

Community based Node Cooperative in Mobile Ad-Hoc Network

Srinivas Merugu
Sphoorthy Engineering
College & Research Scholar (KL
University)

M. Sreedevi
K L University
Vaddeswaram, Guntur

Ch. Priyanka
Sphoorthy Engineering College

ABSTRACT

Generally In a network you will find selfish nodes, which don't want to forward other's information. Because of these nodes network performance will decrease drastically, sometimes even it get disconnected. From last few years research is going on this problem, and there are so many solutions were proposed, which are related to some incentive approaches, and some Game theory approaches. But in case of a large social networks implementing these methods are difficult and also becomes more complicated. So Instead of implementing methods on single node, it is better to divide the network into communities. If we divide the social network into some communities we can define some standard method over the communities to improve the network performance against the selfish nodes.

General Terms

Node cooperation, Community, Selfish Nodes

Keywords

Community Detection, Node Centrality, cluster and friendship based Approaches.

1. INTRODUCTION

Now a day everyone is carrying mobile devices. As we know the mobile nodes will not stay in a particular location they will move in random motion. Delay Tolerant Network is a network in which the link between any two nodes will not exist always, as the mobile nodes are moving we can't assume there exist a path between any two nodes always. In case of mobile nodes there are two constraints that we need to look at: 1. Energy. 2. Memory.

In Delay Tolerant Networks it is much important that each node should cooperate to forward/store some information of others. There are several routing protocols are proposed in wireless environment like AODV and DSR, but these will not work better in case of DTN kind of networks. So there are several other routing protocols were proposed like Epidemic routing, simBet routing, Prophet routing. Most of these protocols assumed that there exist a relay node, which will store the information of other nodes, which are in more contact with that relay node in past. In most of the protocols they didn't consider existence of any selfish nodes. It is very likely that there exist a nodes, which don't want to spent their resources like energy and memory space for others information. So these nodes will degrade the network performance, and also the solutions for DTN also may not work properly. We need to develop an approach that makes the network perform well under consideration of selfish nodes.

1.1 COMMUNITY

Till now we discussed that dividing the social network into communities will helpful in many situations, But what "community" actually means?. A community can be defined as follows

- A group of people which know each other and share interests and knowledge or collaborate to reach a given target.
- A group of nodes of a graph which are more strongly connected to each other than with other nodes in the same graph (in terms of graph theory).

(or)

- A group of nodes of a graph which are more strongly connected to each other than expected in a corresponding random graph.

So in a social network we have to define relationship between nodes, so that we can convert the network into graph, in which nodes can be considered as members (or mobile devices), and links between them is, the metric we have considered between two devices.

2. RELATED WORK

Different approaches have been adapted in literature to the problem of finding non-cooperative nodes in network and making them cooperative. In past approaches they implemented Game theory approaches and some incentive approaches to make nodes cooperative.

In [6] Bridge Qiao Zhao proposed an incentive approach for peer to peer networks. He mainly considered 3 sets of nodes, Defectors, Reciprocators and Cooperators. Cooperators are the nodes which forwards every other nodes in the network. Reciprocators forwards only Cooperator nodes information as well as other reciprocator nodes information but not defector nodes information. Defectors doesn't forward any other nodes information in the network.

In the peer to peer approach they introduced 3 policies between nodes. In each policy the behavior of Reciprocators will change. 3 policies are:

Mirror Incentive Policy: Whenever Reciprocators receives any service, it will provide the services with same probability as requester serves other peers in the network. $g_1(j)=1$, $g_3(j)=0$, $g_2(1)=1$ and $g_2(3)=0$. $g_2(2) = \text{Prob}[\text{a reciprocator will grant request}]$. Where $g_i(j)$: is the probability that a peer with 'i' type services the node j. $i=1$ for cooperators, $=2$ for Reciprocators, $=3$ for Defectors.

Proportional Incentive Policy: This strategy was actually proposed in [7]. In this policy $g_2(1)=1$ and $g_2(3)=0$ remains

same. But incase $g_2(2)=1$, it means every reciprocator node support every other reciprocator.

Linear-Incentive policy: This policy has constant generosity matrix $G=[G_{ij}]$ where $G_{ij}=g_i(j)$. In this policy a node will find defectors, by constantly observing nodes, and assumes cooperators are those nodes which support defectors and remaining nodes as reciprocators. Linear strategy can serve with different probabilities P_c, P_r, P_d as specified by designer.

In [8] they proposed “Credit” based strategy. In their approach each non-cooperative node has imposed some penalty. They also considered different issues like hot areas and cold areas. In case of hot areas the traffic will be more, means node, which are connected to more other nodes in the network. Cold area means, where you find less number of nodes with less traffic. In that case the nodes, which are in hot area will get more credits compared to cold area nodes even though they are less cooperative. The Penalty can be formulated as

$$P = -\frac{\alpha l}{n}$$

Here α is constant, ‘ l ’ is the no. of hops from the message originator and ‘ n ’ is the no. of non-cooperative nodes, which are in the same level (same distance). From the above equation we can say that Penalty increases as no. of defectors are less at particular level. And Penalty also increases as distance increases from message originator. In the same way the co-operative nodes also gets some credits based energy they are losing.

Like this way there are so many approaches were there [9,10] in the past for node-cooperation, but there more complex to implement if we consider very large network. So that is why we are looking for implementation of protocol over communities.

3. PROBLEM STATE

In a adhoc mobile network, we will find lot of selfish nodes, which don’t want to transfer others information by sending their resources for forwarding and storing. Because of these selfish nodes network performance will degrade and sometimes even it may get disconnected.

To avoid such selfish nodes, many solutions proposed based on some methods, But they are not scalable and not provide security over information. So that is why we are forming communities, in which each member will co-operate to the other member and outside of community members they have to follow some strategies. Here it becomes much more scalable, like every member has to maintain strategy for entire community not for each member.

Our problem statement can be briefly described as follows:

- Finding communities in distributed way in Social Networks.
- Applying policy/Game based approach on the communities to perform well under consideration of selfish nodes.

4. COMMUNITY DETECTION

Different approaches have been adapted in literature to the problem of finding communities in social networks. There are several methods have been proposed for graph clustering, roughly they can be classified into 4 categories:

1. Node centric
2. Group centric
3. Network centric
4. Hierarchy centric

4.1 Node Centric

Node centric means each node has to satisfy some properties. Some of the node centric methods are

clique: it is maximal subgraph in which each node connected to every other in the subgraph.

k-clique: It is a subgraph in which the maximum distance between any two nodes is not more than k-hops.

k-core: A sub structure that each node should connect to atleast k-members with in a group.

In below example Clique: {4,10,12} is a clique of size 3.

2-club: {12,1,4,10} it is a sub structure of diameter $\leq k$

2-clique: {12,1,4,10,6}

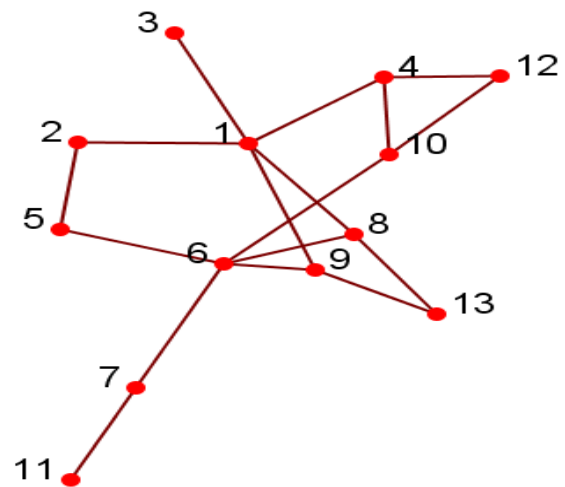


Figure 1: Example for k-clique, k-core, k-club

4.2 Group centric

In this set of nodes (group) has to satisfy some properties.

E.g : the group density \geq a given threshold.

A subgraph $G_s = (V_s, E_s)$ is a δ -dense if

$$\frac{2|E_s|}{|V_s|(|V_s| - 1)} \geq \delta$$

where the denominator is the maximum number of degree possible (complete graph).

4.3 Network centric

To form a group, we need to consider the connections of the nodes globally, and partition the network disjointly. In this grouping can be done based on following metrics:

Node similarity can be measured as how similar their interaction patterns are. We can say two nodes are structurally equivalent if they have same set of neighbors. We can find similarity by following methods:

$$\text{Jaccard}(i,j) = \frac{|N_i \cap N_j|}{|N_i \cup N_j|}$$

$$\text{Cosine}(i,j) = \frac{|N_i \cap N_j|}{\sqrt{|N_i| \cdot |N_j|}}$$

In “cut minimization”, we need to minimize the no. of cut edges between two different sets.

Modularity: modularity can be used to measure the partition quality. In the modularity we will consider some method to partition the graph and each time measure the

quality . Repeat the procedure until modularity is maximized for particular network.

$$\text{Modularity}(Q) = \frac{1}{2m} (\sum_{l=1}^k \sum_{i,j \in C(l)} (A(i,j) - \frac{d(i)d(j)}{2m}))$$

Where

k: no of clusters(communities), d(i): degree of node i, m:total no of edges in network. A(i,j): Adjacency matrix (=1 if edge exist between i and j, =0 else).

The above modularity function giving us the difference between total no.of edges existing with in the community and the edges possible in random graph.

4.3.1 Properties of Modularity

1. Between (-1, 1)
2. Modularity = 0 If all nodes are clustered into one group
3. Can automatically determine optimal number of clusters.

4.4 Hierarchy centric

There are two kinds of approaches in it

4.4.1 Agglomerative Approach:

In this approach every node is considered as separate community at the 1st phase. In each iteration we will aggregate some clusters based on modularity we will continue this procedure until modularity is maximized. In [2] Vincent D. Blondel proposed an Agglomerative algorithm, in which every node assume itself in separate community initial phase. From the 1st phase each node checks with its neighbor community means, each node is added to the one of it's neighbors and find out the modularity. Each node will repeat this procedure with it's neighbors , and node will aggregated with its neighbor node, with whom it get high modularity. This process will repeat until there is no improvement in modularity by adding communities.

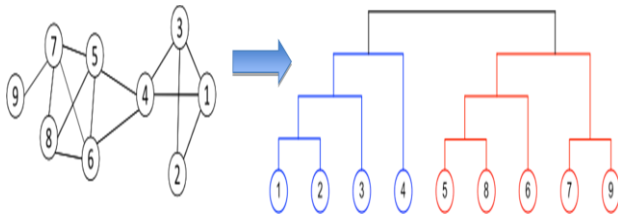


Figure 2: Agglomerative Approach

4.4.2 Divisive Approach

In this approach the whole network is to be considered as one community. In each iteration we will keep dividing the clusters up to some threshold value. Considered “**edge clustering co-efficient**” [3] is one of the metric using which we can divide the network into some clusters. It can be formulated as: for nodes i and j

$$C(g) = \frac{Z(g)+1}{S(g)}$$

Where Z: cycles of length ‘g’ between nodes i and j.

S: possible cycles of length ‘g’ between i and j.

So based on the above metric we can assign some value of ‘C’ to any edge(link). In each iteration we will remove the edge with lowest edge clustering coefficient.

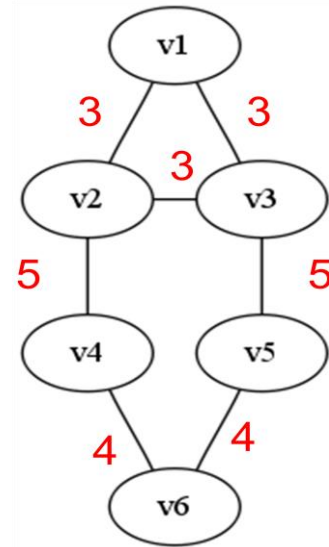


Figure 3: Edge-Betweenness

Remove e(2,4), e(3, 5), Remove e(4,6), e(5,6) and Remove e(1,2), e(2,3), e(3,1). In Figure-3 each edge is associated with its betweenness value. In each iteration we will keep removing edges, which are having high betweenness value. So we can use Edge-Betweenness as Divisive approach.

5. METHOD BASED ON NODE CENTRALITY

In previous proposals most of the algorithms for community detection is based on “node centrality “ only. Mostly in distributed algorithms for community detection , they have assumed one central node and from that central node they developed algorithms to expand the community.

Node centrality mostly based 1.**Betweenness** 2.**Closeness** 3.**Degree Centrality**

In case of Betweenness , we have to consider the nodes, which are existing in different nodes shortest path,(just like a edge betweenness).

Central nodes can also found through the, how much close they are to the remaining all nodes in the network. In the network calculate the closeness for each node and select top ‘k’ high central nodes, where ‘k’ is the no.of clusters required, From each central nodes the neighbors among them you can take as their community.

Degree centrality means the nodes with higher degree will be taken as central nodes. The nodes, with high degree have more probability to having path to other nodes(other clusters) in the network.

In [4] Lakshmish Ramaswamy Gedik and Ling Liu -proposed an distributed approach for finding communities based on cetral nodes, in which they chosen degree of a node as metric to divide the network into communities. They developed **connected-based Distributed clustering(CDC)** approach, in which the central nodes forwards the their reciprocal of their degree to it's neighbors , by keeping some hop-count in the message. So up to some number of hops the message will get transmitted with each node, who ever receives multiply the received degree with it's reciprocal degree. Finally every node will receive atleast one value from any central node. The nodes will select higher value from that and decides itself belong to that community and forwards one corresponding message to the central node .

But in the above proposal they have taken no. of clusters priorly only. It is not really good assumption in social networks. Mainly in community detection we can't decide how many communities you require in prior.

In fig(4) you will find there exist 3 central nodes. In example nodes {3,7,11} are taken as central nodes. These central nodes forward message containing "reciprocal of their degree" with TTL=4 to their neighbors. Each other nodes multiply their degree fraction with the received fraction and forwards to it's neighbors by decrementing TTL.

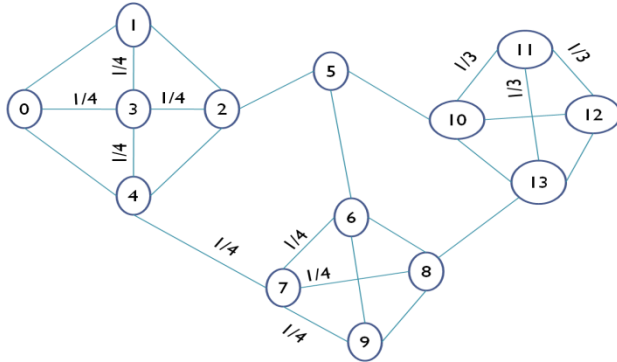


Figure 4: CDC distributed Approach using ttl=4.

In [5] Pan Hui proposed one approach, in which every the central keeps expanding it's community as long as the below condition is satisfied.

(Common neighbor between node i and j) $\geq \delta$ (total neighbors of i and j).

Where 'i' is the central node. At each central node the above condition will be checked, and based on it the communities will be formed.

6. FRIENDSHIP BASE APPROACH

In case of social networks identifying relation between devices(members) play a major role. In our work we considered that the nodes which are in same community will cooperate each other, means they are close enough to participate in communication. Suppose consider "friendship" between members in a network, most of the time they stay together, and friends can cooperate in communication rather than behaving selfish. In some papers they have considered "friendship based routing"[1] in DTN, In this paper they observed the mobile devices, which are most of the time staying together. Based the observed information they calculated one weight to each node, which are in range.

$$SPM(i,j) = \frac{\int_{t=0}^T f(t)dt}{T} \quad \text{and} \quad w(i,j) = \frac{1}{SPM(i,j)}$$

Where $w(i,j)$: weight assigned between devices($w(i,j)$ is bigger, friendship is strong), $f(t)$: Remaining time to the first encounter of these nodes, $SPM(i,j)$: social pressure metric between nodes i and j with in time period: T

7. EXPERIMENTAL RESULTS

7.1 Simulation results on Community Detection

I have taken my Facebook network for the simulation, in which all my friends considered as nodes and edges in between my friends is considered without my existence. The following result was found:

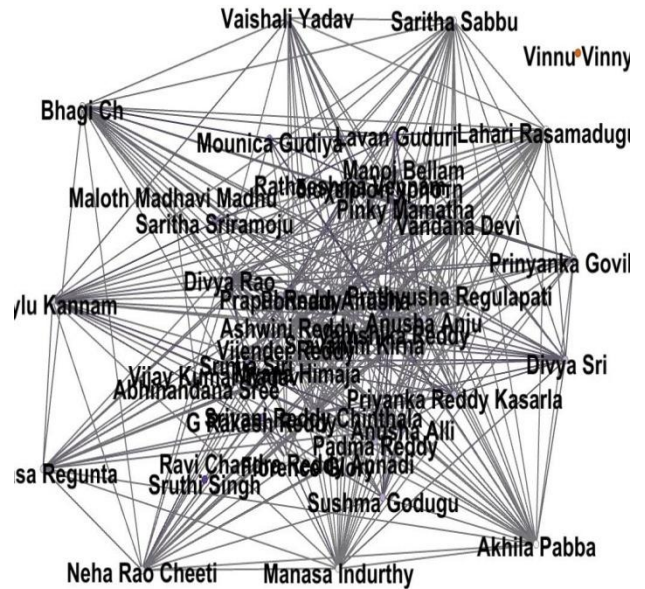


Figure 5: Community Detection

7.2 Communities Formed

community id:26.1

Akhila Pabba, Manasa Regunta, Lahari Rasamadugu, Lavan Guduri, Bhagi Ch, Mounica Gudiya, Vamshika Reddy, Ratheeshma Vennam, Shylu Kannam, Saritha Sabbu, Vandana Devi, Pinky Mamatha, Srivani Reddy Chintala, Divya Sri, Divya Rao, Manasa Indurthy, Padma Reddy, Prathyusha Regulapati, Vaishali Yadav, Srinija Siri, Anusha Anju, Sushma Godugu, Sruthi Singh, Priyanka Reddy Kasarla, Ponnam Anusha, Prinyanka Govikari, Myana Himaja, Anusha Alli, Neha Rao Cheeti, Sravanthi Kirna, Saritha Sriramoju, Ashwini Reddy.

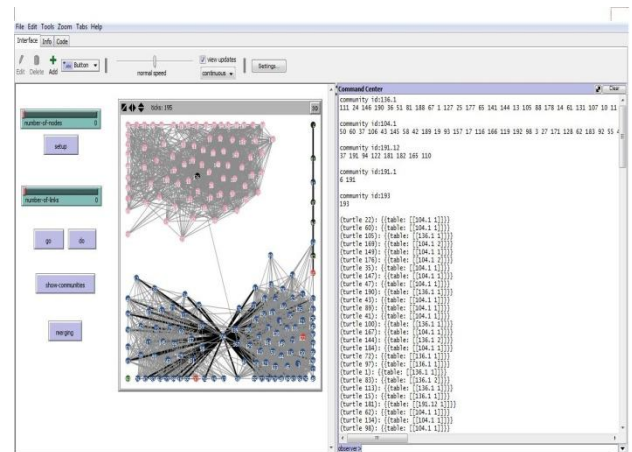


Figure 6: Simulation Results

Community id:16.1

Maloth Madhavi Madhu, Florence Glory, Abhinandana Sree

community id:8.12

G Rakesh Reddy, Ravi Chandra Reddy Annadi,

community id:8.1

Vijay Kumar Yadav , Ravi Chandra Reddy Annadi ,
Manoj Bellam ,Vijender Reddy , Prapul Reddy ,

community id:25

Vinnu Vinny

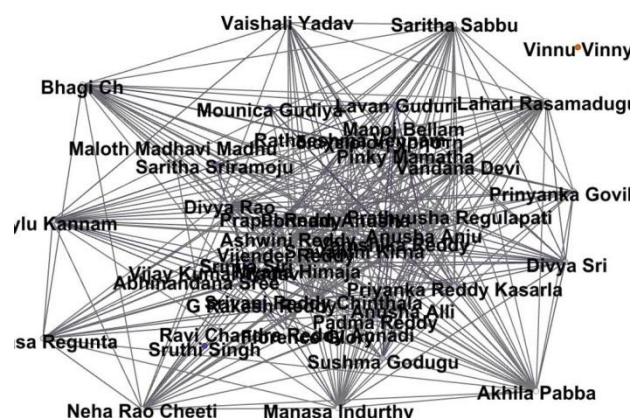
These are the communities formed from the simulation facebook friends network. Total no.of nodes: **43**

Total no.of Edges: **329**

Total communities formed: 5

Avg path-length in a community: **1.3**

From the simulation I got totally 5 communities, which are perfectly divided. Clusters formed based on my school friends, college friends, relatives.



8. CONCLUSION

Community detection is detection of single node which is communicated to group of people in each other and share interest and knowledge. We proposed to community detection methods, Node centrality and friendship based approach. Social network divided into some community, and then improve the performance of network against selfish nodes.

9. REFERENCES

- [1] Bulut, E.; Szymanski, B.K.; , "Friendship Based Routing in Delay Tolerant Mobile Social Networks," *Global Telecommunications Conference (GLOBECOM 2010)*.

- [2010] *IEEE* , vol., no., pp.1-5,6-10Dec.2010
doi:10.1109/GLOCOM.2010.5683082.
- [2] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of community hierarchies in large networks," *J. Stat. Mech.*, vol. 2008, no. 10, page:P10008,2008.
- [3] Santo Fortunato "Community detection in graphs" Complex Networks and Systems Lagrange Laboratory, ISI Foundation, Viale S. Severo 65, 10133, Torino, I-ITALY.
- [4] Lakshmish Ramaswamy; Gedik, B.; Liu, L.; , "A distributed approach to node clustering in decentralized peer-to-peer networks," *Parallel and Distributed Systems, IEEE Transactions on* , vol.16,no.9,pp.814-829,Sept.2005doi:10.1109/TPDS.2005.101
- [5] PanHui,EikoYoneki,Shu-YanChan,JonCrowcroft "Community detection in Delay TolerantNetworks".
- [6] Zhao, B.Q.; Lui, J.C.S.; Dah-Ming Chiu; , "A Mathematical Framework for Analyzing Adaptive Incentive Protocols in P2P Networks," *Networking, IEEE/ACM Transactions on* , vol.20, no.2, pp.367-380, April 2012 doi: 10.1109/TNET.2011.2161770.
- [7] M. Feldman, K. Lai, I. Stoica, and J. Chuang, "Robust incentive techniques for peer-to-peer networks," in *Proc. ACM EC*, 2004, pp.102–111.
- [8] Dayou Qian; Chi Zhou; Jinsong Zhang; , "Cooperation enforcement in ad hoc networks with penalty," *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on* , vol., no., pp.7 pp.-179, 7-7 Nov. 2005 doi: 10.1109/MAHSS.2005.1542794.
- [9] Shanshan Zheng; Tao Jiang; Baras, J.S.; , "Exploiting Trust Relations for Nash Equilibrium Efficiency in Ad Hoc Networks," *Communications (ICC), 2011 IEEE International Conference on* , vol., no., pp.1-5, 5-9 June 2011 doi: 10.1109/icc.2011.5962655.
- [10] Yanchao Zhang; Wenjing Lou; Yuguang Fang; , "SIP: a secure incentive protocol against selfishness in mobile ad hoc networks," *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE* , vol.3, no., pp. 1679- 1684 Vol.3, 21-25 March 2004 doi: 10.1109/WCNC.2004.1311805