An Efficient Anonymous Communication Protocol for Peer-to-Peer Applications over Mobile Ad-hoc Networks

Chao-Chin Chou, *Student Member, IEEE,* David S. L. Wei, *Member, IEEE,* C.-C. Jay Kuo, *Fellow, IEEE,* and Kshirasagar Naik, *Member, IEEE*

***Abstract*— An efficient anonymous communication protocol, called MANET Anonymous Peer-to-peer Communication Protocol (MAPCP), for P2P applications over mobile ad-hoc networks (MANETs) is proposed in this work. MAPCP employs broadcasts with probabilistic-based flooding control to establish multiple anonymous paths between communication peers. It requires no hop-by-hop encryption/decryption along anonymous paths and, hence, demands lower computational complexity and power consumption than those MANET anonymous routing protocols. Since MAPCP builds multiple paths to multiple peers within a single query phase without using an extra route discovery process, it is more efficient in P2P applications. Through analysis and extensive simulations, we demonstrate that MAPCP always maintains a higher degree of anonymity than a MANET anonymous single-path routing protocol in a hostile environment. Simulation results also show that MAPCP is resilient to passive attacks.**

***Index Terms*— peer-to-peer, P2P, anonymity, MANET.**

I. Introduction

THE Peer-to-peer (P2P) network has drawn increasing attention nowadays, and has been widely deployed on the Internet for various purposes, including distributed data storages, file sharing networks, collaborative computing and Internet telephony. The P2P system is popular for its being scalable, fault-tolerant, and self-organized. Meanwhile, mobile ad-hoc networks (MANETs) have been proposed as an alternative to cellular networks for use in areas where fixed infrastructures such as base stations are unavailable. MANET resembles the P2P network in some ways. First, both systems lack fixed infrastructure and network topology. The P2P peers join and leave frequently and unpredictably, while MANET nodes move randomly. Second, both systems require no centralized coordinator for communication. Instead, they both require the cooperation of network nodes for communication. MANET is now emerging as a new paradigm of wireless communication for civilian applications. Nowadays, numerous portable devices such as laptops, PDAs and mobile phones are everywhere, and people use them for their professional and daily lives. The materialization of wireless technologies has changed the scenario of ad-hoc networking, its usage, its players, as well as its importance. Therefore, MANET appears to be an attractive platform for the P2P applications. In fact, P2P applications on Internet are gradually migrating to MANET [1][2][3][4]. Emerging P2P applications over MANET include (1) sharing multimedia files among mobile hand-held devices, (2) sharing traffic, weather and traveling information among moving vehicles, and (3)sharing real-time information among military units on the battlefield.

Providing peer privacy in the P2P network has always been an important topic, which poses even more challenges when facing a P2P system over MANET. First, the open environment in MANET makes its radio signals vulnerable to eavesdropping. Second, the multihop communication in MANET involves untrustworthy nodes in a private conversation. Third, MANET nodes are constrained by limited battery and computing power, which makes computation-intensive schemes such as the public-key cryptography too expensive to be adopted. Therefore, existing solutions for wireline Internet cannot be applied directly on MANET for P2P communication without considerable modifications. This paper presents the *MANET Anonymous Peer-to-peer Communication Protocol* (MAPCP), which serves as an efficient anonymous communication protocol for P2P applications over MANET. MAPCP is designed to be a flexible middleware between the P2P applications and MANET routing protocols. MAPCP employs a broadcast-based mechanism together with a probabilistic-based flooding control algorithm to establish anonymous paths between peers, which requires no hop-by-hop encryption/decryption, hence requires lower computational complexity and power consumption. MAPCP establishes multiple anonymous paths between communication peers within a single query phase, and is highly resilient to node mobility, failure, and malicious attacks. Furthermore, MAPCP provides schemes for communication peers to control the tradeoff between anonymity degree and bandwidth efficiency.

The rest of the paper is organized as follows. Section II reviews previous work on anonymous communication over Internet and MANET. Section III presents the design rationale of MAPCP and gives protocol description in details. Section IV presents an analysis of anonymity degree achieved by MAPCP. Section V evaluates the performance of MAPCP

Manuscript received January 6, 2006; revised August 9, 2006. This paper was presented in part at GLOBECOM 2006, Washington, DC, USA

Chao-Chin Chou and C.-C. Jay Kuo are with the Viterbi School of Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: ccchou@ieee.org; cckuo@sipi.usc.edu).

D. S. L. Wei is with the Department of Computer and Information Science, Fordham University, Bronx, NY 10458 USA (e-mail: wei@dsm.fordham.edu).

Kshirasagar Naik is with the Dept. of ECE, University of Waterloo, Waterloo, Ontario, Canada (e-mail: knaik@swen.uwaterloo.ca).

Digital Object Identifier 10.1109/JSAC.2007.070119.

through extensive simulations. Finally, a conclusion is drawn in Section VI.

II. Related Work

Existing solutions for wireline Internet provide anonymous communication by means of application-layer routing, *e.g.* Mix-net [5], onion routing [6] and Crowds [7]. They require common secrets among the sender and all the proxies en route, and hop-by-hop decryptions along the routing path, which is not affordable to MANET nodes due to the constraints of limited energy and computing power. Several anonymous routing protocols have also been proposed for MANET, *e.g.* ANODR [8] and MASK [9]. In general, these network-layer solutions consist of two phases: *anonymous route discovery phase* and *anonymous data transmission phase*. In the first phase, the sender broadcasts a *route request message* to discover an anonymous route to its communication target. The entire process usually involves hop-by-hop encryption/decryption to conceal the route information from eavesdroppers. Once the anonymous route is established, the sender enters the anonymous data transmission phase and begins to send data packets via the anonymous route. ANODR [8] is the first identity-free anonymous on-demand MANET routing protocol. It employs the *Trapdoor Boomerang Onion*, a variant of the onion that uses only symmetric key cryptography, to build the anonymous routing path. Its major flaw is being sensitive to node mobilities, and the route information is partially revealed if one or more nodes en route are compromised. MASK [9] employs an anonymous neighborhood authentication protocol to establish its routing path instead of using the onion structure, and is claimed to have lower computational complexity than ANODR. While these anonymous routing protocols achieve good performance in providing privacy for point-to-point unicast communication, there is still too much overhead introduced when applying them to P2P applications over MANET. Most P2P applications involve two phases: *query phase* and *data transmission phase*. In the query phase, the file requester broadcasts its query message to the entire network, and the file holders reply to the requester the metadata of the queried file. When the requester received enough query replies, it establishes a unicast connection to each file holder and proceeds to the data transmission phase. In order to provide privacy in P2P applications, communication in both query phase and data transmission phase should be anonymous. Therefore the routing protocols are supposed to guarantee the anonymity of broadcast queries in the first phase, and then establish an anonymous route between the file requester and the file holder. This means two or more rounds of message broadcasts are required since the construction of anonymous routes also requires broadcast of route discovery messages. The situation is even worse when the requester requests files from multiple file holders simultaneously, which is a common scenario in P2P applications, not to mention the hop-by-hop encryption/decryption overhead for building a single anonymous route.

MAPCP differs from previous work in the following aspects. First, MAPCP is not a routing protocol. It lies in between the network layer and the application layer. It is designed to be a flexible middleware specially for anonymous P2P communication. Applications which do not require anonymity can bypass the MAPCP layer to avoid the overhead brought by anonymity. However, applications will find no way to jump over the anonymous routing protocol if an anonymous routing has been employed at the network layer. Second, MAPCP avoids using expensive hop-by-hop encryption/decryption. Instead, it exploits broadcasts and probabilistic flooding control to provide anonymity, thereby consuming much less computing resources and energy. Third, the anonymous paths between the file requester and all possible file holders are established right after the requester receives query replies. No extra route discovery phase is needed and the data transmissions can be started right after enough query replies are received and, therefore, it greatly reduces the overhead from the anonymous path construction. Fourth, MAPCP establishes multiple paths for each communication pair in a single query-reply round. Building multiple paths has been shown to be able to effectively enhance performance in a mobile environment and mitigate disruption caused by path failure or compromised nodes [10]. For most MANET anonymous routing protocols, building multiple paths for a communication pair usually involves multiple route discovery processes. MASK creates multiple paths by multiplexing the route hop-by-hop. MAPCP differs from MASK in that there is no path selection. Packets in MAPCP are forwarded within all established paths. Finally, MAPCP provides schemes for communication peers to control the tradeoff between anonymity degree and bandwidth efficiency.

III. Protocol Design

A. Design rationale

Hop-by-hop encryption/decryption does provide excellent anonymity and content privacy. However, previous study [11][12] shows that the computational complexity and power consumption of a public-key encryption (*e.g.* RSA) are several orders greater than a symmetric-key encryption (*e.g.* AES) and a packet transmission. Therefore, we argue that cryptography should be used conservatively in MANET in which resources is scarce. The MANET communication usually involves one or multiple local broadcasts, even for unicast communication. As discussed in previous work [8][13], broadcast without specifying receiver's real identity effectively achieves the receiver anonymity and thwarts many security attacks [14]. Therefore, we believe that a good solution for anonymous P2P communication over MANET should deal with the tradeoff between resource efficiency (bandwidth efficiency, energy consumption and computational intensity) and the degree of anonymity. Such a solution should lie somewhere between the pure broadcast scheme and the pure cryptographic scheme, as shown in Fig. 1.

B. Protocol Design

The design of MAPCP assumes that each node is in the promiscuous receiving mode on their wireless network interface (which is mandatory for 802.11-based nodes in the ad-hoc mode) and is capable of manipulating the source IP and MAC address of its outgoing packets. Similar to

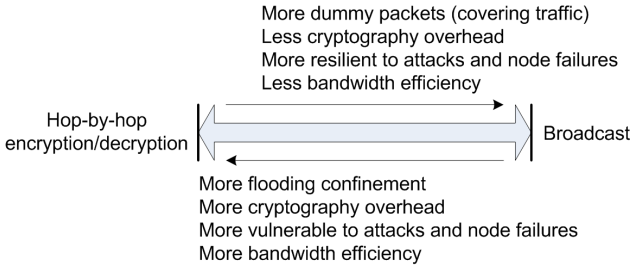


Fig. 1. Tradeoff between hop-by-hop encryption/decryption schemes and broadcast-based schemes

most P2P applications, communication in MAPCP consists of two phases: the query phase and the data transmission phase. MAPCP uses only local broadcasts in both phases. To prevent the broadcast storm problem [15], MAPCP employs a probabilistic algorithm to control packet flooding in the data transmission phase. Conceptually, every node is assigned a rebroadcast probability for each communication session. Nodes along the selected optimal paths are assigned the highest probability while nodes not on the optimal paths are assigned a lower or zero probability. At each node, the forwarding of a data packet depends on the calculated rebroadcast probability. To realize this, each MAPCP node maintains two tables: a destination table of five fields (which include the destination ID pseudonym, the path pseudonym, the δ value, the τ value and the session key) and a path table of four fields (which include the source ID pseudonym, the path pseudonym, the δ value and the τ value).

C. Query Phase

The file requester, S, first generates a one-time public/private key pair PK_S and PK_S^-, a 128-bit random nonce N_S (used as its identity pseudonym) and a random positive integer $\delta = \delta_S > 1$. The overhead of key and pseudonym generation can be traded off by storage since the node can generate a number of keys and pseudonyms in advance. Then, S broadcasts to its neighbors the query message with a forged source address *e.g.* the broadcast address. The query message includes PK_S, N_S, δ and the query string $QString$. This is expressed as

$$S \rightarrow *: \{PK_S, N_S, \delta, QString\}.$$

Besides, S keeps entries $\{null, null, \delta_S, MAX\_INT, null\}$ in its own destination table, where MAX_INT is a very large positive integer.

When node i, $(i \neq S)$, receives a nonduplicate query message, it increases δ by 1 and forwards the query message to its neighbors. Node i checks whether the query can be satisfied. If no, it keeps entries $\{N_S, null, min(\delta), MAX\_INT\}$ in its path table, where $min(\delta)$ is the minimum δ value among all received query messages. Otherwise, if i can satisfy the query (i is a file holder), it generates a random positive number $\tau = \tau_i > 1$, a 128-bit random nonce N_i, another 128-bit random nonce N_i^P, and a one-time symmetric key SK_i. Here, N_i is its identity pseudonym, N_i^P is the path pseudonym and SK_i is the session key for further communication with query originator S. Then, it broadcasts to its neighbors the query reply, which includes N_S, N_i^P, τ, and a PK_S-encrypted part which contains N_i, SK_i and the metadata of the requested file, as shown below:

$$i \rightarrow *: \{N_S, N_i^P, \tau, [N_i, SK_i, metadata]_{PK_S}\}.$$

Note that N_S is used to identify the recipient of this query reply. Node i keeps entries $\{N_S, N_i^P, min(\delta), \tau_i, SK_i\}$ in its destination table.

When node j receives a nonduplicate query reply, it increases τ by 1 and forwards this message to its neighbors. If $j \neq S$, it updates the entry $\{N_S, null, min(\delta), MAX\_INT\}$ in its path table to $\{N_S, N_i^P, min(\delta), min(\tau)\}$, where $min(\tau)$ is the minimum τ value among all received query replies. Otherwise, if $j = S$, it decrypts the encrypted part with PK_S^- to get N_i, SK_i and the metadata, and updates the entry $\{null, null, \delta_S, MAX\_INT\}$ in its destination table to $\{N_i, N_i^P, \delta_S, min(\tau), SK_i\}$.

D. Data Transmission Phase

Once node S collects enough query replies, data transmission between S and each file holder R_i can be done anonymously as follows. S looks up R_i's pseudonym N_{R_i} from its destination table to get $N_{R_i}^P$, δ_S, $min(\tau)$ and session key SK_{R_i} and broadcasts a data message to its neighbors, which contains $N_{R_i}^P$, N_{R_i}, a positive number $\alpha = \delta_S + min(\tau)$, and a SK_{R_i}-encrypted part consisting of N_S and the data (*e.g.* a request for file). This can be written as

$$S \rightarrow *: \{N_{R_i}^P, N_{R_i}, \alpha, [N_S, data]_{SK_{R_i}}\}.$$

When an intermediate node j, $(j \neq S, R_i)$, receives a nonduplicate data message, it looks up $N_{R_i}^P$ in its path table to get $min(\delta)$ and $min(\tau)$, and calculates its *rebroadcast probability* p_j as

$$\begin{aligned} \mu &= \frac{\alpha}{min(\delta) + min(\tau)} \\ p_j &= \begin{cases} \mu\lambda^{(min(\delta)+min(\tau)-\alpha)}, & \text{if } \mu < 1, \\ 1, & \text{otherwise,} \end{cases} \end{aligned} \quad (1)$$

where $0 \leq \lambda \leq 1$ is a real number selected by the protocol. Then, node j forwards this message according to its rebroadcast probability p_j.

When node R_i receives a nonduplicate data message identified by N_{R_i}, it decrypts the encrypted part with session key SK_{R_i} to get N_S and the data. Likewise, if node R_i intents to send a data message to S (*e.g.* the requested file), it looks up N_S from its destination table to get $N_{R_i}^P$, $min(\delta)$, τ_{R_i} and session key SK_{R_i}, and then broadcasts a data message containing $N_{R_i}^P$, N_S, a positive number $\alpha' = min(\delta) + \tau_{R_i}$ and the requested file to its neighbors. When receiving the data message, each intermediate node j, $(j \neq S, R_i)$, calculates its rebroadcast probability p'_j using (1) and forwards the data message according to p'_j.

The selection of λ represents the tradeoff between anonymity and performance. If $\lambda = 1$, the system has the highest anonymity but lower forwarding efficiency, since dummy packets contribute to collision. If λ is close to zero, the system generates the fewest dummy packets and has higher forwarding efficiency. However, since the algorithm establishes multiple anonymous paths in most cases, an acceptable

degree of anonymity is guaranteed even when λ is set to zero. The analysis of anonymity degree will be conducted in Section IV.

Propagation Delay or Hop Count?

The optimal path can be a path with minimum propagation delay or minimum hop counts. On Internet, the propagation delay reflects more precisely the distance between two nodes since a one-hop away node may be kilometers away physically. However in MANET, the one-hop distance is limited by the radio transmission range of a node, and more hops introduce more processing overhead and energy consumption. Furthermore, we observed that the propagation time reflects poorly the real distance of two nodes in MANET, especially when traffic is heavy. Routing protocol such as AODV [16] buffers the broadcast packets for an random time before sending them to the MAC layer. Moreover, the 802.11 MAC layer senses the carrier before transmitting a broadcast packet, and postpones the transmission if it senses a busy channel. Therefore when the traffic load is high, a packet may be queued for a very long time, and a node receives a packet earlier than other nodes may forwards the packet the last. Therefore, a route decision made according to the propagation time in a high traffic load period (*e.g.* the query phase, in which the network is overwhelmed by broadcast messages) may not be a right decision when the traffic load is back to normal. Therefore, the flooding control algorithm in MAPCP uses hop count information to decide the optimal path between two nodes.

The identity pseudonyms and path pseudonyms are used in MAPCP to identify the packet receiver and rebroadcast probability respectively for each communication session. Therefore, no pseudonym collision is allowed among all live communication sessions in the network. In case of pseudonym collision, the packet may be forwarded to the wrong target. Currently MAPCP ignores this problem and leaves it to the applications due to the following reasons. First, as studied in [8], for a l-bit pseudonyms, the probability of collision $p_{collision}$ when m pseudonyms are selected is

$$p_{collision} = 1 - \frac{\prod_{i=0}^{m-1}(2^l - i)}{(2^l)^m}$$

which decreases exponentially as l increase linearly, and is extremely small when l is equal to 128 bits[1], as used in MAPCP. Second, since the receiver is identified by the identity pseudonym, in case of path pseudonym collision, there is still chance for the receiver to receive the packet due to the broadcast-based communication nature of MAPCP. Third, since the identity pseudonyms can be renewed at each packet exchange, in case of identity pseudonym collision, the error can be confined within a single packet transmission.

Fig. 2 shows two examples of probability assignment results of the flooding control algorithm with λ equals to 0.9. Nodes marked by the darkest color are assigned rebroadcast probability one. The lighter the node color, the lower the probability it has. Nodes marked by the lightest color are assigned probability lower than 0.5. The samplings are conducted in a static 700m-by-700m network field, and nodes are homogeneous with radio transmission range being 250m. Fig. 2(a) presents evenly distributed nodes with a distance of 100m between their vertical and horizontal neighbors. This figure shows an ideal result that all nodes on possible shortest paths (in terms of hop counts) are assigned the highest probability. Fig. 2(b) presents a randomly generated topology, and shows that the probability assignments are not always perfect (i.e. only nodes on optimal paths are selected) due to random topologies and unpredictable collisions of query messages and query replies.

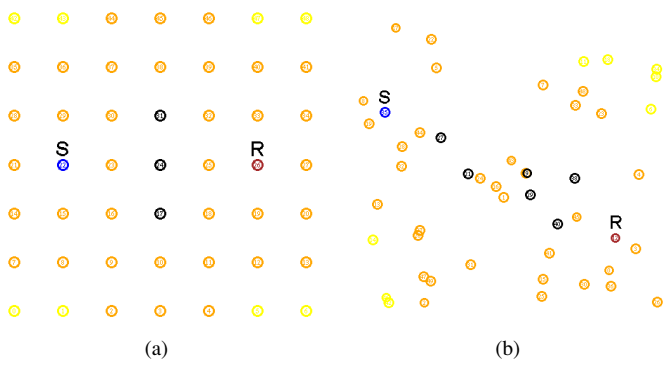


Fig. 2. Probability assignment results of flooding control in (a) a grid topology, and (b) a randomly generated topology in the 700m-by-700m network field. S is the sender, and R is the receiver.

IV. Security Analysis

Attacks to the P2P communication protocols can be roughly divided into two categories: the service attacks, in which attackers try to paralyze the P2P service (*e.g.* DoS attacks) or steal the message content, and the anonymity attacks, in which attackers try to pin down the communication parties. The design of MAPCP aims at the protection against anonymity attacks, and leaves service attacks to existing solutions such as content encryptions. This section discusses the anonymity degree of MAPCP under different attack scenarios. First, the anonymity degree is quantized using the entropy-based metric proposed by Díaz *et al.* [17] and Serjantov *et al.* [18]. Second, we discuss popular anonymity attacks and how MAPCP thwarts these attacks.

A. Degree of Anonymity

We consider the sender anonymity (the receiver anonymity can be obtained in a similar way and the anonymity degree will be around the same in two-way communication). Throughout the analysis of anonymity, we follow the definition of anonymity given by Pfitzmann and Köhntopp in [19]: "*Anonymity is the state of being not identifiable within a set of subjects, the anonymity set*", and the *anonymity set* is defined as "*the set of all possible subjects who might cause an action*". In a hostile environment, adversaries can assign each suspicious node a probability of being the message sender. The less number of suspicious nodes (i.e. the smaller the anonymity set), the higher probability each suspicious node can get. Apparently, an anonymity set which includes all nodes in a system and all nodes are equally suspicious provides the highest degree of anonymity. Unfortunately, the wireless

[1]As shown in Kong's work [8], the probability is even smaller than the probability of detection failure of a 128-bit MD5 checksum.

network is an open environment, in which all messages are broadcast in the air and are vulnerable to eavesdropping. By monitoring the node activities and the traffic flying in the air, adversaries are able to gathering information to distinguish different nodes with different probabilities to shrink the anonymity set.

The degree of anonymity can be quantified by the entropy-based metric proposed by Díaz *et al*. [17] and Serjantov *et al*. [18]. Consider a set ϕ of N nodes ($|\phi| = N$), and the anonymity attackers assign each node i in ϕ a probability p_i of being the sender according to the information eavesdropped from the system. The *entropy of this system* $H(\phi)$ is defined as:

$$H(\phi) = -\sum_{i \in \phi} p_i log_2(p_i)$$

The system has the maximum entropy H_{max} when all nodes in ϕ are equally suspicious, i.e. $p_i = \frac{1}{N}\ \forall i \in \phi$. Therefore:

$$H_{max} = -\sum_{i \in \phi} \frac{1}{N} log_2(\frac{1}{N}) = log_2(N)$$

The *degree of anonymity* provided by the system d_ϕ now can be defined as:

$$d_\phi = \frac{H(\phi)}{H_{max}}$$

Apparently d_ϕ is zero when $|\phi| = 1$ (the anonymity set consists of only one node), and $0 \leq d_\phi \leq 1$.

Therefore, if adversaries observed that there are n nodes involved in a communication session while the other $(N-n)$ nodes are quiet, they can shrink the anonymity set ϕ' to a smaller one that consists only these n active nodes ($|\phi'| = n$), and assigned each node in ϕ' the probability $\frac{1}{n}$, while others with zero probability. The anonymity degree of this system now becomes:

$$d_{\phi'} = (-\sum_{i \in \phi'} \frac{1}{n} log_2(\frac{1}{n})) \frac{1}{log_2(N)} = \frac{log_2(n)}{log_2(N)}$$

For a single-path routing protocol such as AODV and ANODR, the value of n is roughly equal to the number of hops of its discovered route. In MAPCP, since the anonymous paths are decided by the rebroadcast probability of each node $p_i^{rebroadcast}$, the value of n is then determined by the number of relay nodes, which is different in each communication session (a single run of packet exchange between the sender and the receiver). Let's define the random variables R_i, $i = 1, ..., N$, by

$$R_i = \begin{cases} 1 & \text{if node } i \text{ rebroadcasts the packet;} \\ 0 & \text{otherwise.} \end{cases}$$

Then the value of n, which is equal to the expected number of relay nodes in a communication session, is found to be

$$n = \sum_{i=1}^{N} E[R_i] = E[R] = \sum_{i=1}^{N} p_i^{rebroadcast}$$

Since the flooding control algorithm of MAPCP assigns rebroadcast probability one to all nodes on all possible optimal paths (when $X = (\delta_S + min(\tau))$), even with the settings of lowest anonymity (i.e. $\lambda = 0$), the value of n is still much larger than the hop counts of a single path. Therefore, MAPCP always provides higher anonymity degree than single-path (anonymous) routing protocols.

B. Traffic Analysis

In a more hostile environment, adversaries can detect the flow of packets and track down the source and destination by means of *traffic analysis attacks*. Traffic analysis can be launched by analyzing the timing corrections (timing attack) or the content correlations (messaging coding attack) exhibited by packets, as described below.

1) Timing attacks and flooding attacks: In timing analysis attacks [20], adversaries monitor a specific area and use temporal dependency between transmissions to trace a victim message's forwarding path. An effective way to thwart the timing attacks is to introduce more randomness of transmissions to hide the real traffic patterns. The Mix-net [5] uses playout buffers in the mix nodes to store and reorder received data packets, and to inject dummy packets into the buffer if necessary. However, this can be compromised by sending $n-1$ messages to trace a victim message when a playout buffer of size n is used by each mix node, which is also called *flooding attacks*. In ANODR [8], a variant playout-buffer scheme is used to thwart the timing attacks, and the hop-by-hop payload shuffling is used to stop the flooding attacks. MAPCP adopts similar schemes used in Mask [9] that relies on collaboratively generated dummy packets to conceal the real traffic patterns. Furthermore, we observed that the timing information required for launching the timing attack is much difficult to be obtained in wireless networks than in wired networks, especially when the wireless channels are overwhelmed by broadcast packets. Routing protocol such as AODV buffers the broadcast packets for a random time before sending them to the MAC layer. Moreover, the 802.11 MAC layer senses the carrier before transmitting a broadcast packet, and postpones the transmission if it senses a busy channel. Therefore when the traffic load is high, there are good chances that a node receives a packet earlier than other nodes, but forwards it much later than some other nodes. This makes the measurement of propagation delay insignificant since it does not reflect any more precisely the location of nodes or the forwarding paths. This observation, together with the artificially and probabilistically generated dummy packets from MAPCP, and the multipath characteristics in MAPCP, constitute an effective defense against the timing attacks.

2) Message coding attacks: Signatures of packets such as identical content, identification, and unchanged packet length can be clues for adversaries to recognize the correlation of packets and track the flow of packets. Hop-by-hop encryption, payload shuffling and random padding on forwarding packets effectively thwart this type of attacks while introduce cryptographic overhead and performance degradation [8][9]. MAPCP does not need to employ hop-by-hop encryption/decryption since the anonymous paths are constructed probabilistically and it does not need to have pairwise shared keys between adjacent nodes. However, the path pseudonym, which is used by the relay nodes for determining the rebroadcast probability, is unchanged during the entire communication session. Nevertheless, in MAPCP, the path

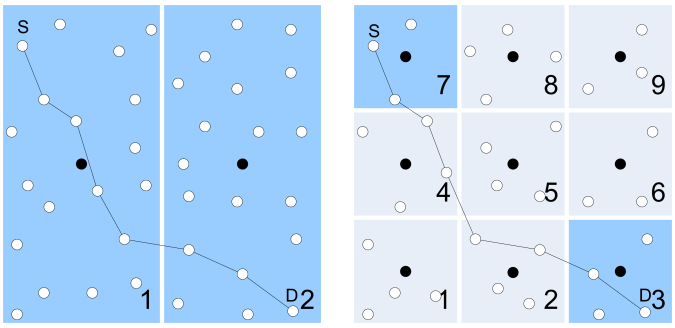


Fig. 3. By traffic analysis such as timing analysis and payload matching, colluded attackers (represented by black nodes) can divide the network space into smaller cells and shrink the anonymity set into a specific cell.

pseudonym does not reveal the real transmission paths: every node with rebroadcast probability greater than zero may rebroadcast the received packets. Adversaries can only see that there is a crowd of nodes forwarding packets with identical path pseudonym, and the observed crowd changes from time to time since nodes forward packets probabilistically. Furthermore, there is no link between the communication parties' real identities and the identity pseudonyms they are using, and the identity pseudonyms can also be changed by the sender or receiver at any time (since they share the session key and the sender's public key). Therefore, the information gained from message coding attacks is quite limited.

To analyze the degree of anonymity provided by MAPCP under traffic analysis attacks, consider the scenarios in which colluded attackers are able to divide the network space into smaller *cells*, as shown in Fig. 3. Suppose node S sends a message to node D, and the attackers divide the network space into nine cells. By timing and payload analysis, the attackers may find out that the message is originated in cell 7. Therefore, they can assume that the sender must reside in cell 7, and assign active nodes in this cell the highest probability of being the sender, while nodes in other cells probability zero. Therefore, the size of anonymity set is shrunk to the number of active nodes in that cell. The more cells the attackers divide, the smaller the anonymity set is. Apparently, dummy packets and multiple paths increase the size of anonymity set. Simulation results presented in the next section demonstrate the impact of traffic analysis on the anonymity degree.

V. Performance Evaluation

The simulation is performed based on ns-2 [21]. MAPCP is implemented as a transport agent sitting on the top of the routing agent, and a Gnutella-like P2P client is implemented at the application layer to simulate the behavior of P2P applications. We compare the performance of the two systems: (1) P2P client on the top of MAPCP with AODV as its routing protocol (MAPCP system), and (2) P2P client on the top of AODV directly [2] (AODV system). The IEEE 802.11 with the distributed coordination function (DCF) for wireless LANs is used as the MAC layer in the simulation. The radio model uses characteristics similar to Lucent's WaveLAN, with 2 Mbps channel capacity, 250m radio propagation range, and the two-way ground reflection propagation model as the physical-layer path loss model. 50 nodes are randomly distributed within the 700m-by-700m and 1000m-by-1000m fields respectively. Simulation lasts 900 seconds and each result is averaged over at least 10 runs with randomly generated topologies. MAPCP is evaluated using the following metrics:

[2]Though AODV is not an anonymous routing protocol, it still can be used to represent the single-path anonymous routing protocol in this case.

A. Degree of anonymity

We investigate the degree of sender anonymity in the scenario in which colluded attackers, by means of traffic analysis, divide the network into some smaller cells. Recall that parameters λ and α determine the anonymity degree of MAPCP. MAPCP is first evaluated under different λ with α set to $\delta_S + min(\tau) + \sigma$, where $\sigma = 0$. Then, the value of λ is fixed at 0 and α is increased by one ($\sigma = 1$) to evaluate the effect of α to the anonymity degree. We simulate 100 randomly selected one-to-one communication pairs over 20 randomly generated static network topologies, and each sender sends out one 512-byte data packet. The entropy metric defined in Section IV is used to measure the anonymity degree.

Figs. 4(a), 4(b) and 4(c) demonstrate the anonymity degree of MAPCP and AODV when the network is divided into 1, 2 and 9 cells respectively. The ticks on x-axis represent the upper bound of the linear distance between the sender and the receiver. For example, a point with $x = 500$ represents an averaged anonymity degree of all sender-receiver pairs with distance less than 500 meters but greater or equal to 250 meters. Since the radio transmission of each node is 250 meters, the x-axis also represents the linear distance in terms of hop counts. These figures show that the anonymity degree of both systems increases as the distance increases since there are more nodes involved in packet forwarding. Furthermore, MAPCP achieves higher anonymity than single-path routing protocols (represented by AODV) in all scenarios, which has justified that broadcast is an effective approach in providing anonymous communication. The figures also show that the anonymity degree of MAPCP increases as λ increases, since more nodes are involved in packet forwarding and in the generation of dummy packets. However, this is accompanied with degradation of efficiency in packet delivery since higher traffic leads to more packet collisions. Moreover, as seen in the figures, when the sender is one-hop away from the receiver, both protocols achieve the lowest anonymity degree, especially when the number of cells (created by adversaries) increases. AODV and MAPCP with $\lambda = 0$ provide almost zero anonymity for peers within one-hop distance when the network is divided into 2 or more cells. The reason is that no relay node is needed in this short distance, and the anonymity set consists of only the sender and the receiver themselves if no other node help generate dummy packets. This gives an insight that an anonymous communication protocol should provide covering when communication pairs are close to each other, *e.g.* trusted nodes generate dummy traffic to cover the real traffic patterns. In MAPCP, the covering can be provided by using a larger α, *e.g.* $\alpha > (\delta_S + min(\tau))$, as shown in Fig. 4(d). The increase of α involves more neighbor nodes in packet forwarding and hence helps conceal the location of the sender.

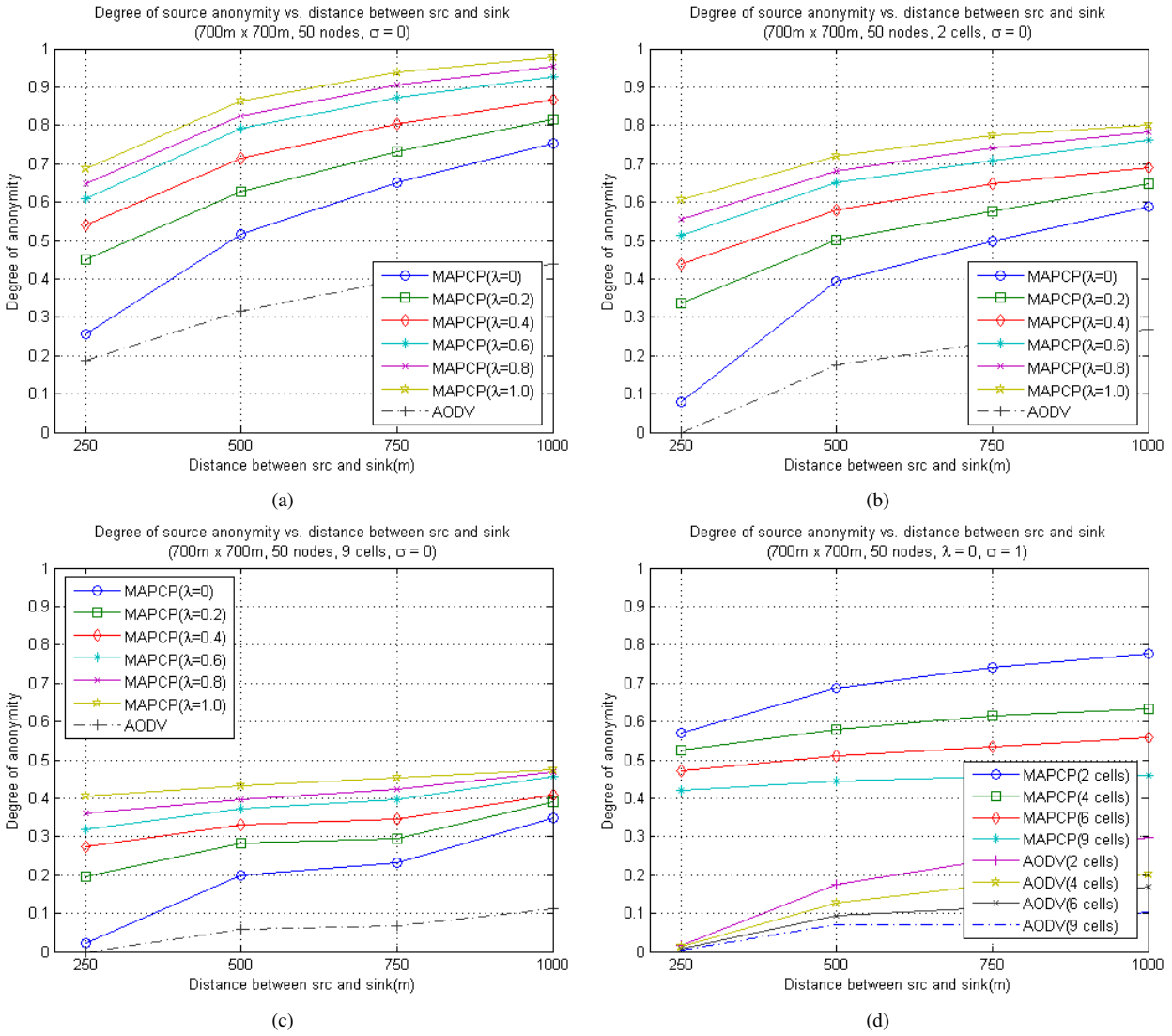


Fig. 4. Degree of anonymity in the 700m-by-700m field divided into (a) 1 cell, (b) 2 cells, and (c) 9 cells. (d) Degree of anonymity with a larger α value.

B. Performance of packet delivery

MAPCP is evaluated in terms of its performance of packet delivering and is compared to the routing performance in AODV. Both protocols are evaluated in *high mobility* and *low mobility* environments. In a high mobility environment, the node speed ranges from 0 to 20m/s with zero pause time (nonstop movement), while in a low mobility environment, the node speed is fixed at 20m/s and the pause time ranges from 0 to 900sec. The random waypoint mobility model is used for both scenarios. Simulation uses CBR sessions to generate data traffic in a rate of 4 packets per second with 512-byte data packets. To demonstrate the impact of traffic load, two different traffic settings are evaluated. The low-traffic setting constantly maintains 5 live communication pairs during the 900sec simulation, while the high-traffic setting constantly maintains 10 pairs. Each pair exchanges 100 data packets.

In this MAPCP simulation, $\lambda = 0$ and $\alpha = \delta_S + min(\tau)$. Two performance metrics are used: (1) the packet delivery fraction (PDF), which is the ratio of the number of packets received by the receiver to the number of data packets sent by the sender; (2) the average end-to-end delay of data packets, which is the duration from the generation of a data packet by the sender to the reception of it by the receiver. To simulate the cryptographic overhead in MAPCP, the computational delay of the ECAES public key cryptography (42ms for decryption and 160ms for encryption) [8] is added to the sender and the receiver upon the reception of each query reply and query message, respectively.

Fig. 5 shows the performance of packet delivery of both protocols in the 700m-by-700m and 1000m-by-1000m network fields respectively. As seen, MAPCP does not perform as good as AODV in packet delivery ratio, which is as expected, since MAPCP trades performance for anonymity and has not been optimized for end-to-end communication. The major reason of the performance degradation in MAPCP is that the broadcast-based communication causes more collisions, since there is no RTS/CTS exchange for channel reservation as in 802.11 DCF. The situation is worse when the traffic load gets higher. As seen in both Fig. 5(a) and Fig. 5(c), the PDF of MAPCP is about 95% in the 5-pair scenario, but only about 90% in the 10-pair scenario. However, the PDF of MAPCP does not degrade significantly as the node mobility increases, which

proves that the broadcast-based communication scheme adapt well to node mobility. Fig. 5(b) and Fig. 5(d) show that the average end-to-end delay of data packets in MAPCP increases as the traffic load goes high, which is also as expected since higher traffic load indicates more chances of sensing a busy channel by the 802.11 MAC layer, and hence longer buffering before transmitting the broadcast packets. Moreover, the collision of query replies may lead to the retransmission of query messages, which also introduce more encryption delay at the receiver end.

The figures also give an insight that the PDF of both protocols degrades in the network of lower node density, as shown in Fig. 5(e) and Fig. 5(g). Furthermore, the low-node-density environment magnifies the impact of node mobility. Apparently, the discovery of relay nodes for multihop communication is much harder in a sparse network than in a dense network. Nevertheless, the figures show that the end-to-end delay of MAPCP in 10-pair traffic load decreases significantly when the node density goes lower. The reason partially comes from that the channel is less busy in lower node density, which shortens the buffering delay in MAC layer and hence decreases the end-to-end delay of data packets.

C. Protocol Overhead

The overhead of MAPCP is measured in terms of the normalized number of packet transmissions and its energy consumption.

1) Normalized number of packet transmissions: We measure the normalized number of control packets, which is the ratio of the total number of control packets transmitted by any node to the total number of data packets received by all receivers, and the normalized number of data packets, which is the ratio of the total number of data packets transmitted by any node to the total number of data packets received by all receivers. We compare the overhead of MAPCP with that of AODV in one-to-many communication, in which senders and receivers are randomly chosen.

Fig. 6(a) shows the performance in terms of the normalized number of control packets. As seen, the control overhead introduced by MAPCP almost remains the same, while the overhead in AODV is proportional to node mobility. For an anonymous routing protocol, more control overhead means higher cryptography overhead and higher energy consumption. Furthermore, the normalized control overhead in MAPCP decreases significantly as the number of receivers increases. This proves that MAPCP establishes anonymous paths from one peer to multiple peers more efficiently. The normalized number of data packets shown in Fig. 6(b) indicates that MAPCP generates more redundant packets in the data transmission phase, which is as expected since MAPCP provides anonymity by generating dummy traffic. However, these packet transmissions are spread over all involved nodes instead of concentrated on nodes en route. Therefore, the energy consumption per MAPCP node, as seen in the following discussion, is still acceptable.

2) Energy consumption: We compare the energy consumption of MAPCP with that of single-path anonymous routing protocols using hop-by-hop encryption/decryption. A general hop-by-hop encryption/decryption protocol is implemented to imitate the behavior of ANODR. The implementation consists of two phases: the *anonymous route discovery* phase and the *anonymous data forwarding* phase. In the anonymous route discovery phase, the route discovery (RD) packets are broadcast to the entire network, while the route reply (RR) packets are unicast back to the source. Each node, upon receiving a nonduplicate RD packet, performs one AES encryption (to hide the route) and one AES decryption (to decrypt the trapdoor information). Each node en route, upon receiving a nonduplicate RR packet, performs one AES decryption. In the anonymous data forwarding phase, data packets are forwarded along the anonymous path established in the previous phase. For a comparison with MAPCP, nodes en route also generate dummy packets to show the extra energy consumption. The number of dummy packets generated per node en route is an adjustable parameter in the simulation. The simulation is conducted in 700m-by-700m high-mobility environment. We measure the total energy consumption, which include energy consumed by key generations, encryptions, decryptions, packet broadcasts and unicasts, according to the numbers provided in [11] and [12]. Our imitating protocol may not operate exactly the same with ANODR, however the number of these cryptographic operations will be roughly the same. Furthermore, be advised that for P2P applications, the extra overhead of query broadcasts should always be added when anonymous routing protocols are used.

Figs. 6(c) and 6(d) show the energy consumption in the route construction phase and the data transmission phase respectively. As seen, in the route construction phase, the energy consumed by MAPCP remains constant, while that of the hop-by-hop encryption/decryption based protocol increases linearly as node mobility increases, which is due to more route rediscovery processes as the mobility increases.

In the data transmission phase, we compare MAPCP with the hop-by-hop encryption/decryption based protocol that also generates different number of dummy packets, which can reflect the extra energy consumption from generating dummy packets. We measure the total energy consumed in the data transmission phase during the entire simulation process. The result is then normalized by the number of nodes involved in the communication and its resulting packet delivery fraction. The ratio in the legend in Fig. 6(d) indicates the ratio of the number of total sent data packets to the number of total sent dummy packets. Since MAPCP broadcasts the data packets onto all anonymous paths it established between the communication parties, the number of packet transmissions in MAPCP is expected to be much larger. As seen in Fig. 6(b), the packet transmission is about 5 times more than that in the single-path anonymous routing protocol without dummy packets. However, as seen in Fig. 6(d), the energy consumed by MAPCP is as low as that consumed by the hop-by-hop encryption/decryption based protocol with 1:5 ratio of data packets to dummy packets. This shows that when providing the same anonymity degree, the energy consumption in the data transmission phase is similar in both protocols. Recall that MAPCP consumes much lower energy in the route construction phase. Therefore, MAPCP is expected to prolong the network lifetime compared to the hop-by-hop encryption/decryption based protocols.

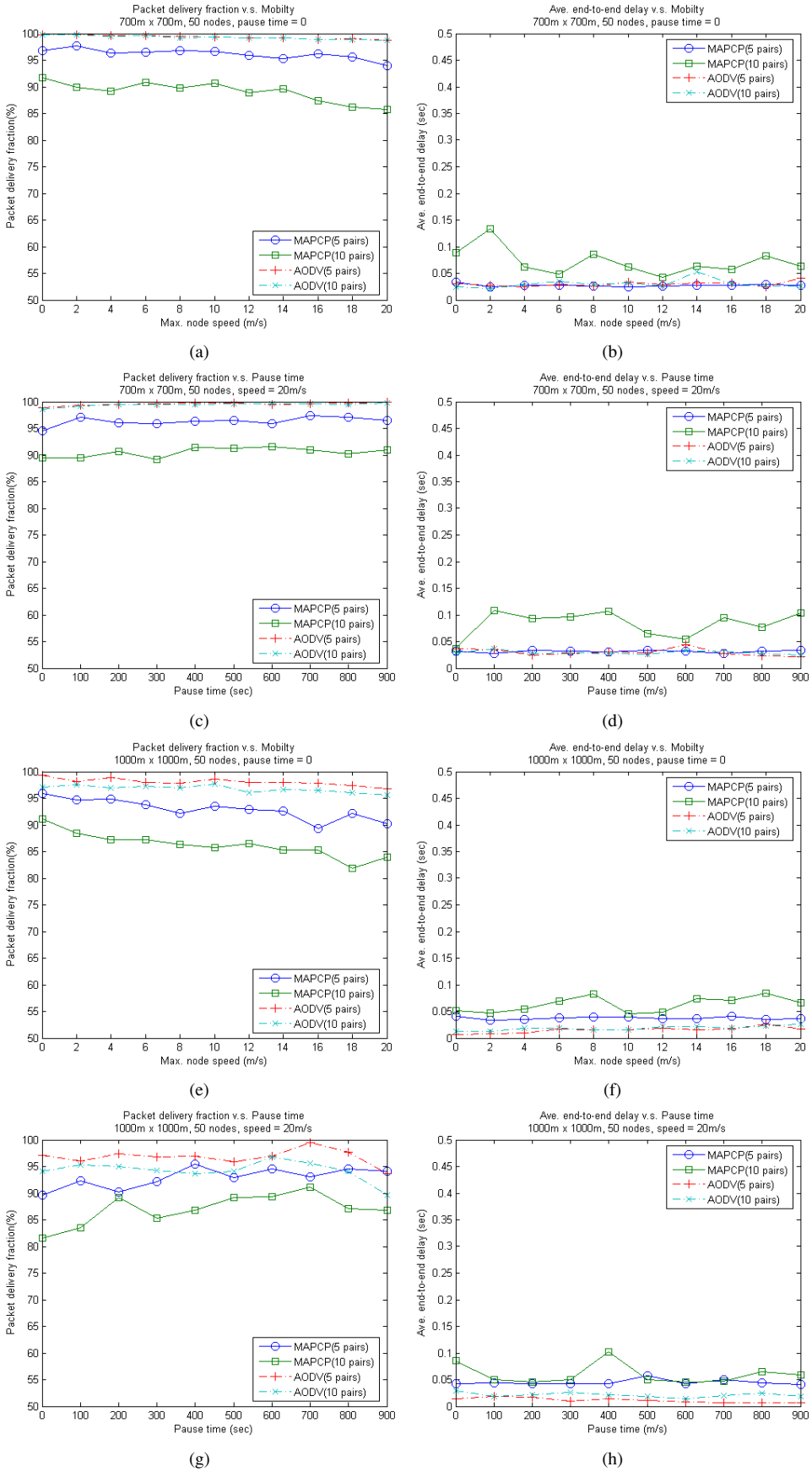


Fig. 5. (a)(b)(c)(d) Packet delivery fraction and end-to-end delay in the 700m-by-700m field with (a)(b) high mobility and (c)(d) low mobility. (e)(f)(g)(h) Packet delivery fraction and end-to-end delay in the 1000m-by-1000m field with (e)(f) high mobility and (g)(h) low mobility.

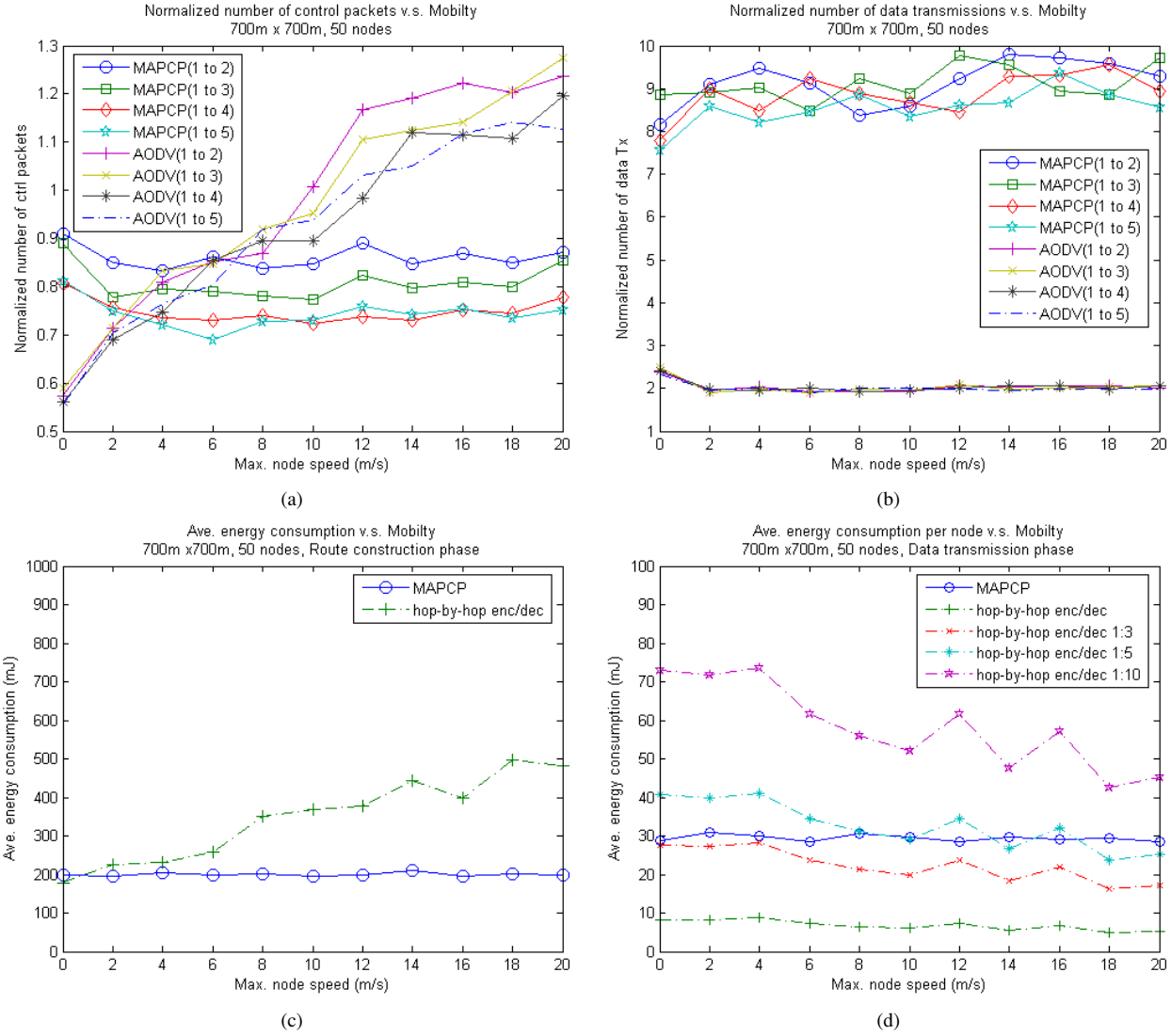


Fig. 6. Overhead in terms of (a) normalized number of control packets, (b) normalized number of data packets, (c) energy consumption in route construction phase, and (d) energy consumption in data transmission phase.

D. Effect of multipath in hostile environments

We investigate the effect of multiple paths created by MAPCP in hostile environments where compromised nodes perform *selective attacks*. Selective attack is the simplest passive attack in which the compromised node drops data packets traveling through it. For a comparison with the single-path routing protocols, AODV is also simulated. We evaluate both protocols in networks with 10% and 30% compromised nodes, and 5 CBR session pairs are constantly maintained during the 900sec simulation period. Each CBR session sends 100 512-byte data packets in a rate of 4 packets per second. The results are shown in Fig. 7. As seen in Fig. 7(a), AODV achieves only about 85% and 75% in PDF when there are 10% and 30% compromised nodes respectively, while MAPCP still maintains a PDF of higher than 90% in both cases. The difference of the performance between two protocols is more significant in a sparse network, as seen in Fig. 7(c). The results prove that providing multiple paths is an effective defence to malicious attacks, and is essential to a secured communication protocol. Furthermore, by comparing Figs. 5(b) and 5(f) with Figs. 7(b) and 7(d), we found that the delay of MAPCP is almost intact, while the delay of AODV increases significantly, especially in the sparse network (Fig. 5(f)). The increase in delay partially comes from the increased number of route re-discover processes in AODV when packets are maliciously dropped. For an anonymous communication protocol, more route re-discover processes means more broadcasts of route request packets and more cryptographic overhead, which is really a concern in a resource constrained environment such as MANET. An interesting scenario shown in Fig. 7(a) is that the selective attack somewhat improves the PDF of MAPCP when the traffic load is high, which is due to the alleviation of packet collisions when redundant data packets are dropped by the compromised nodes.

VI. Conclusion

An efficient anonymous communication protocol, called MAPCP, for P2P applications over MANET was proposed. MAPCP uses broadcast-based communication scheme and probabilistic flooding control to establish multiple anonymous

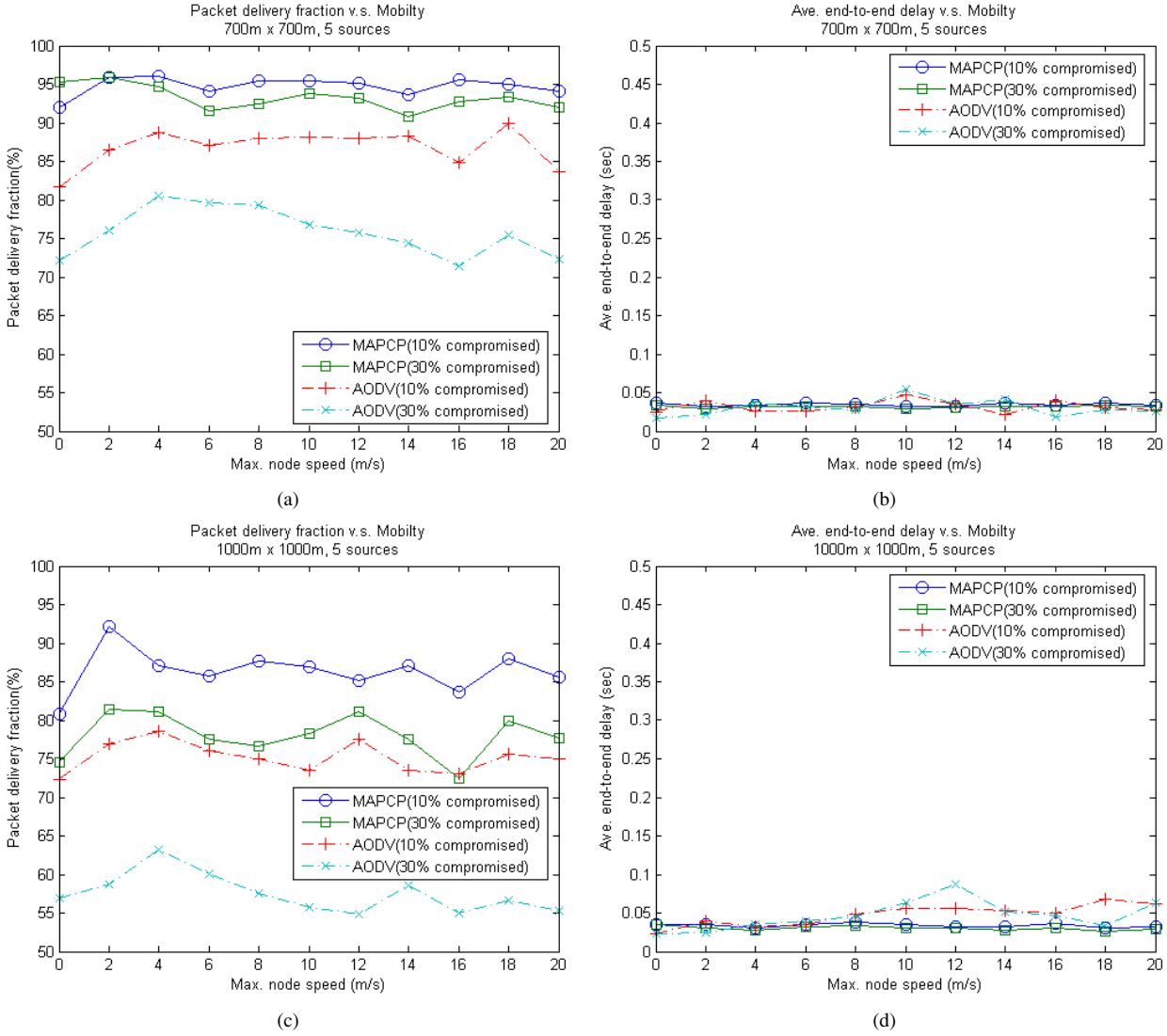


Fig. 7. Simulation results in the hostile environments. The packet delivery fraction and end-to-end delay in (a)(b) the 700m-by-700m field and (c)(d) the 1000m-by-1000m field.

paths within a single query phase. It was shown by computer simulation that MAPCP achieves a high anonymity degree even when colluded adversaries divide the network into several smaller cells. MAPCP also maintains high packet delivery fraction even under selective attacks. MAPCP is designed to be a middleware protocol sitting in between applications and network layer routing protocols and can be easily implemented on any existing MANET.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their insightful comments and suggestions.

REFERENCES

[1] M. Conti, E. Gregori, and G. Turi, "A cross-layer optimization of gnutella for mobile ad hoc networks," in *Proc. ACM MobiHoc'05*, 2005, pp. 343–354.

[2] G. Kortuem, J. Schneider, D. Preuitt, T. G. C. Thompson, S. Fickas, and Z. Segall, "When peer-to-peer comes face-to-face: Collaborative peer-to-peer computing in mobile ad hoc networks," in *Proc. IEEE P2P'01*, 2001.

[3] G. Ding and B. Bhargava, "Peer-to-peer file-sharing over mobile ad hoc networks," in *Proc. IEEE PERCOMW'04*, 2004.

[4] D. Ahmet and C.-C. Shen, "Mobile ad hoc p2p file sharing," in *Proc. IEEE WCNC'04*, 2004, pp. 114–119.

[5] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Commun. ACM*, vol. 24, no. 2, pp. 84–90, 1981.

[6] M. G. Reed, P. F. Syverson, and D. M. Goldschlag, "Anonymous connections and onion routing," *IEEE J. Select. Areas Commun.*, vol. 16, no. 4, 1998.

[7] M. K. Reiter and A. D. Rubin, "Anonymous web transactions with crowds," *Commun. ACM*, vol. 42, no. 2, pp. 32–48, 1999.

[8] J. Kong and X. Hong, "Anodr: Anonymous on demand routing with untraceable routes for mobile ad-hoc networks," in *Proc. ACM MobiHoc'03*, June 2003.

[9] Y. Zhang, W. Liu, and W. Lou, "Anonymous communications in mobile ad hoc networks," in *Proc. IEEE INFOCOM'05*, 2005.

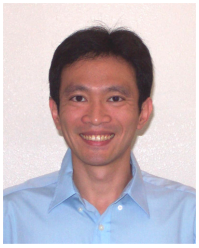
[10] M. Pearlman, Z. Haas, P. Sholander, and S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc networks," in *Proc. ACM MobiHoc'00*.

[11] L. M. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *Proc. IEEE Infocom'01*, Anchorage, AK, US, 2001.

[12] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha, "Analyzing the energy consumption of security protocols," in *Proc. ISLPED'03*, 2003.

[13] C. Shields and B. N. Levine, "A protocol for anonymous communication over the internet," in *Proc. ACM CCS'00*, 2000, pp. 33–42.

[14] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, vol. 1, no. 2–3, pp. 293–315, Sept. 2003.
[15] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proc. ACM MobiCom'99*, New York, NY, USA, 1999, pp. 151–162.
[16] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. IEEE WMCSA'99*, 1999, pp. 90–100.
[17] C. Díaz, S. Seys, J. Claessens, and B. Preneel, "Towards measuring anonymity," in *Proc. Privacy Enhancing Technologies Workshop (PET'02)*, R. Dingledine and P. Syverson, Eds. Springer-Verlag, LNCS 2482, Apr. 2002.
[18] A. Serjantov and G. Danezis, "Towards an information theoretic metric for anonymity," in *Proc. Privacy Enhancing Technologies Workshop (PET'02)*, R. Dingledine and P. Syverson, Eds. Springer-Verlag, LNCS 2482, Apr. 2002.
[19] A. Pfitzmann and M. Köhntopp, "Anonymity, unobservability, and pseudonymity: A proposal for terminology," in *Proc. Workshop on Design Issues in Anonymity and Unobservability*, 2000, pp. 1–9.
[20] J.-F. Raymond, "Traffic analysis: protocols, attacks, design issues, and open problems," in *Proc. International workshop on Designing privacy enhancing technologies*. New York, NY, USA: Springer-Verlag New York, Inc., 2001, pp. 10–29.
[21] "The network simulator - ns-2." [Online]. Available: http://www.isi.edu/nsnam/ns/



Chao-Chin Chou received the B.S. degree in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 1997 and the M.S. degree in communication engineering from the National Tsing Hua University, Hsinchu, Taiwan, in 2001. He is currently working towards the Ph.D. degree in electrical engineering at the University of Southern California.

During year 2002-2003, he joined the Computer Systems and Communication Laboratory, Institute of Information Science, Academia Sinica, Taiwan, as a research assistant. His current research interests are in the areas of wireless ad-hoc networks and peer-to-peer networks.

David S. L. Wei received his Ph.D. degree in Computer and Information Science from the University of Pennsylvania in 1991. He is currently a Professor of Computer and Information Science Department at Fordham University. From May 1993 to August 1997 he was on the Faculty of Computer Science and Engineering at the University of Aizu, Japan (as an Associate Professor and then a Professor). Dr. Wei has authored and co-authored more than 70 technical papers in the areas of distributed and parallel processing, wireless networks and mobile computing, optical networks, and peer-to-peer communications in various archival journals and conference proceedings. He served on the program committee and was a session chair for several reputed international conferences. He served as a co-chair of Power Aware Communication and Software, Minitrack in the Software Track at the 34th Hawaii International Conference on Systems Sciences (HICSS-34). He was a lead guest editor of IEEE Journal on Selected Areas in Communications for the special issue on Mobile Computing and Networking, and is a guest editor of IEEE Journal on Selected Areas in Communications for the special issue on Peer-to-Peer Communications and Applications. Currently, Dr. Wei focuses his research effort on wireless networks, mobile computing, and peer-to-peer communications.

C.-C. Jay Kuo received the B.S. degree from the National Taiwan University, Taipei, in 1980 and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1985 and 1987, respectively, all in Electrical Engineering.

He is Director of the Signal and Image Processing Institute (SIPI) and Professor of Electrical Engineering, Computer Science and Mathematics at the University of Southern California (USC). His research interests are in the areas of digital image/video analysis and modeling, multimedia data compression, communication and networking and multimedia database management. Dr. Kuo has guided about 70 students to their Ph.D. degrees and supervised 15 postdoctoral research fellows. He is a co-author of about 120 journal papers, 650 conference papers and 7 books. Dr. Kuo is a Fellow of IEEE and SPIE. He is Editor-in-Chief for the Journal of Visual Communication and Image Representation, and Editor for the Journal of Information Science and Engineering, LNCS Transactions on Data Hiding and Multimedia Security and the EURASIP Journal of Applied Signal Processing.

Dr. Kuo received the National Science Foundation Young Investigator Award (NYI) and Presidential Faculty Fellow (PFF) Award in 1992 and 1993, respectively.

Kshirasagar Naik received his BS and M. Tech degrees from Sambalpur University, India, and the Indian Institute of Technology, Kharagpur, India, respectively. He received an M. Math degree in computer science from the University of Waterloo and a Ph.D. degree in electrical and computer engineering from Concordia University, Montreal.

He worked as a faculty member at the University of Aizu in Japan, and Carleton University in Ottawa. At present he is an associate professor in the Department of Electrical and Computer Engineering, at the University of Waterloo. He was a visiting associate professor at the Research Institute of Electrical Communications at Tohoku University, Sandai, Japan, during May-November 2003. He served as a program co-chair of the 5th International Conference on Information Technology held in Bhubaneswar, India, in December 2002. He was a co-guest editor of a special issue of IEEE JSAC on Mobile Computing and Networking published in June 2005. Now he is a co-guest editor of a special issue of IEEE JSAC on Peer-to-Peer Communications and Applications. His research interests include testing of communication protocols, wireless communication, resource allocation in cellular networks, sensor networks, ad hoc networks, MAC protocols, personal area networks, mobile computing, and peer-to-peer communication.