AN APPROACH ON CLUSTER-BASED ROUTING PROTOCOL FOR DELAY-TOLERANT MOBILE NETWORKS (DTMNS)

T. Lakshmi*, L. Raghavendra Raju, D. Jamuna

1Dept. of IT, Jaya Prakash Narayan College of Engineering, Mahabubnagar, Andhra Pradesh
2Dept. of CSE, Jaya Prakash Narayan College of Engineering, Mahabubnagar, Andhra Pradesh

ARTICLE INFO

Corresponding Author: T. Lakshmi,
Dept. of IT, Jaya Prakash Narayan College of Engineering, Mahabubnagar, Andhra Pradesh

Keywords: Clustering method, delay-tolerant networks, routing, Theory based approach.

ABSTRACT

In proposed paper distributed clustering scheme and proposes a cluster-based routing protocol for Delay-Tolerant Mobile Networks (DTMNs). The basic idea is to distributively group mobile nodes with similar mobility pattern into a cluster, which can then interchangeably share their resources for overhead reduction and load balancing, aiming to achieve efficient and scalable routing in DTMN. Due to the lack of continuous communications among mobile nodes and possible errors in the estimation of nodal contact probability, convergence and stability become major challenges in distributed clustering in DTMN. To this end, an exponentially weighted moving average (EWMA) scheme is employed for on-line updating nodal contact probability, with its mean proven to converge to the true contact probability. Based on nodal contact probabilities, a set of functions including \textit{Sync()}, \textit{Leave()}, and \textit{Join()} are devised for cluster formation and gateway selection. Finally, the gateway nodes exchange network information and perform routing.

INTRODUCTION

The intermittent connectivity among mobile nodes, especially under low nodal density and/or short radio transmission range, the Delay-Tolerant Network (DTN) technology has been introduced to mobile wireless communications. Shared Wireless Info-Station Delay/Fault-Tolerant Mobile Sensor Network (DFT-MSN) and mobile Internet and peer-to-peer mobile ad hoc networks. In the existing system we cannot make the link continuously between the source and target domain. In the domain adaptation approaches that do not use the auxiliary data because the source and destination domains are far away from each other. All queries are done manually. Here manually we have to check out the syntax and errors. This process takes more time, money, energy. Difficult to understand the meaning. In proposed system the basic idea is to distributively group mobile nodes with similar mobility pattern into a cluster, which can then interchangeably share their resources (such as buffer space) for overhead reduction and load balancing.

DTN is fundamentally an opportunistic communication system, where communication links only exist temporarily, rendering it impossible to establish end-to-end connections for data delivery. In such networks, routing is largely based on nodal contact probabilities (or more sophisticated parameters based on nodal contact probabilities). The key design issue is how to efficiently maintain, update, and utilize such probabilities. Most DTN protocol”, where every node plays a similar role in routing. Clustering has long been considered as an effective approach to reduce network overhead and improve scalability. Various clustering algorithms have been investigated in the context of mobile ad hoc networks. However, none of them can be applied directly to DTN, because they are designed for well-connected networks and require timely information sharing among nodes. A proposed paper proposes a DTN hierarchical routing (DHR) protocol to improve routing scalability. DHR is based on a deterministic mobility model, where all nodes move according to strict, repetitive patterns, which are known by the routing and clustering algorithms. It cannot be generalized to such networks with unknown mobility as DTN-based peer-to-peer mobile ad hoc networks.

We propose distributed clustering and cluster-based routing protocols for Delay-Tolerant Mobile Networks (DTMNs). The basic idea is to autonomously learn unknown and possibly random mobility parameters and to group mobile nodes with similar mobility pattern into the same cluster. The nodes in a cluster can then interchangeably share their resources for overhead reduction and load balancing, aiming to achieve efficient and scalable routing in DTMN. Clustering in DTMN is unique and non-trivial, because the network is not fully connected. Due to the lack of continuous communications, mobile nodes may have inconsistent.
information and therefore respond differently. As a result, it becomes challenging to acquire necessary information to form clusters and ensure their convergence and stability. Although it is largely understood by the research community that clustering helps to improve network scalability, no previous work has been done in such emerging unique networks. In our protocol, an exponentially weighted moving average (EWMA) scheme is employed for on-line updating nodal contact probability, with its mean proven to converge to the true contact probability. Subsequently, a set of functions including \textit{Sync()}, \textit{Leave()}, and \textit{Join()} are devised to form clusters and select gateway nodes based on nodal contact probabilities. Finally, the gateway nodes exchange network information and perform routing.

2. DISTRIBUTED CLUSTERING:

We proposed clustering algorithm for DTMN, which undergoes the following steps. First, each node learns direct contact probabilities to other nodes. It is not necessary that a node stores contact information of all other nodes in network. Second, a node decides to join or leave a cluster based on its contact probabilities to other members of that cluster. Since our objective is to group all nodes with high pair-wise contact probabilities together, a node joins a cluster only if it’s pair-wise contact probabilities to all existing members are greater than a threshold.

2.1 Clustering Method

The process of dividing the network into interconnected substructure is called clustering and the interconnected substructures are called clusters. The cluster head (CH) of each cluster act as a coordinator within the substructure. Each CH acts as a temporary base station within its zone or cluster. It also communicates with other CHs. The Cluster based routing provides an answer to address nodes heterogeneity, and to limit the amount of routing information that propagates inside the network. The grouping of network nodes into a number of overlapping clusters is the main idea behind clustering. A hierarchical routing is possible by clustering in which paths are recorded between clusters instead of between nodes. It increases the routes lifetime, thus decreasing the amount of routing control overhead. The cluster head coordinates the cluster activities inside the cluster. The ordinary nodes in cluster have direct access only to cluster head and gateways. The nodes that can hear two or more cluster heads are called gateways.

With introduced for the election of cluster heads in mobile networks include the Highest-Degree, the Lowest-Identifier, Distributed Clustering Algorithm, and the Weighted Clustering Algorithm (WCA).

1) Highest-Degree (HD) algorithm: It uses location information for cluster formation. It elects the cluster head from the highest degree node in a neighborhood.

2) The Lowest-Identifier algorithm: The node with the minimum identifier (ID) is elected as a cluster head. This causes battery drainage resulting in short lifetime span of the system.

3) The Distributed Clustering Algorithm: It is a modified Version of the Lowest-Identifier algorithm. Each cluster selects its cluster head from its neighboring nodes having the lowest ID. In this algorithm every node can determine its cluster and only one cluster, and transmits only one message.

4) Weighted Cluster Algorithm: It employs combined metrics-based clustering. In order to calculate a weight factor for every node a number of metrics, including node degree, CH serving time and moving speed, are taken into consideration and in increased number of overheads. The cluster set-up procedure is invoked, when a node moves to a region which is not covered by the cluster head, throughout the whole system.

2.2 Advantages of Clustering

Clustering in AdHoc networks has many advantages compared to the traditional networks. They are as follows:

1) It allows the better performance of the protocol for the Medium Access Control (MAC) layer by improving the spatial reuse, throughput, scalability and power consumption.

2) It helps to improve routing at the network layer by reducing the size of the routing tables.

3) It decreases transmission overhead by updating the routing tables after topological changes occur.

4) It helps to aggregate topology information as the nodes of a cluster are smaller when compared to the nodes of entire network. Here each node stores only a fraction of the total network routing information.

5) It saves energy and communication bandwidth in ad-hoc Networks.

2.3 Issues of Clustering:

The highly dynamic and unstable nature of MANET’s makes it difficult for the Cluster based routing protocol to divide a mobile network into clusters and determination of cluster heads for each cluster. Clustering reduces communication and control overheads due to pre determined paths of communication through cluster heads. It is vital for scalability of media access protocols, routing protocols and the security infrastructure. Routing protocols which considers only bidirectional links may have link asymmetry due inefficient or abnormal routing. Untapped network capacity is represented by the undiscovered unidirectional links, which reduces the network connectivity. A large number of mobile terminals are managed by a MANET using a cluster topology. The construction and maintenance of a cluster structure requires additional cost compared with a topology control without cluster. The maintenance cost for a large and dynamic mobile network requires explicit message exchange between mobile node pairs. As the network topology changes quickly and concerns many mobile nodes, the number of information message exchange grows to reach a critical point. This information exchange consumes a lot of network bandwidth and energy in mobile nodes. A ripple effect of re-clustering occurs if any local events take place like the movement or the death of a mobile node, as a result it may lead to the re-election of a new cluster-head. When a new cluster-head is re-elected it may cause re-elections in the whole of the cluster structure. Thus, the performance of upper-layer protocols is affected by the ripple effect of re-clustering one of the major drawbacks of clustering in MANETs is that some nodes consume more power when compared to others nodes of the same cluster. As special node like a cluster-head or a cluster-gateway manage and forward all messages of the local cluster their power consumption will be high compared to ordinary nodes.
3. CHALLENGES FOR CLUSTERING AND ROUTING

3.1 Online Estimation of Contact Probabilities: Pair-wise contact probability has been widely used as a routing parameter in opportunistic networks. We also make use of it in our algorithm. However, one of the major problems in DTN is how to obtain this parameter distributively. A naive approach is to keep the entire meeting history. This approach, while providing robustness, is costly in storage and lacks agility to adapt to changes in mobility pattern. Therefore we adopt a simple and effective approach, named exponentially weighted moving average (EWMA). Node maintains a list of contact probabilities for every other node, which it has met before. It is updated in every time slot.

3.2 Fractional Clusters: Due to possible errors in the estimation of contact probabilities and unpredictable sequence of the meetings among mobile nodes, many unexpected small size clusters may be formed. To deal with this problem, we employ a merging process that allows a node to join a better cluster, where the node has a higher stability. The merging process is effective to avoid fractional clusters.

3.3 Inconsistent Cluster Membership and Gateway Selection: The problem of inconsistency may appear in both cluster membership and gateway selection. For example, Node leaves its current cluster and joins the new cluster. Given the network with low connectivity, other members of are not timely informed of this change and thus falsely assume that Node is still a cluster member. The inconsistency problem exists also in gateway selection for a similar reason. For instance, two nodes in the same cluster may have two different gateways to another cluster. Or a node may lose its gateway to an adjacent cluster because the gateway node has left. We deal with the inconsistency problems by employing a synchronization mechanism where nodes exchange and keep only up-to-date information.

3.4 Cluster Member with Low Contact Probability: Node with a very low nodal contact probability may still appear in the member list of another node. The main reason is that a mobile node may change its mobility pattern in real life applications.

3.5 Clustering Meta Information

Without loss of generality, a node, e.g., Node maintains its ID, its cluster ID, a cluster table, and a gateway table as its local information. The cluster table consists of four fields, namely, Node ID, Contact Probability, Cluster ID, and Time Stamp. Each entry in the table is for a node ever met by Node.

4. DISTRIBUTED CLUSTERING METHOD:

A node then decides its actions subsequently. Specially, a node will join a new cluster if it is qualified to be a member. Similarly, a node leaves its current cluster if it joins a new cluster, or it is no longer qualified to be in the current cluster. When two member nodes meet, they trigger the synchronization process to update their information. We define three main functions, namely Join, Leave, and Sync.

1) Slot-Timeout Event: Update Contact Probability:

Slot-Timeout event is generated by the end of every time slot, triggering the process of updating the contact probabilities by using the EWMA. Once the contact probabilities are updated, the Gateway Update procedure is invoked to update the gateway table. As discussed earlier, the gateway table maintains a list of gateways to each cluster. Since Node " has updated its contact probabilities to all nodes, it may potentially choose better gateway.

2) Meet-A-Node Event Update Cluster Information:

The Meet-A-Node event is generated upon receiving the Hello message (exchanged between two meeting nodes). Sync () procedure includes two steps for synchronizing cluster members and gateways, respectively. The basic idea of synchronization is to update the membership based on the latest information of Nodes and more specifically, Node sends to Node a set of its current cluster members along with the time stamps.

3) Synchronization of Gateways:

Nodes have different gateways to the same clusters in their gateway tables. Thus synchronization is needed to keep the better one with higher contact probability. For each of such clusters, the node whichever has lower contact probability gives up, and updates its gateway table. If Nodes don’t pass the membership check, one of them must leave the cluster. Two issues are involved in the Leave () procedure. First, we identify the leaving node based on its stability, defined as its minimum contact probability with its cluster members. Second, the leaving node creates a new cluster that consists of itself only. It keeps the current cluster table, because all information in the table is still valid, and resets the gateway table to be empty. Then, it sends its new cluster ID to the other node to update the cluster table accordingly.

4.1 Routing mechanisms in clustering:

4.1.1 Intra-cluster Routing

If Nodes are in the same cluster since all nodes in a cluster have high contact probability, direct transmission is employed here. In other words, Node transmits the data message only when it meets nodes. No relay node is involved in such intra-cluster routing.

4.1.2. One-hop Inter-cluster Routing

If Node looks up its gateway table. If an entry for is found, there exists a gateway, Node sends the data message to gateway. Upon receiving the data message, the gateway looks up its gateway table to and Node cluster ID. Whenever, it meets any node, cluster forwards the message, which in turn delivers the data message to Node through intra-cluster routing.

4.1.3 Multi-hop Inter-cluster Routing

If and Node and gateway table, will fail to deliver the data message, because the destination (Node) is not in any cluster that is reachable by Node "s gateways. As a result, the data transmission from Node to Node needs to be devised for multi-cluster routing. Given the low connectivity in delay-tolerant mobile net works, on-demand routing protocols do not work effectively here, because the flooding-based on-demand route discovery leads to extremely high packet dropping probability. On the other hand, any table-driven routing algorithms may be employed for multi-hop inter-cluster routing. For simplicity, a link state-like routing scheme is used. In this the protocol, every gateway node builds and distributes a Cluster Connectivity Packet (CCP) to other gateways in the network. The CCP of Gateway comprises its cluster ID, and a list of clusters to which it serves as the gateway and the actual implementation of CCP also includes a sequence number to eliminate outdated information. Once a gateway node accumulates a sufficient
set of CCPs, it constructs a network graph. Each vertex in the graph stands for a cluster. A link connects two vertices if there are gateways between these two clusters. The weight of the link is the contact probability of the corresponding gateway nodes. Based on the network graph, a shortest path algorithm is employed to find routing paths and establish the routing table. Each entry in the routing table consists of the ID of a destination cluster and the ID of the next hop cluster, in order to reach the destination. If Node "$" is not a gateway, it doesn't maintain the routing table and thus has no clue about routing. As a result, it asks the first gateway node it meets for routing information. One-hop inter-cluster routing is employed to send the data message to any node. The above procedure repeats until the data messages are delivered to the destination.

4.1.4. Load Balancing:

Load balancing is an effective enhancement to the proposed routing protocol. The basic idea is to share traffic load among cluster members in order to reduce the dropping probability due to queue overflow at some nodes. Sharing traffic inside a cluster is reasonable, because nodes in the same cluster have similar mobility patterns, and thus similar ability to deliver data messages. Whenever the queue length of a node exceeds a threshold, denoted by 1, it starts to perform load balancing.

CONCLUSION

We have proposed clustering and cluster-based routing in DTMN. The basic idea is to let each mobile node to learn unknown and possibly random mobility parameters and join together with other mobile nodes that have similar mobility pattern into a cluster. The nodes in a cluster can then interchangeably share their resources for overhead reduction and load balancing in order to improve overall network performance. Due to the lack of continuous communications among mobile nodes and possible errors in the estimated nodal contact probability, convergence and stability become major challenges in distributed clustering in DTMN. To this end, an exponentially weighted moving average (EWMA) scheme has been employed for on-line updating the contact probabilities. Including \textit{Sync()}, \textit{Leave()}, and \textit{Join()} has been devised for cluster formation and gateway selection. Finally, the gateway nodes exchange network information and perform routing.

REFERENCES