Energy-Efficient and Path Manipulation Scheme for P2P Networks Using Trust Management

Priyadarshi Mishra Anand R Manoj Challa Dr Jitendranath Mungara

ABSTRACT:

In cooperative networks, transmitting and receiving nodes recruit neighboring nodes to assist in communication. We model a cooperative transmission link in wireless networks as a transmitter cluster and a receiver cluster. We then propose a cooperative communication protocol for establishment of these clusters and for cooperative transmission of data. We derive the upper bound of the capacity of the protocol, and we analyze the end-to-end robustness of the protocol to data-packet loss, along with the tradeoff between energy consumption and error rate. The analysis results are used to compare the energy savings and the end-to-end robustness of our Protocol with two non-cooperative schemes, as well as to another cooperative protocol published in the technical literature. The comparison results show that, when nodes are positioned on a grid, there is a reduction in the probability of packet delivery failure by two orders of magnitude for the values of parameters considered. Up to 80% in energy savings can be achieved for a grid topology, while for random node placement our cooperative protocol can save up to 40% in energy consumption relative to the other protocols. The reduction in error rate and the energy savings translate into increased lifetime of cooperative sensor networks.

Keywords: wireless networks, neighboring nodes, end-to-end robustness, cooperative protocol, energy savings.

I. INTRODUCTION

Peer-to-peer (P2P) is an alternative network model to that provided by traditional client-server architecture. P2P networks use a decentralized model in which each machine, referred to as a peer, functions as a client with its own layer of server functionality1. A peer plays the role of a client and a server at the same time. That is, the peer can initiate requests to other peers, and at the same time respond to incoming requests from other peers on the network. It differs from the traditional client-server model where a client can only send requests to a server and then wait for the server's response. With a client-server approach, the performance of the server will deteriorate as the number of clients requesting services from the server increase. However, in P2P networks overall network performance actually improves as an increasing number of peers are added to the network. These peers can organize themselves into ad-hoc groups as they communicate, collaborate and share bandwidth with each other to complete the tasks at hand (e.g. file sharing). Each peer can upload and download at the same time, and in a process like this, new peers can join the group while old peers leave at any time. This dynamic re-organization of group peer members is transparent to end-users. To study trustworthiness in mobile P2P trust management systems, we first investigate the effectiveness of various decentralized and distributed trust ratings aggregation schemes on MANETs. Specifically, the popular trust schemes including the received ratings aggregation, weighted average of ratings, Bellman- Ford based algorithm, total and ultimate trust schemes are thoroughly investigated and compared. Based on the analytical results, we propose an efficient, accurate, robust and scalable light weight trust ratings aggregation scheme, referred to as M-trust, for mobile P2P networks. We further

propose a trust ratings aggregation algorithm that acquires trust ratings not only from direct recommendations but also from recommendations from distant nodes. Results obtained from extensive simulations show that this proposed scheme can decrease the time required to compute the list of trust ratings and reduce the required storage space. The comparison to other schemes shows that M-trust possesses the excellent overall performance in terms of accuracy, reliability, convergence speed, rate of detecting malicious peers under various constraints of mobility, trust threshold and network out-degree.

I Existing System

Wireless networks provide a less robust communication than wired networks due to frequent broken links and a higher error rate. Security is also more challenging in multihop wireless networks because the open medium is more susceptible to outside attacks and the multihop communication makes services more vulnerable to insider attacks coming from compromised nodes. Although an effective mechanism against outside attacks, authentication is not sufficient to protect against insider attacks because an adversary that compromised a node also gained access to the node's cryptographic keys. Insider attacks are also known as Byzantine attacks and protocols able to provide service in their presence are referred to as Byzantine-resilient protocols.

Peer-to-Peer Key Management in Distributed Communication Systems

The key management scheme bootstraps and maintains the security associations in the network, that is, it creates, distributes and revokes keying material as needed by the networking services. The proposed key management scheme breaks the routing-security interdependency cycle and exploits the unpredictable and dynamic network topology to the advantage of security.

Group Key Management in Distributed Communication Systems

The scheme is founded on a comprehensive survey of existing schemes and their suitability for ad hoc networks. Our group key management scheme exploits the dynamic group membership and network topology to assist with the bootstrapping of security associations for the group communication system protocols. These protocols include unicast routing, group membership service, multicasting, group key agreement and data sharing. We also show how to bootstrap the group communication system by proposing a progressively robust, primary-partition group membership service. The membership service exploits the inherent capability of the group communication system to mitigate the impact of frequent group membership changes and routing failures.

Distributed-Key Management in Distributed Communication Systems

In this system, distributed-key (secret sharing) management mechanisms are proposed for generic, distributed communication systems. Specific attention is given to secret sharing in a setting without any form of online authority.

The proposed Distributed-Key Management Infrastructure (DKMI) gives group members the capability to share, update and redistribute a secret in support of a threshold cryptosystem.

Threshold-Multisignatures in Distributed Communication System

In this system presents a threshold-multisignature scheme that allows group signatures to be generated in a collaborative

fashion. The proposed scheme guarantees the signature verifier that at least a defined threshold of group members participated in the generation of the group-oriented signature and that the identities of the signers are traceable. The characteristics of secure and robust threshold-multisignature schemes are defined and it is shown that the proposed scheme satisfies these properties.

II Proposed System

Threshold-Three Layers schemes can be differentiated from threshold group signatures by the fact that by definition, in the latter, the individual signers remain anonymous since it is computationally hard to derive the identities from the group signature, with the exception of the group managers. In contrast, by the above-defined trace ability property of threshold-three level schemes, the individual signers are publicly traceable and do not enjoy anonymity. Consequently, the trace ability property of threshold-three level schemes allows the individual signers to be held accountable in the public domain and renders the unlink ability property of threshold group signature schemes, as defined in, inapplicable. In our model of cooperative transmission, every node on the path from the source node to the destination node becomes a cluster head, with the task of recruiting other nodes in its neighborhood and coordinating their transmissions. Consequently, the classical route from a source node to a sink node is replaced with a multihop cooperative path, and the classical point-to-point communication is replaced with many-to-many cooperative communication. The path can then be described as ―having a width, where the ―width of a path at a particular hop is determined by. The number of nodes on each end of a hop. ―width does not need to be uniform along a path. Each hop on this path represents communication from many geographically close nodes, called a sending cluster, to another cluster of nodes, termed a receiving cluster.

The nodes in each cluster cooperate in transmission of packets, which propagate along the path from one cluster to the next. Our model of cooperative transmission for a single hop is further depicted. Every node in the receiving cluster receives from every node in the sending cluster. Sending nodes are synchronized, and the power level of the received signal at a receiving node is the sum of all the signal powers coming from all the sender nodes. This reduces the likelihood of a packet being received in error. We assume that some mechanism for error detection is incorporated into the packet format, so a node that does not receive a packet correctly will not transmit on the next hop in the path.

Our cooperative transmission protocol consists of two phases. In the routing phase, the initial path between the source and the sink nodes is discovered as an underlying ―one-node-thick path. Then, the path undergoes a thickening process in the ―recruiting-and-transmitting phase. In this phase, the nodes on the initial path become cluster heads, which recruit additional adjacent nodes from their neighborhood.

III. Implementation

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods. The different

modules to be used are Neighbor Nodes detection, Link weight manipulation,path manipulation message transfer, Energy calculation.

Neighbor Nodes detection:

P2P Networks: A P2P Network is a decentralized network. The network is Cooperative because it does not rely on a preexisting infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks. Instead, each node participates in routing by forwarding data for other nodes, and so the determination of which nodes forward data is made dynamically based on the network connectivity.

Algorithm1:Neighbouring nodes detection using aggregation method

1: initialize list for all peers
2: **loop**
3: **for** each request(*x*) **do**
4: reply(*x*, list)
5: **if** x is direct peer and *x* _ list **then**
6: list.add(request(*x*))
7: **end if**
8: **end for**
9: **for** each *x* where *x* _ list **do**
10: **if** x is direct peer **then**
11: **if** list.value <= request(x).value
and list.value > MAX **then**
12: list.update(request(x).value)
13: **else**
14: request (y) where y≠x **and** y_ list
15: **end if**
16: **end if**
17: **if** x is not direct peer **then**
18: **loop until** time_limit
19: end if
20 **if** list.ttl(x) = 0 **then**
21: **if** list.update(request(*x*))= no reply **then**
22: remove.list(x)
23: **end if**
24: **end if**
25:end for

Link Weight Manipulation:

Each node shares there link weight between the neighbor nodes. While P2P routing defers the final route selection, the candidate forwarding nodes should still be selected in advance. This provides the different path from source node to the destination node.

Algorithm 2: Link weighted Average algorithm

1: initialize list X <- list
2: initialize list Z as empty
3: **for** each *x* where *x* _ X **and** x ≠ direct peer **do**
4: **if** x.request(z).value > MAX **and** x. > 0.**then**
5: Z.add(x.request(z).value)
6: X.update(x, _)
7: **end if**
8: **end for**
9: **if** Z≠ empty **then**
10: X.update(Compute value of (1) for each z
where z _ Z)
11: **end if**

Path Manipulation

A P2P consists of a collection of mobile nodes interconnected by multihop wireless paths with wireless transmitters and receivers. Such networks can be spontaneously created and operated in a self-organized manner, because they do not rely upon any preexisting network infrastructure. Path will be manipulated by sending the MAC from sender node to the receiver node. The receiver node will verify the MAC and the path will be manipulated. It provides more security and avoids the malicious node as well as increases the communication between the different nodes

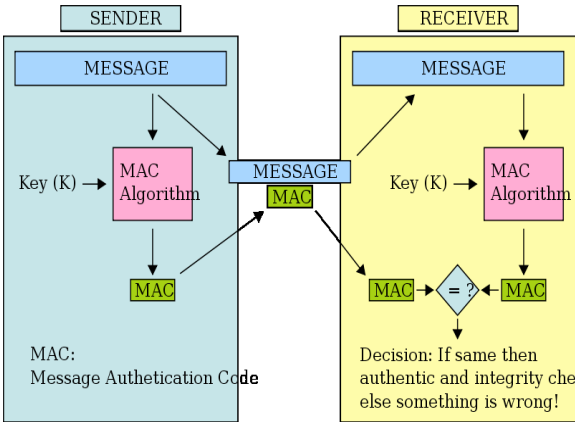

Fig1 MAC verification

CD_MAC algorithm:

Message Authentication Code= hash function with secret key:

1. Description of h public
2. X arbitrary length ⇒ fixed length m (32 . . . 160 bits)
3. Computation of hK(X) "easy" given X and K
4. Computation of hK(X) "hard" given only X, Even if a large number Of pairs {Xi, hK(Xi)} Is known Calculation of hK(X) without Knowledge of secret key: forgery
   - Verifiable or not verifiable
   - Selective or existential

Message Transfer:

The links in the networks where each link is a transmitter and receiver. Two links cannot transmit at the same time (i.e., "conflict") if there is an edge between them. Note that this framework includes the "node-exclusive model" and "two-hop interference model" mentioned as two special cases.

Algorithm 4: ECC algorithm

Embedding plaintext on an elliptic curve suppose we'd like to encrypt some plaintext with ECC. There has to be a method, which takes some arbitrary text and embedded it in elliptic curves, i.e. which gives a bijection between the points on an elliptic curve and a plaintext block. We sketch such an algorithm.

Step 1: We choose an alphabet with N letters and fix the length l of a plaintext block. The characters of the alphabet are then identified with the numbers 0, . . . , N −1. With the following assignment we get a bijection between the plaintext blocks w and the numbers 0 _ xw _ Nl: w = (a0a1 . . . al−1) 7! xw = a0Nl−1 + a1Nl−2 + · · · + al−2N + al−1, 0 _ xw _ Nl

Idea: For such an xw there need not be a point on the elliptic curve. But it should be possible to find the "next" curve point x1 close to xw efficiently. Given a number k we'd like to have a high probability (i.e. 1 − (1/2)k) for xw _ x1 < xw + k.

Step 2: We choose an appropriate k, i.e. that the success probability is high and that q > kNl. For each j we obtain an element of Fq through kxw + j. We take the first curve point (j _ 0) Pw with x-coordinate _ kxw, i.e. Pw = (kxw + j, _) 2 E(Fq).

Step 3: We can recover the plain text block from the point by xw = bx k c

Energy Calculation

Ad hoc network is a collection of wireless, mobile, dynamic, and arbitrarily located nodes. The nodes cooperate with each other to create an infra structure less temporary low cost network. The high mobility of nodes results in rapid changes in the routes, thus requiring some mechanism for determining new routes with minimum overheads and bandwidth consumption. Such infrastructure less networks use multicast routing protocols to manage random and uncertain events like rescue missions, disaster recovery, crowd control etc. The typical MANET routing protocols of IETF are shortest routing protocols and do not consider the energy aware problem. Because the ad hoc network is energy constrained system

with the portable devices. The energy saving of network is important rather than shortest path. The existing multicast routing protocols suffer from many drawbacks. The shortest path consumes more energy due to repeated usage. This makes network partition and reduce the network lifetime. This paper presents a protocol called "Energy Efficient Multicast Routing Protocol (EEMRP)" which has extended the lifetime of each mobile node by evenly utilization of energy.

The simulator for evaluating routing protocol of MRP is implemented with the network simulation version 2. Our simulation assumes the initial energy with 2, 4, 6 and 8 Joule and 100 mobile hosts placed randomly within a 1000m×1000 m area. Radio propagation range for each node is 250 m and channel capacity is 2 M bit/s. The node mobility speed is between 0m/s and 40 m/s generated by uniform distribution and the pause time is 0 s, which means the node, is always moving in the entire simulation period. Each simulation executes for 600s.The simulation altogether produces 40 kinds of stochastic topologies, each group of initial energy corresponds 10 kinds and The collected data is the averaged over those 10 runs.

Measurement of time and energy

The following formula used to find the energy level in each node.

Energy (E) = Power × time --- equation (A)

That is, when a node is transmitting or receiving a packet, the energy consumption is directly Proportional to transmitting or receiving power And the transmitted time.

The time is calculated as Time = 8 × Packet size/Bandwidth-- equation (B)

Substituting equation B in equation A

Etx = Ptx x 8 x Packet size /Bandwidth--- equation (C)

Erx = Prx x 8 x Packet size /Bandwidth--- equation (D)

Where Etx and Erx are energy consumed when packet is transmitted and received respectively. Ptx and Prx are power consumed when packet transmitted and received respectively.

The energy consumed when nodes are forwarding a packet is equal to the sum of transmitting and receiving the packet,

Et = Etx + Erx ---------- equation (E)

When a node is participated to forward a packet then net energy is calculated as

Energy = E – Et ---------- equation (F)

When a node is not participated to forward a Packet then net energy is calculated as

Energy = E – Es ---------- equation (G)

Where Es is sleeping node energy.

When a node is not participated to forward a Packet then net power is calculated as

Power (P) = Power – Battery Sleep Power -- Equation (H)

The MRP measures the energy of each node and finds the optimal route based on the energy information that is available in node cache. The node satisfies the threshold level chosen for packet transmission.

Energy Route Algorithm

Step 1: Get the Data Packet, NET size and multicast size as input parameters.

Step 2: If NET size value is equal to 1 then forward data packet.

Step 3: If NET size is not equal then compare The NET IP address with multicast IP address and check the nodes are visited Or not.

Step 4: If it is not visited then assign the Multicast IP to the node and forward it.

Step 5: Repeat the third step until the entry is equal to NET size and multicast size

Farthest Node Detection Algorithm

This algorithm Used to find the farthest node,

When no group member is found in NET.

Step 1: Get the Data Packet IP as input Parameters.

Step 2: Compare NET Battery power with Minimum Battery Power.

Step 3: If NET battery power is less than Minimum Battery Power then get the Farthest node.

EEMRP Algorithm (Data Packet DP):

This algorithm used to route the data packet to group of nodes with efficient energy saving.

Step 1: check the group is same or not.

Step 2: If it is same then update the data packet header and process step 3.

Step 3: If group size is greater than zero then Run best neighbor node selection algorithm.

Step 4: If the group is not same and group size is less than zero then exit from the function.

IV. Architecture

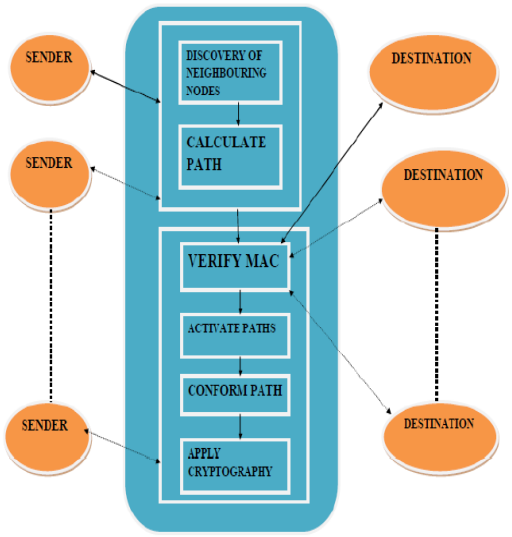

Fig 3: Architecture of proposed system

V.Result & Simulation

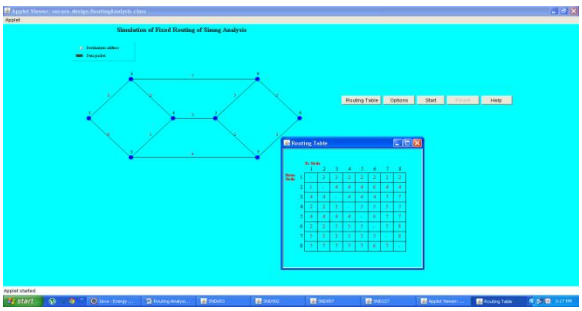

Fig 4: Calculation of energy based on routing

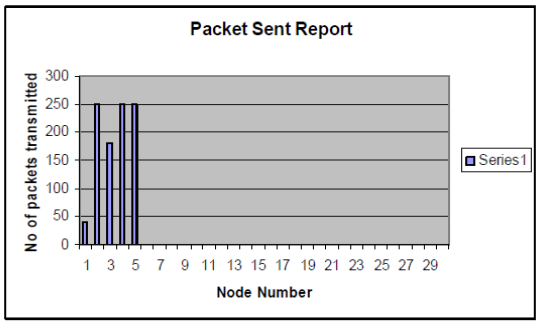

Fig 5: Number of packet send on different Nodes

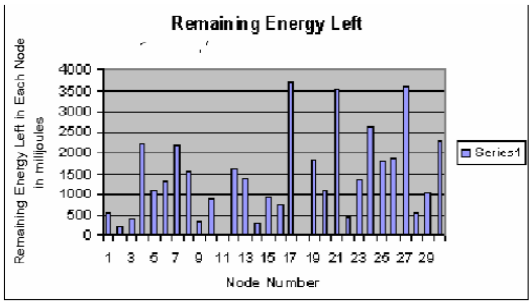

Fig 6: Energy left on each node

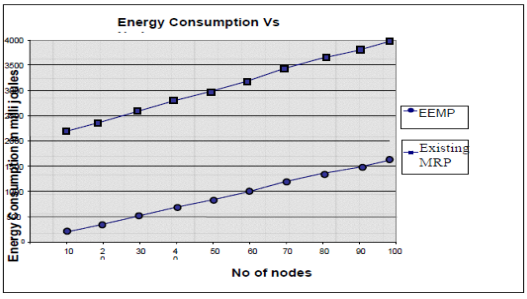

Fig 7: Performance Evaluation

VI. Conclusion

The comparison results show that, when nodes are positioned on a grid, there is a reduction in the probability of packet delivery failure by two orders of magnitude for the values of parameters considered. Up to 80% in energy savings can be achieved for a grid topology, while for random node placement our cooperative protocol can save up to 40% in energy consumption relative to the other protocols. The reduction in error rate and the energy savings translate into increased lifetime of cooperative sensor networks.

References

1] S. Bahtiyar, M. Cihan, and M. Aglayan, "A model of security information flow on entities for trust computation", Proc. of the 10th IEEE International Conference on Computer and Information Technology (CIT 2010), Bradford, UK, pp. 803- 809, 2010.

[2] J. Baras and T. Jiang, "Managing trust in self-organized mobile hoc networks", Proc. of the 2th Annual Network and Distributed Systems Security Symposium(NDSS'2005), San Diego, USA, pp.93-98, 2005.

[3] J. Bi, J. Wu, W. Zhang, "A Trust and Reputation based Anti- SPIM Method", Proc. of the 27th IEEE Conference on Computer communication, USA, pp. 2485-2493, 2008.

[4] X. Chen, G. Chen, J. Liu, X. Luo, X. Li, B. Li, "Trust factors in P2P networks", Proc. of the IEEE International Workshop on Semantic Computing and Systems, (WSCS'2008), Huangshan, China, pp.49-54, 2008.

[5] A. W. Hang, Y. Wang, and M. P. Singh, "An adaptive probabilistic trust model and its evaluation", Proc. of the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'2008), Estoril, Portugal, pp. 1485– 1488, 2008.

[6] L. Hogie, P. Bouvry, and F. Guinand, "The MADHOCsimulator." http://www-lih.univ-lehavre.fr/hogie/madhoc.

[7] T. Huynh, N. Jennings, N. Shadbolt, "An integrated trust and reputation model for open multi-agent systems". Journal of Autonomous Agents and Multi-Agent Systems Vol. 13, No. 2, pp. 119–154, 2006.

[8] J. Li, R. Li, and J. Kato, "Future trust management framework for mobile ad hoc networks: security in mobile ad hoc

[9] S. Lim, Choi Keung, N. Griffiths, "Towards improved partner selection using recommendations and trust". In Falcone, R., et al., eds.: Trust in Agent Societies (TRUST'2008). Volume 5396 of Lecture Notes in CS. Springer, pp. 43–64, 2008.

Authors

Mr Priyadarshi Mishra is presently doing Master of Technology in computer networks and engineering in CMR Institute of Technology, Bangalore Karnataka. He obtained his Bachelor of Technology degree in computer science and engineering from Padmanava College Of Engineering,Rourkela,odisa


Mr. R.Anand has completed his M.E(CSE) from Jayam College of Engineering & Technology Coimbatore in 2009. He is presently working as Associate Professor, CMR Institute of Technology, Bangalore. He presented nearly 10 papers in national and international conferences. His research areas include Wireless Adhoc networks


Mr.Manoj Challa is pursuing Ph.D. in S.V.University, Tirupati. He completed his M.E(CSE) from Hindustan College of Engineering, Tamil Nadu in 2003. He is presently working as Associate Professor, CMR Institute of Technology, Bangalore.He presented nearly 10 papers in national and international conferences.His research areas include Artificial intelligence and computer networks


Dr.M.Jitendranath is double doctorate in Electronics and Computer Science Engineering and working as Professor & Dean of Research in Computer Science Engineering department in CMRIT, Bangalore. He has published 35 papers in the area of Mobile ad hoc Networks international journals