulated the performance of Chord in a MANET and found that it too performed poorly. They determined that this occurred not because of the overhead of maintaining the DHT, but because of its pessimistic timeout and failover strategy. When a node moves or a packet is lost, Chord assumes that the node has left the network and removes its state information from the overlay, leading to incorrect behaviour for lookups.

A game–theoretic approach to overlay topology construction was first proposed by Fabrikant *et al.* [1]. The advantage of this approach is that selfish node–agents, working in their own self–interest, and without any centralized design or coordination, are able to determine which links to connect between themselves. The networks are intended for peering agreements between autonomous systems on the Internet, not as P2P overlay networks, but the approach is general enough to have inspired our overlay topology control algorithm, as discussed in Section III. We avoid the cross–layer approach in favour of maintaining the separation of the layers. This makes the topology control algorithm independent of the routing protocol.

The network creation game, as it is dubbed, requires nodes to pay for the links that they establish to others while also considering the distance to all destinations. The cost to establish a link is constant, the distance is the hop count, and agents try to minimize their total cost. A single parameter, α is used as a measure of the tradeoff between establishing more links and decreasing the distance to other nodes. Therefore, it is in an agent's interest to have fewer direct neighbours, while keeping the entire network as close in distance as possible. The authors show that determining the Nash equilibrium, the stable topology, is an NP–hard problem. Fabrikant *et al.* also discuss the "price of anarchy", a measure of the difference between the socially optimal topology and the topology attained when agents act in their own self–interest, and provide upper and lower bounds for it, based on their game.

Chun *et al.* [12] extend the work of Fabrikant *et al.* by simulating the network creation game under various network sizes and configurations. They show that without constraining the node degree, the resulting topologies are star configurations, in which a small number of nodes will maintain a large number of connections. This allows other nodes to "free ride" off of those nodes by connecting to them and achieving both a small number of connections and also a low distance. This results in less resilient topologies because a small number of node failures will result in a large degree of disconnection. Therefore, the authors conclude that there is a fundamental tradeoff between performance and resilience.

Moscibroda *et al.* [13] also extend the work of Fabrikant *et al.*, examining the effect of the network creation game on P2P topologies. The latency between nodes is used as the distance metric, and the authors provide an upper and lower bound on the price of anarchy. They go on to prove that there exist metric spaces for which no Nash equilibrium exists, meaning that even in the absence of peer churn, the topology will never stabilize. Our results appear to confirm this for P2P–MANET networks as well.

III. P2P–MANET Overlay Topology Control

Once a group of MANET nodes choose to form a P2P network, the problem of organizing them in terms of a topology arises. Nodes are "closer" to some nodes than others due to the underlying physical network. Determining how peers should form logical network connections with one another is the basis of the overlay topology problem.

The benefit of selecting neighbours with greater energy remaining, those that are likely to be longer–lived, is that it provides a more stable topology. With longer–lived neighbours, peers will need to seek out new neighbours less frequently.

A topology with the set of links that gives a closer distance from a given peer to all others is preferred because it means that overlay traffic will pass over fewer links to reach its destination. This means faster delivery time and less energy consumption due to fewer intermediate MANET nodes being required to forward packets.

The problem of finding an optimal topology for the subset of nodes in a MANET that are participating in the P2P overlay can be formulated in game–theoretic fashion using a network creation game. The model assumes that peers are selfish and seek to minimize their own costs. This model is appropriate for MANETs because it does not require central coordination. An exhaustive search of the solution space allows us to determine the minimum cost topology, though this solution may not be a Nash equilibrium. The Nash equilibrium of the game produces a solution that is stable; one from which no peer wishes to deviate. Finding both the minimum cost solution and the Nash equilibrium of the network creation game have been shown to be NP–hard [1].

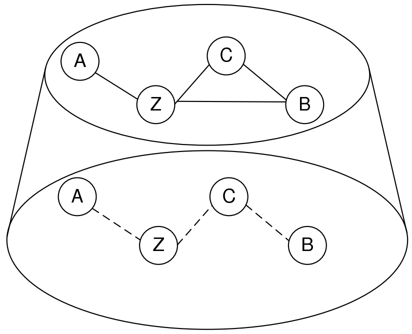
A. System Model

We follow a game–theoretic model of network creation, similar to that first proposed by Fabrikant *et al.* [1]. The game players are the overlay members, and their strategy choices create an undirected graph that represents the overlay topology. The strategy of each peer indicates which edges connect it to other peers. The union of all of these sets of edges produces the resulting undirected graph.

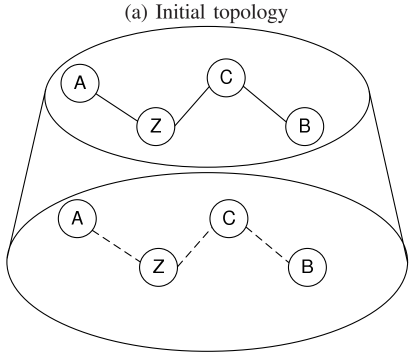
Clearly, the highest connectivity level possible results from a fully connected mesh network, one in which each peer is connected to every other peer. In this case, the overlay distance between peers is always one, and the underlay distance is the smallest number of physical hops between them. However, this topology results in very high overhead due to the cost of maintaining these links, and has a link complexity of $O(n^2)$.

Having fewer neighbours reduces the number of links that must be maintained and the cost that goes along with it, but increases the distance between peers. There must be a tradeoff between distance and link overhead costs. There are many definitions of distance that may be used including hop distance, delay, bandwidth, etc. In this paper we adopt a notion of distance commonly used in P2P networks, referred to as the *stretch*. We define the stretch as the ratio between the total number of physical hops traversed and the shortest physical distance possible. This definition is used because in MANETs all traffic must ultimately be sent via the wireless links, in hop by hop fashion. Therefore, shorter physical hop distances are preferred since this reduces latency and energy consumption.

The network creation game has n players, $P = \{p_0, p_1, \ldots, p_{n-1}\}$, each being a member of the P2P overlay. The strategy space of player p_i is the set $S_i = 2^{P \setminus p_i}$, representing all possible edges to all possible peers, excluding p_i itself. The strategy chosen by a peer, $s_i \in S_i$, is the set of links to be established with other peers. If $p_j \in s_i$ then peer i is establishing a link to peer j in the strategy s_i. Since the link is undirected, peer j also establishes a link to peer i. Otherwise, no link is established between



(a) Initial topology



(b) After topology change

Fig. 2: An example of a topology change

i and j in the graph. The combination of the strategies $s = (s_0, s_1, \ldots, s_{n-1}) \in S_0 \times \cdots \times S_{n-1}$ form the undirected graph $G[s] = (P, \bigcup_{i=0}^{n-1}(p_i \times s_i))$.

We define $s^{-p_i,p_j} = (s_0, s_1, \ldots, s_i \setminus p_j, \ldots, s_j \setminus p_i, \ldots, s_{n-1}) \in S_0 \times \cdots \times S_{n-1}$. This is the strategy s with the link between p_i and p_j removed. We also define $s^{+p_i,p_j} = (s_0, s_1, \ldots, s_i \cup p_j, \ldots, s_j \cup p_i, \ldots, s_{n-1}) \in S_0 \times \cdots \times S_{n-1}$. This is the strategy s with a link between p_i and p_j added.

Figure 2 shows how the overlay topology might change in an effort to reduce the cost. In Figure 2a, peer Z is connected to peers A, B, and C at the overlay level. A and C are one hop away in the underlay, and B is two hops away. In Figure 2b Z has dropped the connection to B and is now only connected to A and C. Z now avoids the extra cost of maintaining the link to B, but relies on C to reach peer B. Since it needs to use C to reach B in the underlay network in any case, Z has not increased the distance between itself and B, but has eliminated the overhead of maintaining that link.

B. A P2P–MANET Creation Game

We now propose a P2P–MANET creation game, based on the network creation game first proposed by Fabrikant *et al.* [1].

The cost to each peer has two components. The first component incorporates the energy level of the neighbouring peer, since it is preferable to connect to peers which have longer lifetimes. The second component is the sum of distances from the peer to all others, given that the strategy being evaluated is in place. A parameter, α, is used to capture the tradeoff between establishing a link and the change in distances to peers caused by the establishment of that link. Specifically, the cost to peer i of strategy s is

$$C_i(s) = \alpha \sum_{j \in N_i} e_j + \sum_{j=0}^{n-1} d_{G[s]}(i,j) \quad (1)$$

where N_i is the set of neighbours of peer i, e_j is the energy level of peer j, and $d_{G[s]}(i,j)$ is the stretch from peer i to peer j in graph $G[s]$.

The total cost is the sum of all peers' costs

$$C(G[s]) = \sum_{i=0}^{n-1} C_i(s) \quad (2)$$

Given Equation 2, it is possible to find the lowest possible total cost, which results in the optimal minimum cost topology. However, this solution may not be stable. Each peer is attempting to maximize his or her own payoff, which in this case is given by Equation 1. This means that if the global minimum cost is not a Nash equilibrium, peers will have an incentive to change their strategy in order to increase their payoff. The resulting Nash equilibrium, if it exists, is a stable topology because no peer will have an incentive to deviate from the set of links it produces. However, the total cost of the topology may exceed the global minimum.

A pure Nash equilibrium of this game is a strategy s such that for each player i and for all s' that differ from s only in the ith component, $c_i(s') \geq c_i(s)$. That is, no player has an incentive to unilaterally deviate from its selected strategy since this would increase its cost. This strategy is known as the best response. The difference between the "socially optimal" or minimum cost solution and the Nash equilibrium is referred to as the *price of anarchy* [1].

It has been shown in Fabrikant *et al.* that computing the best response of peer i is NP–hard [1]. This is because p_i must pick a subset of vertices from the entire graph, which is a reduction from the dominating set problem. Computing the global minimum cost is also NP–hard, for the same reason.

In an undirected overlay of n peers, there are $\frac{n(n-1)}{2}$ possible links and therefore, to find the global minimum cost, an exhaustive search must examine $2^{\frac{n^2-n}{2}}$ possible link combinations. To determine the Nash equilibrium, for each peer at each step, 2^{n-1} strategies must be examined, since for a given peer, the link to each of the other peers must be considered. These strategy spaces are exponential in size, and require exponential time complexity. Therefore, it is only practical to find the optimal solution for relatively small networks. For larger networks, the exponential time complexity strategy search must be replaced with other techniques such as random local searches.

As in Chun *et al.* [12], we make use of a random local search to examine the solution space. We adopt this strategy to find both the global minimum cost and the Nash equilibrium. It is important to note that the search may converge to topologies that are not Nash equilibria. Furthermore, there is no guarantee that a Nash equilibrium even exists.

Our random local search algorithm to find the minimum cost operates as follows and is shown in Algorithm 1. We start from a random topology and note its cost. Next, a random peer is selected, and its connection to each of the other peers is examined in turn. If the connection doesn't exist, the total cost of the entire topology is determined as if it were connected. If this results in a decrease in the total cost, then the link is established. If the connection does exist, the total cost of the entire topology is determined as if it were not connected. If this results in a decrease in the total cost, then the link is disconnected. When the change in cost results in only a small difference, ϵ, a new random peer is selected and the process is started anew. This continues until the maximum time allowed, T_{max}, has elapsed. When the algorithm terminates, the graph $G[s]$ is the overlay topology of the lowest cost graph found, and its cost is $C(G[s])$.

Algorithm 1 Random local search to find the minimum cost

$G[s] \leftarrow$ random topology
$C_{min} \leftarrow C(G[s])$
while $t < T_{max}$ **do**
  select random peer, p_i
  for each $p_j \neq p_i$ **do**
    if $p_j \in N_i$ **then**
      if $|C_{min} - C(G[s^{-p_i,p_j}])| < \epsilon$ **then**
        break
      else if $C(G[s^{-p_i,p_j}]) < C_{min}$ **then**
        $N_i \leftarrow N_i \setminus p_j$
        $C_{min} \leftarrow C(G[s^{-p_i,p_j}])$
      end if
    else
      if $|C_{min} - C(G[s^{+p_i,p_j}])| < \epsilon$ **then**
        break
      else if $C(G[s^{+p_i,p_j}]) < C_{min}$ **then**
        $N_i \leftarrow N_i \cup p_j$
        $C_{min} \leftarrow C(G[s^{+p_i,p_j}])$
      end if
    end if
  end for
end while

The random local search algorithm for determining the Nash equilibrium is similar, and is given in Algorithm 2. Again, an initial, random topology is created as a starting point. Every neighbour of every peer in the network is now examined in turn. For a given peer, p_i, the effect of disconnecting each established link on the cost to p_i, and p_i alone, is observed. If p_i's cost decreases, the link is severed. Next, a random peer $p_k \notin N_i$ is selected. p_i then determines its cost if it were to establish a connection to p_k. If the cost decreases, the link is established. This process is repeated for every peer in the overlay. When no peer changes any of its links, the resulting topology is considered a Nash equilibrium. Note that this topology may not be a true Nash equilibrium. The other way the algorithm may terminate is if the maximum time, T_{max} has been exceeded.

We now discuss the complexity of determining the cost of the topology of n peers. To determine the cost for a particular peer p_i, each of the remaining $n-1$ destinations must be examined. Determining the energy of the peer requires a simple lookup. However, determining the stretch of the resulting graph requires an examination of the overlay hop distance to each of the $n-1$ possible destinations. For each destination p_j, the minimum overlay distance must be determined. This requires that each possible path from p_i to p_j be examined using the remaining $n-2$ peers as a potential first hop. Therefore, determining the total stretch of the resulting distance graph requires $O(n^3)$ time. In the case of the Nash equilibrium, each peer can calculate the distance in parallel, so the time complexity is $O(n^2)$ for each peer. The stretch must be calculated for each potential link addition or deletion in Algorithms 1 and 2.

Algorithm 2 Random local search to find the Nash equilibrium

$G[s] \leftarrow$ random topology
$C_i^{Nash} \leftarrow C_i(s)$
while $t < T_{max}$ **do**
  $unchanged \leftarrow true$
  for each p_i **do**
    for each $p_j \neq p_i$ **do**
      if $p_j \in N_i$ **then**
        if $C_i(s^{-p_i,p_j}) < C_i^{Nash}$ **then**
          $N_i \leftarrow N_i \setminus p_j$
          $C_i^{Nash} \leftarrow C_i(s^{-p_i,p_j})$
          $unchanged \leftarrow false$
        end if
      end if
    end for
    select a random $p_k \notin N_i$
    if $C_i(s^{+p_i,p_k}) < C_i^{Nash}$ **then**
      $N_i \leftarrow N_i \cup p_k$
      $C_i^{Nash} \leftarrow C_i(s^{+p_i,p_k})$
      $unchanged \leftarrow false$
    end if
  end for
  if $unchanged = true$ **then**
    break
  end if
end while

IV. Analysis of Topologies

The P2P–MANET creation game was evaluated in a solver written in the C++ programming language. In our simulations, we use 100 MANET nodes, with the number of nodes participating in a P2P overlay varying from 50 to 100 in increments of 10. The network area is 1500 m $\times$ 1500 m and the transmission rate is 54 Mbps. The two ray ground radio propagation model along with an omnidirectional antenna are used by all nodes. The random waypoint mobility model is used, with all nodes evenly distributed in the simulation area. Nodal velocities are distributed according to a uniform distribution, with a minimum 1 m/s and a maximum 3 m/s, and a uniformly distributed pause time with mean 60 s.

Constant bit rate (CBR) traffic of 1000 byte packets sent 100 times per second is sent between members of the P2P overlay. The traffic starts and stops at random times, between random peers. This creates overlay traffic, which is used to change peers' energy levels.

TABLE I: Energy consumption constants used in simulations

m_{send}	1.89	$\mu W \cdot sec/byte$
b_{send}	246	$\mu W \cdot sec$
m_{recv}	0.494	$\mu W \cdot sec/byte$
b_{recv}	56.1	$\mu W \cdot sec$
$b_{sendctl}$	120	$\mu W \cdot sec$
$b_{recvctl}$	29.0	$\mu W \cdot sec$

The energy consumption model used is the linear model proposed by Feeney [14]. Each MAC layer operation takes a certain amount of power as defined by $cost = m \times size + b$ where m is the incremental cost of the operation, b is the fixed cost, and $size$ is the amount of data sent or received. The constants are obtained by physical measurements for a Lucent IEEE 802.11 WaveLAN PC Card from [14] and are summarized in Table I.

The MANETs were run for two simulation hours, and the underlay network topology was taken at thirty minute intervals, and input to the P2P–MANET creation game solver. The results obtained in all experiments are based on 10 independent runs and have a 95% confidence level, which is indicated via error bars. Though overlay sizes of 50–100 nodes were tested, due to space constraints, only the figures for 60 and 100 overlay peers are given below. These were chosen because they provided the most Nash equilibria.

Obtaining the actual minimum cost for an overlay by performing an exhaustive search of the solution space is infeasible for the network sizes examined here, 50–100 overlay nodes. We therefore perform a random local search of the solution space using Algorithms 1 and 2. An initial random point of the solution space is examined, and we proceed to walk to progressively better results in the area as discussed in Section III-B. When the difference between successive results is less than 1%, we jump to a new random point of the solution space and begin walking again. The results of the Nash equilibrium are presented where they could be determined since a Nash equilibrium could not be found in all experiments. The minimum cost and Nash equilibria are not constrained by a maximum degree in our experiments.

The α parameter is set to the values 1, 10, 100, and 1000 to determine its effects. The P2P–MANET creation game software was run for a maximum period of 24 hours for each experiment to determine the lowest cost, and another maximum period of 24 hours to determine the Nash equilibrium.

A. Simulation Analysis

In more than half the experiments shown, no Nash equilibria could be found within the time constraints. Even in those scenarios where Nash equilibria were found, no more than 6 out of the 10 independent test runs for that particular experiment were successful in determining the equilibrium value. This indicates that finding the Nash equilibrium is difficult at best, and that it may not even exist in most cases. If no Nash equilibrium exists, then the peers will not converge to a stable topology since peers will always have an incentive to change their neighbours. In a P2P–MANET, where peers are both mobile and churning, it poses an even greater problem.

TABLE II: The price of anarchy

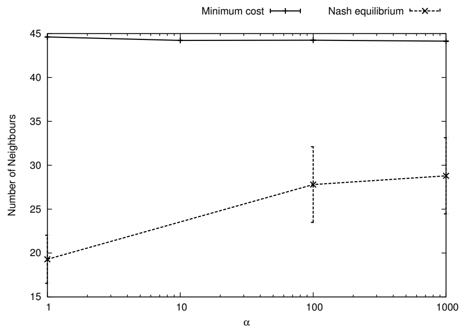
$n = 60, \alpha = 1$	1.35
$n = 60, \alpha = 100$	1.29
$n = 60, \alpha = 1000$	1.08
$n = 70, \alpha = 1$	1.88
$n = 70, \alpha = 100$	1.31
$n = 80, \alpha = 1$	1.64
$n = 80, \alpha = 100$	1.62
$n = 90, \alpha = 1$	1.47
$n = 100, \alpha = 1$	1.37
$n = 100, \alpha = 10$	1.58
$n = 100, \alpha = 100$	1.63

Table II provides a value for the price of anarchy for those cases where a Nash equilibrium could be found. The price of anarchy is defined as the ratio of the Nash equilibrium cost over the minimum cost solution. The price was found to be between 1 and 2, indicating that the stable topology may be as much as twice the cost of the minimum cost one, with no discernable trend. There is no general algorithm that finds Nash equilibria, so a random search of the solution space, as was done in this paper, will produce unpredictable results, as seen both by the results in the table and also those that do not appear since they were not found.

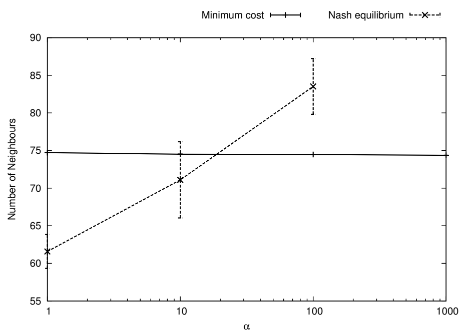
Figure 3 shows how many neighbours each peer has on average. The minimum cost topologies tend to maintain a very high number of neighbours, which doesn't change much despite increasing values of α. The Nash equilibrium topologies available show that it also maintains a high number of neighbours, which actually increases as it becomes more costly to maintain neighbours. This indicates that the peers are penalized for being selfish, because it would be expected that they would maintain fewer neighbours. But only a small number of them can do so before the others are forced to compensate by increasing their degree, resulting in a higher average degree. The Nash equilibria also have larger confidence intervals due to the more volatile nature of the equilibria.

Figure 4 shows the total stretch values obtained for the networks. As α increases, the cost of maintaining more links increases. As seen earlier, this has no effect on the minimum cost topologies, and hence their stretch values do not change. In fact, the stretch of the minimum cost is very close to the minimum possible stretch. This results in a very highly interconnected network, in which most peer–pairs are connected. For the Nash equilibrium topologies, it can be seen that the stretch increases with increasing α. This behaviour would be expected if the number of neighbours decreased, but as seen previously, the Nash equilibrium topologies tend to keep more neighbours with increasing α values. This indicates that the quality of the selected neighbours is not as good as the minimum cost topologies' neighbours.

Figure 5 provides the average underlay hop distance a peer's neighbours are from it. Lower values mean that the peer has selected closer nodes, which indicates that the overlay is similar in topology to the underlay. As α increases, the cost of maintaining links increases. The minimum cost toplogies connect to nearly all peers, resulting in higher overall neighbour distances, for all overlay sizes and values of α. Once again, the values do not change with increase α. The Nash equilibrium

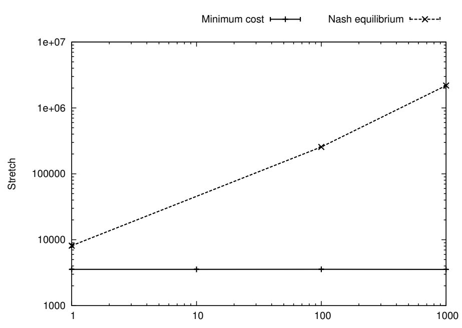


(a) 60 peers

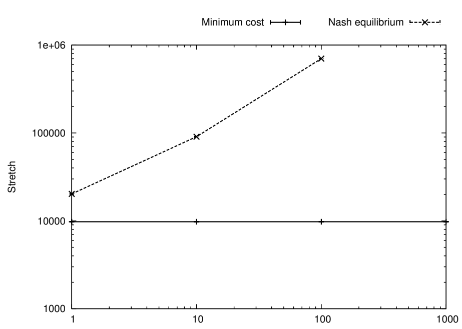


(b) 100 peers

Fig. 3: Average number of neighbours

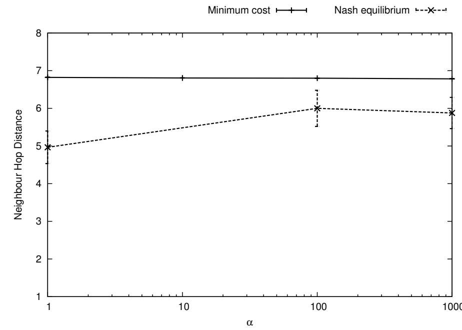


(a) 60 peers

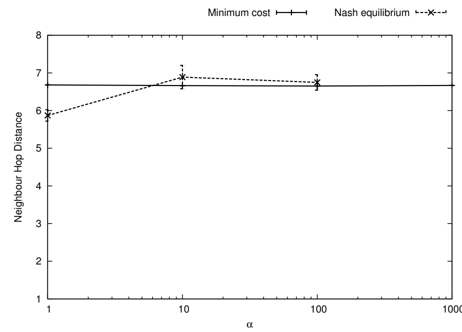


(b) 100 peers

Fig. 4: Total stretch



(a) 60 peers



(b) 100 peers

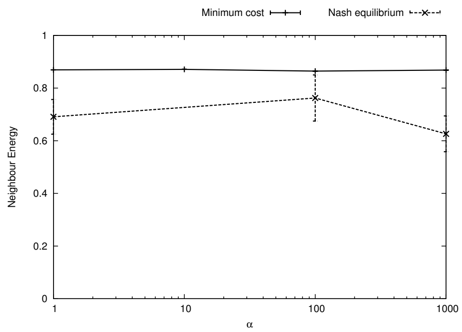
Fig. 5: Average neighbour hop distance

topologies show that, unsurprisingly, as more neighbours are gained, the distance to them increases.

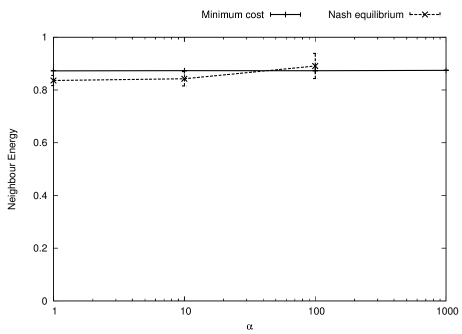
Figure 6 provides the average normalized energy consumed for each peer's neighbours. Lower values mean that the peer is connected to peers with more energy remaining. As α increases, the minimum cost topologies does not change, while the Nash equilibrium topologies' random behaviour prevents any observable trend. We can determine that in the Nash equilibrium, peers usually choose longer–lived peers than the minimum cost topology, mainly because nearly all node–pairs are connected in the minimum cost topology.

The next three figures provide the K metric values. These are a measure of the topology's resilience and are adapted from [15]. The K is the ratio of all connected peer–pairs in the network divided by the total number of distinct peer–pairs in the network. We refer to this value, calculated for the overlay topology selected, as K $metric_{normal}$. We call K $metric_{random-failure}$ the K when 10% of peers are randomly selected and removed from the topology. This measures the effect of a failure of a random 10% of the peers, due to effects such as peer mobility, voluntarily leaving the overlay, or running out of energy. Finally, we call K $metric_{popular-failure}$ the K when the 10% of the peers with the highest degree are removed. This measures the effect of excessive traffic on the most popular peers, which may result in their running out of energy, or their leaving the overlay to prevent a large drain on their battery.

Figure 7 shows the K $metric_{normal}$ for the topologies that were determined. A value of 1 means that each peer–pair is connected, as would be the case in a fully connected mesh. We see that the minimum cost topologies have a very



(a) 60 peers



(b) 100 peers

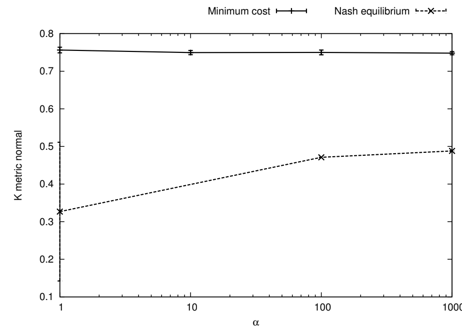
Fig. 6: Average neighbour energy

high K metric because such a large share of the peers are interconnected. The Nash equilibria have mostly lower K values, though it increases because peers maintain more links as α increases.

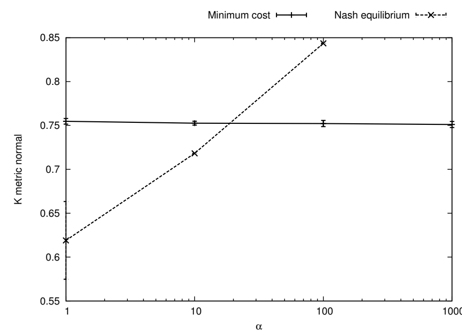
Figure 8 shows the $K\ metric_{random-failure}$. Here, the minimum cost topologies show a reduced K metric, but as is expected, the value is still very high due to the very high level of connectedness of the topology. The Nash equilibria also exhibit similar behaviour, but in the case of 100 peers, the Nash equilibrium topologies are more resilient than the minimum cost topology at a lower value of α, as evidenced by the lower value of α at which the Nash equilibrium topology's K exceeds that of the minimum cost topology.

Figure 9 shows the $K\ metric_{popular-failure}$. Once again, the minimum cost topologies perform very well because since nearly all peer–pairs are connected, the effect of removing the most connected 10% is minimal. Furthermore, these values are all unchanging with α. The Nash equilibria, as expected from previous results, show an increasing K metric with increasing α since more peers become connected and once again, this results in a K metric that is sometimes higher than the minimum cost topologies.

These figures show that the minimum cost topologies are highly resilient, but that this comes due to a large degree of interconnection. The Nash equilibria are also highly resilient, and surprisingly, the resilience increases as the cost of maintining links increases. This runs counter to expectations, and results because peers must keep more neighbours with increasing α as a result of the selfish nature of peers' behaviour. A small number of peers are able to maintain fewer connections, but to compensate for this, most peers must keep more links.

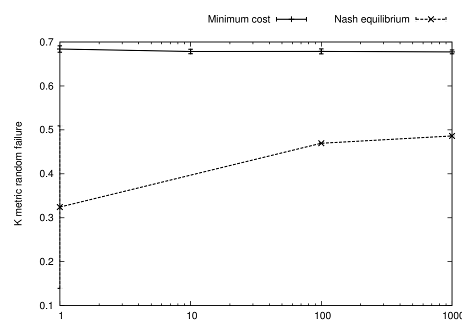


(a) 60 peers

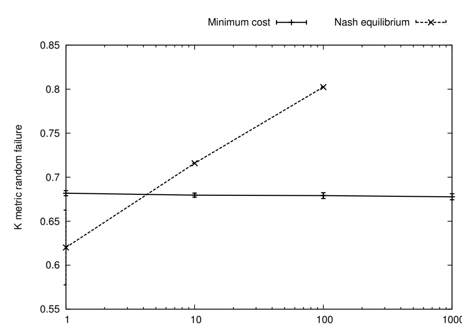


(b) 100 peers

Fig. 7: K metric normal

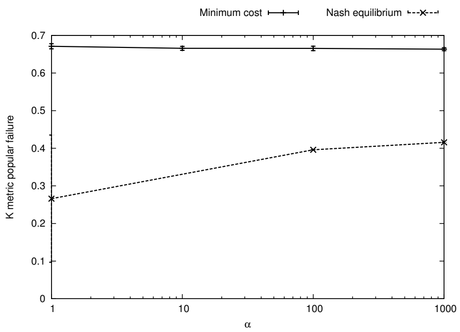


(a) 60 peers

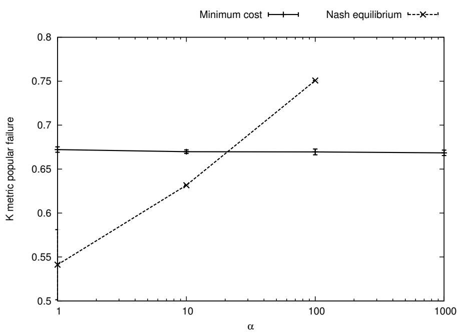


(b) 100 peers

Fig. 8: K metric random failure

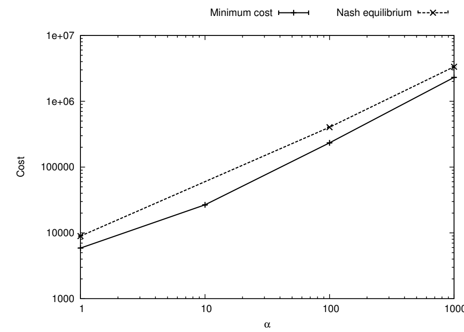


(a) 60 peers

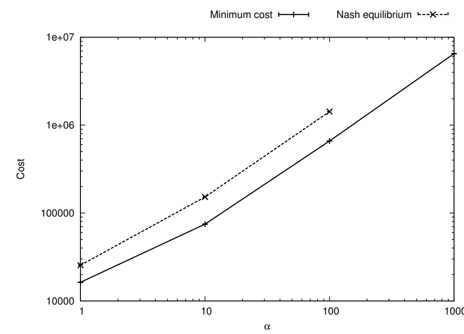


(b) 100 peers

Fig. 9: K metric popular failure



(a) 60 peers



(b) 100 peers

Fig. 10: Total cost

This results in a larger overall cost for the topologies, as seen in Figure 10, where the total cost, $C(G[s])$ for various values of α are shown.

V. Conclusions

This paper studied the topologies that resulted from a game–theoretic overlay topology control algorithm for P2P–MANETs. The results show that the created topologies were highly interconnected, resulting in very resilient networks. However, Nash equilibrium topologies increased the number of neighbours as the cost of keeping neighbours grew, resulting in increasing costs. Furthermore, Nash equilibrium topologies were not found more frequently than they were, indicating that a stable topology is unlikely in most cases. The P2P–MANET game requires *a priori* knowledge of all peers, their energy levels, and their overlay and underlay distances. With this information, it determines the entire overlay topology all at once. This means that if a single peers joins or leaves the overlay, the entire computation must be performed again. In a dynamic MANET and P2P overlay, it is expected that peers will be constantly joining and leaving, in addition to being mobile. Therefore, in the future, we will devise a heuristic algorithm that does not require complete information and builds the topology in steps, as peers join and leave.

References

[1] A. Fabrikant, A. Luthra, E. Maneva, C. H. Papadimitriou, and S. Shenker, "On a network creation game," in *ACM Symposium on Principles of Distributed Computing*, 2003, pp. 347–351.

[2] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup protocol for internet applications," in *ACM SIGCOMM*, September 2001, pp. 149–160.

[3] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems," in *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, November 2001, pp. 329–350.

[4] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A scalable content-addressable network," in *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, 2001.

[5] Gnutella, "The annotated gnutella protocol specification v0.4," http://rfc–gnutella.sourceforge.net/developer/stable/index.html.

[6] H. Balakrishnan, M. F. Kaashoek, D. Karger, R. Morris, and I. Stoica, "Looking up data in P2P systems," *Communications of the ACM*, vol. 46, no. 2, pp. 43–48, 2003.

[7] D. N. da Hora, D. F. Macedo, L. B. Oliveira, I. G. Siqueira, A. A. Loureiro, J. M. Nogueira, and G. Pujolle, "Enhancing peer-to-peer content discovery techniques over mobile ad hoc networks," *Computer Communications*, vol. 32, pp. 1445–1459, August 2009.

[8] Y. Liu, "A two–hop solution to solving topology mismatch," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 11, pp. 1591–1600, November 2008.

[9] Y. Tamura, S. Kasahara, Y. Takahashi, S. Kamei, and R. Kawahara, "Inconsistency of logical and physical topologies for overlay networks and its effect on file transfer delay," *Performance Evaluation*, vol. 65, no. 10, pp. 725–741, October 2008.

[10] F. Delmastro, "From Pastry to CrossROAD: CROSS-layer Ring Overlay for AD hoc networks," in *Pervasive Computing and Communications Workshops (PerCom)*, 2005, pp. 60–64.

[11] C. Cramer and T. Fuhrmann, "Performance evaluation of chord in mobile ad hoc networks," in *MobiShare*, September 2006.

[12] B.-G. Chun, R. Fonseca, I. Stoica, and J. Kubiatowicz, "Characterizing selfishly constructed overlay routing networks," in *IEEE INFOCOM*, vol. 2, 2004, pp. 1329–1339.

[13] T. Moscibroda, S. Schmid, and R. Wattenhofer, "On the topologies formed by selsh peers," in *5th International Workshop on Peer–to–Peer Systems*, February 2006.

[14] L. M. Feeney, "An energy–consumption model for performance analysis of routing protocols for mobile ad hoc networks," *Mobile Networks and Applications*, vol. 6, no. 3, pp. 239–250, 2001.

[15] S.-T. Park, A. Khrabrov, D. Pennock, S. Lawrence, C. Giles, and L. Ungar, "Static and dynamic analysis of the internet's susceptibility to faults and attacks," in *IEEE INFOCOM*, vol. 3, 2003, pp. 2144–2154.