Efficient Routing using Partitive Clustering Algorithms in Ferry-based Delay Tolerant Networks

Behrouz Jedari[1] , Rouhollah Goudarzi[2], and Mehdi Dehghan[3]

[1, 2]Computer and IT Engineering Department, Islamic Azad University of Qazvin, Iran
[3]Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran
{behrouz.jedari, goudarzi.ru}@qazviniau.ac.ir,
dehghan@aut.ac.ir

Abstract

The Delay Tolerant Networks (DTNs) generally contain relatively sparse nodes that are frequently disconnected. Message Ferrying (MF) is a mobility-assisted approach which utilizes a set of mobile elements to provide communication service in ferry-based DTNs. In this paper, we propose a Density-Aware Route Design (DARD) algorithm using partitive clustering algorithms along a validity index for identifying the suitable node clusters and assigning ferries to these clusters. In the proposed algorithm, unlike using multiple ferries in a single route (SIRA algorithm) or dividing the deployment area into grid as static (NRA and FRA algorithms), the manner of node's distribution and their density in network are regarded as clustering metric. Evaluation results for comparing our scheme with existing routing algorithms demonstrate that DARD either minimizes message delivery delay or by preserving message delay, it reduces resource requirements in both ferries and nodes resulting in increasing ferries efficiency.

Keywords: *Delay Tolerant Networks, Message Ferrying, Partitive Clustering*

1. Introduction

Numerous routing algorithms such as DSR [1] and LAR [2] have been developed for data delivery in wireless ad hoc networks. The current algorithms make the assumption that the network graph is connected and fail to route messages if there is not a complete route from source to destination at the time of sending. Under these conditions, most existing routing algorithms will fail to deliver messages to their destinations since no route is found due to network partition. This raises the question of how to deliver data in a constantly disconnected network?

Delay tolerant networks (DTNs) [3] in situations such as relatively sparse nodes and energy constraints are characterized by the possible non-existence of end-to-end paths. For instance, preserving end-to-end connectivity is not always possible in MANETs, especially in the presence of factors such as node mobility and physical obstacles. Such factors cause networks to partition, either temporarily or permanently. To overcome this issue, node mobility is exploited to physically carry messages between disconnected parts of the network. These schemes are referred to as *mobility-assisted routing* [4] that employs the *Store-Carry-and-Forward* (SCF) model.

In general, mobility-assisted approaches can be classified as *reactive* and *proactive* schemes. In reactive schemes such as epidemic routing [5], applications rely on movement that is inherent in the devices themselves to help deliver messages. When disconnected, nodes passively wait for their own mobility to allow them to re-connect. Since encounters between nodes can be unpredictable and rare, these approaches suffer potentially low data delivery rates and large delays. In proactive approaches, nodes modify their trajectories proactively for communication purposes. *Message Ferrying* (MF) scheme is proposed in [6], is a proactive approach which utilizes a set of special mobile nodes called message ferries to provide communication services for nodes in the network. Similar to their real life analog, message ferries move around the deployment area and take responsibility for carrying data between nodes. MF can be used effectively in a variety of applications including battlefields, wide area sensing, non-interactive Internet access and anonymous communication.

In this paper, identifying well qualified node clusters as well as assigning ferries to them and designing efficient route for these ferries have been studied. Utilizing multiple ferries in a single route or dividing the deployment area into grid and assigning ferries for each cell as static are considered the available methods shortcomings. In this paper, a density-aware route design (DARD) algorithm by the aid of *k-means clustering* [7] and hybrid of this algorithm with *self-organizing maps* (SOM) [8] for node clustering have been proposed. In order to determine optimal clusters, we have used *Davies-Bouldin* validity index [9]. At last, by creating a *minimum spanning tree* (MST) between the node clusters, in order to connect two adjacent clusters in spanning tree, one of these two clusters with the least neighboring cluster is chosen, and then the nearest node from this cluster is selected for message transfer between two clusters.

This paper is organized as follows. Section 2 reviews the related work in mobility-assisted routing and ferry-based DTNs. In Section 3, route design principles in DTNs with multiple ferries are presented. In section 4, overviews of the partitive clustering algorithms as well as a sample of the applied algorithm in the proposed scheme are introduced. We propose our route design scheme in Section 5. Section 6 presents the performance evaluation with simulation results. This paper concludes in Section 7.

2. Related work

Mobility-assisted communication is an active research area in sparse DTNs, ad-hoc networks, sensor networks and robotics community. In the ZebraNet project [10], sensors are attached to zebras and used to study the behavior of wildlife. Due to device form factor and energy constraints, only short range radios can be used. In the DakNet project [11], vehicles are used to transport data between remote areas such as villages and cities to provide store-and-forward Internet access.

Li and Rus [12] consider proactive movement of nodes to deliver messages in a disconnected environment and present an algorithm to compute optimal node trajectories. In [6], Zhao et al. introduce the message ferry scheme. In [13], Tariq et al. consider the route design problem for a single ferry in sparse ad-hoc networks. Viswanathan et al. in [14] have studied the delivery quality of service (QoS) for certain urgent messages in the constrained and the relaxed constrained MF systems.

All the above work does not discuss the route design problem for multiple ferries. Authors in [15] study the route design problem for multiple ferries. Basically there are two extremes of route design for multiple ferries. At one end, we can have all ferries serve the whole network and follow a single route that passes through all nodes (Fig.1 (a)). This is a *single route approach* (SRT). At the other end, each ferry can serve a different portion of nodes and follow a different route. In order to keep the network connected, some ferries may serve one or more nodes in common (Fig.1 (b,c)). This is a *multiple routes approach* (MRT). Finally, they conclude that all the algorithms are scalable and achieve similar results. In [16], an analytically tractable model to a quantitative one describes the message delivery delay in a multi-ferry scheme is proposed. In Our work, instead of dividing the networks into grid, the manner of node's distribution and their density in network are regarded as clustering metric of nodes. We give detailed comparison of different design schemes and articulate their trade-offs via simulations.

3. Network Model

As shown in Fig. 1, ferries follow pre-designed routes by which they can regularly visit places where stationary nodes. Upon contacting with a node, the ferry uploads messages originated from that node and downloads data destined for the node. The network is sparsely deployed so the radio range of a node is smaller than the distance between any pair of nodes; otherwise a cluster of connected nodes can communicate with each other through traditional ad-hoc routing mechanisms, leaving ferries carrying data among gateway nodes that are located in isolated clusters. We assume that bandwidth requirement between each pair of nodes is known and constant over time. In practice, while traffic might not be known in advance, traffic demand can often be estimated. This is true especially for the MF scheme, which is expected to operate at large time scales, e.g., in minutes or hours.

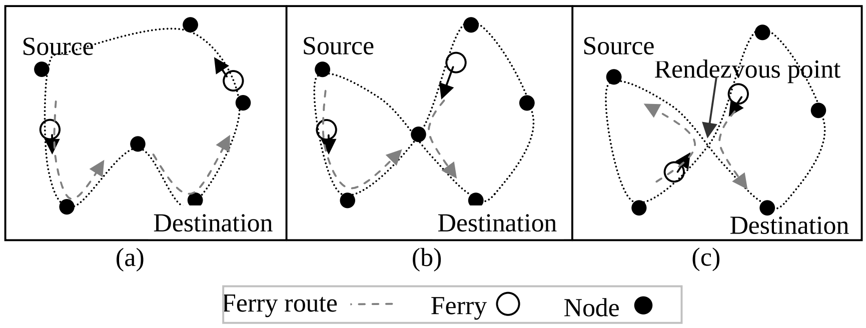


Figure 1. MF schemes: (a) Single route with 2 ferries, (b) Multiple routes with a relaying node, (c) Multiple routes with one ferry

We suppose there are M ferries, f_i $(1 \le i \le M)$, and N nodes, n_i $(1 \le i \le N)$ where N is much large than M. Any route design scheme for multiple ferries can be represented by an allocation of nodes to ferries $R = \{R_i\}$, where $R_i = \{n_j\}$ is a subset of nodes assigned to f_i. Note that a given node can be assigned to more than one ferry. Two ferries are regarded as connected if they share common nodes in their allocation sets. In order to make the network connected, we require that:

- $\bigcup_{i=1}^{M} R_i = \{n_i,...,n_N\}$. Means each node must be served by at least one ferry.
- $\forall R_i, R_j, \exists$ a sequence of $R_{i0}...R_{ih}$, such that $R_{i0} = R_i$ and $R_{ih} = R_j$ and $R_{ik} \cap R_{ik+1} \neq \phi (0 \le k < h)$. This means that any node is reachable from any other node by a ferry or a series of ferry relays.

3.1. Route Design in Ferry-based DTNs

The ferry route problem consists of finding an optimal route T such that the bandwidth requirements for all nodes are met and the average delay is minimized. Rather than addressing the combined problem, Zhao and Ammar in [6] break it into two sub-problems. The first one seeks to find a route that minimizes the average delay for the expected traffic matrix without considering the bandwidth requirements. The second sub-problem extends the route generated in the first sub-problem, if necessary, to meet the bandwidth requirements.

The delivery delay of a message consists of two parts: the waiting delay that is the sum of the time waiting for the arrival of any ferry, and the carrying delay which is the time a message carried by any ferries from source to destination. Let d_i denote the average message delivery delay between a pair of nodes assigned to f_i. T_{ij} is the normalized traffic from nodes in R_i to nodes in R_j. In particular, T_{ii} represents the normalized traffic originated from and destined to the nodes in R_i. According to our model, the message delivery delay in MF scheme is [16]:

$$d = \sum_{i=1}^{M} T_{ii} d_i + \sum_{i \neq j} T_{ij} \left(\sum_{f_k \in path(f_i, f_j)} d_k\right) \quad \textbf{(1)}$$

Where *path* (f_i, f_j) is an ordered set of ferries along the path from f_i to f_j. For SRT, $\forall_i$, $R_i = \{n_i, ..., n_N\}$, and $\forall i \neq j$, $T_{ij} = 0$. So we have $d^{SRT} = \sum_{i=1}^{M} d_i T_{ii}$. Now we consider how to extend the ferry route, if necessary, to meet the bandwidth requirements of the nodes. For any given route, the achieved data rate of a node is λW where λ is the fraction of time the node communicates with ferries. We need to extend the amount of time ferries spend in the vicinity of those nodes that do not otherwise have enough time to transmit or receive their data. In practice, the extension could consist of changing ferry speed, so that the ferry spends more time on certain parts of the route. We need to extend the amount of time ferries spend in the vicinity of those nodes that do not otherwise have enough time to transmit or receive their data. In practice, the extension could consist of changing ferry speed, so that the ferry spends more time on certain parts of the route. This problem is formulated using linear programming (LP) as follows. Let x_i be the length of detour in the vicinity of node i. We assume ferries move to the location of each node, thus the total length of the ferry route that is within the radio range of node i is x_i+2r. Let S_i be the total data rate for node i which is the sum of data rates in both transmission and reception. By distributing traffic load equally to ferries, each ferry is responsible for supporting a data rate of S_i / M. Thus we have:

$$\frac{(x_i + 2r)W}{L + \sum_{j=1}^{N} x_j} \geq \frac{S_i}{M} \quad \textbf{(2)}$$

Where L is the length of the ferry route before extension. After transformation, we get the following optimization problem:

Minimize $\sum_{i=1}^{N} x_{i,}$ **(3)**

Subject to $MWx_i - S_i \sum_{j=1}^{N} x_j \geq S_i L - 2MrW, x_i \geq 0, and, 1 \leq i \leq N$

The above problem can be solved efficiently using methods like Simplex [17].

4. Partitive Clustering Algorithms

Clustering algorithms attempt to organize unlabeled input vectors into clusters such that points within a cluster are more similar to each other than vectors belonging to different clusters. There are multitudes of clustering methods in the literature, which can be broadly classified into the following categories: hierarchical clustering, Partitive clustering, grid-based clustering and model-based clustering. In this paper, the Partitive clustering category is considered.

Partitive clustering algorithms divide a data set into a number of clusters, typically by trying to minimize some criterion or error function. If the number of clusters is unknown, the partitive algorithm can be repeated for a set of different number of clusters, typically from two to $\sqrt{N}$, where *N* is the number of samples in the data set. Partitive methods are better than hierarchical ones in the sense that they do not depend on previously found clusters. On the other hand, partitive methods make implicit assumptions on the form of clusters. For example, k-means algorithm [7] tries to find spherical clusters.

4.1. The k-means type Algorithms

Let $X = \{X_1,..., X_N\}$ be a set of *N* objects. Object $X_i = \{X_{i,1},..., X_{i,c}\}$ is characterized by a set of *c* variables. The *k-means* type algorithms [7], search for a partition of *X* into *k* clusters that minimizes the objective function *P* with unknown variables *U* and *Z* as follows:

$$P(U,Z) = \sum_{l=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{m} u_{i,l} d(x_{i,j}, z_{l,j}) \quad \textbf{(4)}$$

Subject to $\sum u_{i,l} = 1, \quad 1 \le i \le n$

Where:

- U is an $n \times k$ partition matrix, $u_{i,l}$ is a binary variable, and $u_{i,j}$ indicates that object *i* is allocated to cluster *l*;
- $Z = \{Z_1,..., Z_K\}$ is a set of *k* vectors representing the centroids of the *k* clusters;
- $d(x_{i,j}, z_{l,j})$ is a distance or dissimilarity measure between object *i* and the centroid of cluster *l* on the j_{th} variable. If the variable is numeric, then:

$$d(x_{i,j}, z_{l,j}) = (x_{i,j} - z_{l,j})^2 \quad \textbf{(5)}$$

Regarding the fact that the k-means algorithm has the capability of assigning the suitable clusters for different distributions of the nodes, this algorithm has some inherent shortcoming as follows:

- Number k is often not known advance
- Clustering results depend heavily on centroid initialization.
- It is sensitive to outliers.

Hence, for covering the algorithms shortcoming, the hybrid of k-means clustering with *self-organized maps* (SOMs) as a two-level method is used to find optimal clusters of nodes. The self-organizing maps Proposed by Kohonen [8] is an unsupervised neural-network approach that provides a similarity graph of input data. The success of the SOM algorithm lies in its simplicity that makes it easy to understand, simulate and be used in many applications. The basic SOM consists of a set of neurons usually arranged in a two-dimensional structure such that there are neighborhood relations among the neurons. A typical simplified version of the SOM algorithm consists of two steps iterated for every sample: (a) finding the best matching units; (b) adaptation of the weights. Initially, all neuron weights are initialized by uniform distribution.

5. Route Design using Partitive Clustering Algorithms

In this section, we have studied the manner of using clustering methods for route design in ferry-based DTNs. In the following, a two dimensional node position in deployment area is utilized as a clustering metric. The term *cluster* means a group of nodes belong to a common centroid, and distance between nodes is done through a similarity function as Euclidean distance. In the rest of this paper, two partitive clustering algorithms are presented that perform Node clustering and allocate ferries to each cluster. Our proposed algorithm performs nodes clustering and ferry route designing between the nodes in two steps as follows: At first step, two clustering algorithms, mentioned in section 4, for network clustering are utilized, called DARD-I and DARD-II. In the next step, by choosing gateway nodes between adjacent clusters, a linear programming approach, stated in section 3.1, is used for establishing communication between nodes. Regarding the fact that the proposed methods make use of nodes for message relay, the evaluation comparison of these methods is also done with NRA and SIRA algorithms. In the next section, all the cases will be explained in details.

5.1. Node Assignment in DARD-I

In the DARD-I algorithm, using the k-means algorithm, k points are initialized as clusters centroid. This points are displayed with * in the Fig 2. Determining the initial values for k and evaluating the quality of acquired clusters are of those issues that affect clustering quality as well as optimizing ferries traveling route. On the other hand, by reducing the length of traveling route inside each cluster and using acceptable ferries in network, message delivery and buffers consumption in ferries and nodes would be also decreased. It is apparent that the less clustering error in acquired clusters, the efficient route designing between nodes will exists, it also will satisfy some qualitative metrics.

Identifying the number of final clusters and allocating primary values to the cluster centroid are the most important problems that k-means algorithm and generally, most of partitive methods suffer from them. Considering the fact that in k-means algorithm initially values for k are selected randomly, and these values influence the quality of final clusters, hence, in the evaluation, we have repeated clustering operation for sets of different numbers of clusters with different centroids several times and finally, clusters with high quality are selected as final clusters. Regarding the experimental results, we have considered the maximum number of clusters as $\sqrt{N}$, typically from two to $\sqrt{N}$, where N is the number of nodes in the network. At last, clusters with minimum Validity index are chosen as final clusters.

Due to Spherical shape of clusters in k-means algorithm, it increases the efficiency of the designed routes for message ferries. Unlike available methods such as NRA algorithm, finding the spherical-shaped clusters with low clustering error reduce the length of ferry route within clusters, which finally leads to a rapid message delivery. Finding an optimal route inside each cluster as well as extending route to meet a node’s bandwidth requirement are perform based on section 3.1.

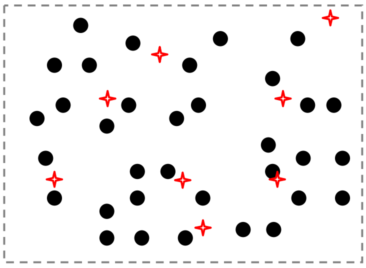


Figure 2. Node clustering using k-means method in DARD-I algorithm

5.2 Node Assignment in DARD-II

In the second method so called DARD-II algorithm, for achieving optimal clusters and designing the efficient routes through networks, a hybrid method is utilized. The given proposed method minimizes clustering error remarkably; this condition leads to reducing the length of ferries traveling routes. Partitioning the network into very small area or and assigning central coordinates of each area as the centroid are the first steps in the proposed algorithm. At the first stage, distributed nodes are allocated to the nearest cluster, that in assigning nodes to the given clusters, SOMs are used. In this algorithm, each of mapping units are recognized as a cluster that their numbers experimentally are selected the same value as $5\sqrt{N}$ and are much more than the number of given final clusters.

In the second stage, instead of clustering the nodes, central values of the mapping units are grouped to reach a group of nodes. In clustering the mapping units, k-means algorithm is used. The most eye-catching characteristics of the two-level method is minimizing clustering components from N node to $5\sqrt{N}$ of the mapping units that the more optimized clusters of node are gained by preserving nodes settlement focus.

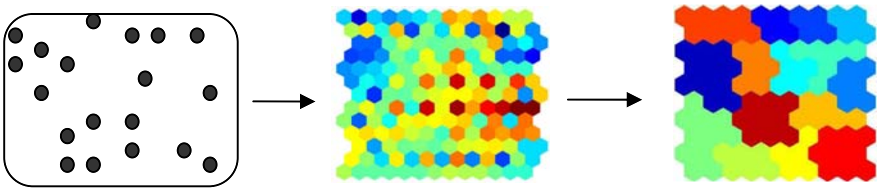


Figure 3. Network clustering using a hybrid method in DARD-II algorithm

5.3. Identifying Optimal Clusters with Davies-Bouldin Validity Index

To select the best one among different partitioning, each of these can be evaluated using some kind of validity index. Several indices have been proposed. In our simulations, we used the Davies–Bouldin index [9], which uses for within-cluster distance and for between clusters distance. According to Davies–Bouldin validity index, the best clustering minimizes following equation:

$$\frac{1}{C}\sum_{m=1}^{C}\max_{l\neq m}\left\{\frac{S_c(Q_m)+S_c(Q_l)}{d_{ce}(Q_m,Q_l)}\right\} \quad \textbf{(6)}$$

In this equation, C is the number of clusters, m and l are representative of 2 different clusters. $S_c(Q_l)$ is intra-cluster distance and d_{ce} is inter-cluster distance. The Davies–Bouldin index is suitable for evaluation of k-means partitioning because it gives low values, indicating good clustering results for spherical clusters.

5.4. Connectivity between Clusters using Relaying Nodes

After clustering the nodes by one of the proposed algorithms, DARD-I or DARD-II, in this section, the policy of gateway nodes assignment for establishing communication between clusters and message transfer through the network is presented. The generated messages by the nodes are divided into 2 groups: local and non-local messages. The active ferry (or ferries) in each cluster is responsible for the local message delivery at inter-cluster. Also, for carrying and delivering the non-local messages from one node to node in another cluster, the message should pass multiple relaying nodes as well as active ferries inside the clusters.

Looking at the issue of message delivery to non-local targets and assuming that every cluster in the network is considered as a node of one graph, it is possible to perform the manner of clusters connectivity to each other based on graph-based approaches or to do routing and message delivery between nodes by establishing *minimum spanning tree* (MST) between clusters. Considering that full-connected graph formation between clusters increases the length of ferries traveling route, in this section, by finding MST between nodes, we have used message relaying by gateway nodes for message communication.

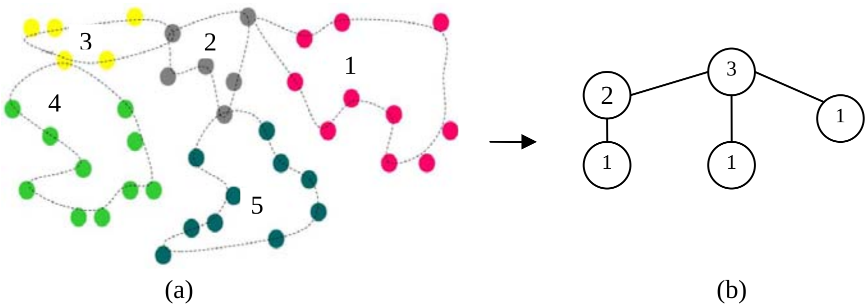


Figure 4. (a): Communication between clusters with node relaying in DARD algorithms, (b): Each node shows a cluster with an adjacent value

For identifying gateway nodes, first nodes coordinate average is calculated in every cluster. Then the established points to the number of clusters are regarded as a node of one graph that Euclidean distance average between clusters is shown by an edge which is considered as message delivery cost between clusters. Now we are going to find MST between nodes to minimize message delivery cost between areas or on the other hand, the sum of communication edges cost. In the current paper, *Dijkstra's algorithm* is used to find MST between nodes as shown in Fig. 4. Each node in this figure is considered as a cluster and the inter-node edges shows communication route between clusters, and numbers inside every node indicate the degree of the node or of gathering rate or of message flow traffic in the given cluster. The nodes labeled as 4(a), for example, are connected to 4 clusters performing as a bottleneck in message communication between different parts of the tree and should tolerate high traffic and also be able to transmit to the other area.

In NRA, the nearest node pair is selected at the neighboring area and one of these nodes as a relaying node is responsible for establishing communication between areas. In this algorithm, if the network is partitioned into 3 rows and 3 columns, the number of relaying nodes will be 12. But in DARD-I and DARD-II algorithms, if the number of acquired clusters is *d*, so the number of relaying nodes will be *d-1*.

After identifying *MST* between the clusters, we will discuss the manner of establishing communication between clusters. The communication edges between clusters connect the neighbor clusters by means of the nearest nodes. Now, for choosing one of the two communication nodes between 2 clusters, it is possible to assign one of the nodes as the common node between 2 clusters that the ferries of both meet this node during the route and delivery non-local messages to this node. On the other hand, the ferry of the neighboring cluster by establishing communication with gateway node also receivers the non-local messages from the gateway node and thus, message communication between nodes is created.

Fig 4(b) shows applied policy in identifying relaying nodes between the clusters of Fig 4(a). As you notice, the cluster 2 is a three-degree cluster that is connected to the neighboring cluster 1, 3 and 5 having 1, 2 and 1 degree respectively. Regarding high degree of the cluster 2, that tolerant high flow traffic, developing route for communicating this cluster with other neighboring clusters isn't done from this cluster and the active ferries in the neighboring clusters extend routes for visit relaying nodes.

6. Simulation Results

In this section, we evaluate the performance of the Message Ferrying schemes using simulation. We model from 50 to 300 nodes distributed in a rectangular area 4000m × 4000m. All ferries are deployed which moves at a speed of 20 m/s. We allow messages to generate until a simulation time of 4000 seconds and run the algorithms until 6000 seconds. In our simulation, we choose IEEE 802.11 as the MAC layer protocol. The antenna is omni-directional and the radio transmission range in ferries and nodes is 100m. We consider ad hoc traffic, where each node sends data to a randomly chosen node. The wireless network interface transmits data at a rate of 1Mbits/sec. Messages with size of 1 Kbytes are generated as a Poisson process. We simulate a fair sharing scheduling policy in which each node gets equal access to the ferry when the ferry is within range of multiple nodes. For each setting, the result is averaged over 10 runs with different random seeds.

6.1. Performance Metrics

The objective of our algorithms is to either minimize message delivery delay or by preserving message delay, it reduces resource requirements in both ferries and nodes resulting in increasing ferries efficiency. Although the applications in sparse MANET should be delay tolerant, shorter delivery delay and resource consumption is still preferable. Therefore, we use the following metrics in our evaluation of the algorithm:

Message delivery delay: Average delay between the time a message is generated and the time the message is received at the destination.
Buffer consumption: Number of stored messages in ferries and nodes buffers.
Energy consumption: Number of sent and received messages for nodes and ferries.

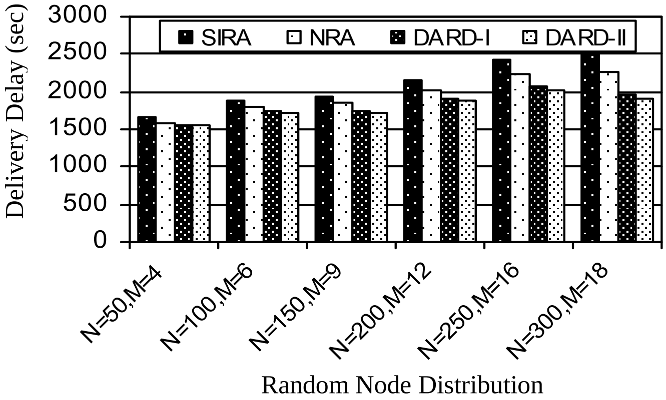


Figure 5. Message delay under different numbers of nodes (N) and ferries (M)

6.2. Comparison of Algorithms

In this section, we compare the performance of the proposed algorithms in previous sections. Unlike the previous methods, DARD algorithm determines the number of optimal ferries according to the node distribution, thus for each network setting those results having the same number of ferries are evaluated.

Fig. 5 shows the delivery delay for routes computed using the four algorithms. We make the following observations. First, these algorithms achieve similar delay when the number of ferries is small. Second, DARD-I and DARD-II achieve the lowest delay when the number of ferries is large. In contrast, SIRA performs worst due to the fact that the ferries must visit all nodes which significantly increase the length of each route. Third, as a result, by increasing the number of nodes and ferries, delivery delay in both DARD algorithms are considerably reduced as compared with FRA and NRA.

Fig 6 depicts the buffer requirement in the nodes and ferries. Because of the fact that SIRA doesn't use relaying nodes, it requires less buffering in nodes as compared to NRA and DARD. In SIRA, the route must visit all nodes, thus the length of the route is normally much larger than routes in NRA or DARD. With a longer route, data will be kept in ferry buffers for a longer period of time, leading to larger buffer requirements in ferries. Similarly, the buffer requirement for SIRA and NRA is large because of longer route.

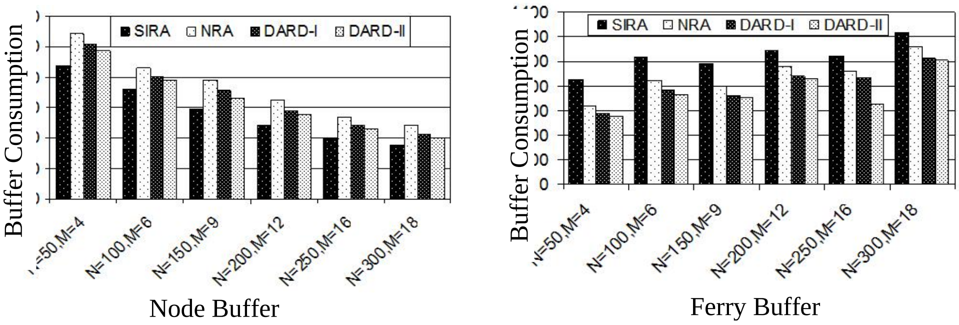
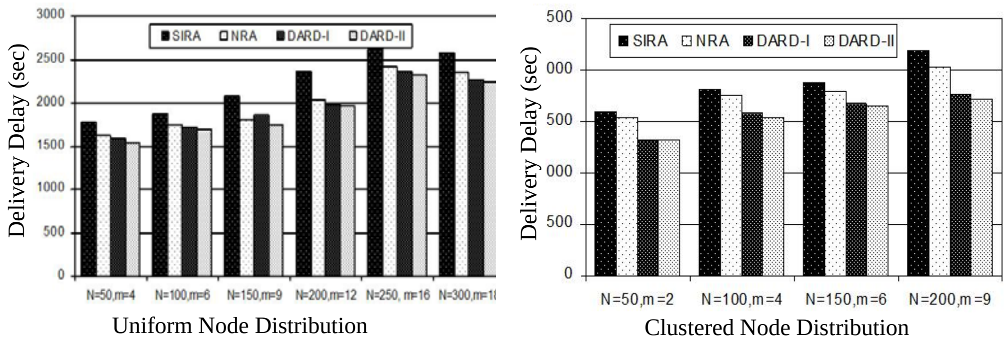


Figure 6. Buffer requirement under different numbers of nodes (N) and ferries (M)



(a) (b)

Figure 7. Message delivery delay under Uniform and Clustered node distribution

Impact of Node Distribution. We now study the performance of the message delivery delay in proposed algorithms under different node distributions and different number of ferries and nodes. Fig 7(a) and Fig 7(b) shows the delay for ad hoc traffic in Uniform and clustered node distribution, respectively. For Uniform node distribution, we can see that the delay increases with the throughput per node.

Impacts of Traffic load. In the following, we have studied the performance of the DARD scheme under different traffic loads. Fig 8 shows the delay for ad hoc traffic. As shown in the Fig 8, the DARD algorithm in normal traffic (upper to 3 messages per second) includes less delay. Also, DARD by creating minimum spanning tree between the clusters designs just one route between the nodes and it losses its efficiency in high traffic as compared to NRA, where there is multiple route between the nodes. It is obvious that SIRA has high delay than the others.

Fig 9 shows the buffer requirements in ferries and nodes under different traffic models. In fact, the increase in buffer requirement and from the longer routes when traffic load is high. This is because the ferries need to spend more time communicating with nodes. When ferry routes become longer, waiting delay for data in nodes is larger. So the node buffer requirements increase. In addition, ferries need to receive more data from nodes in each visit when traffic load is high, requiring more buffers to hold the data.

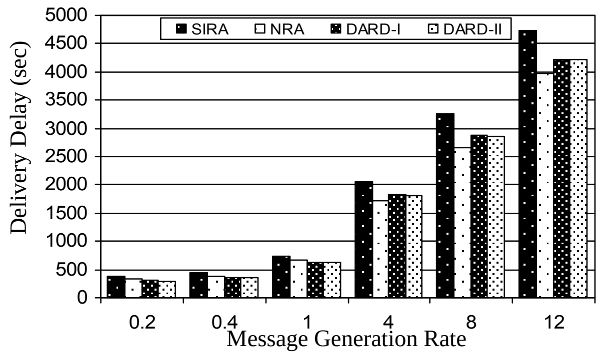


Figure 8. Message Delivery delay under different traffic loads, N=150, M=12

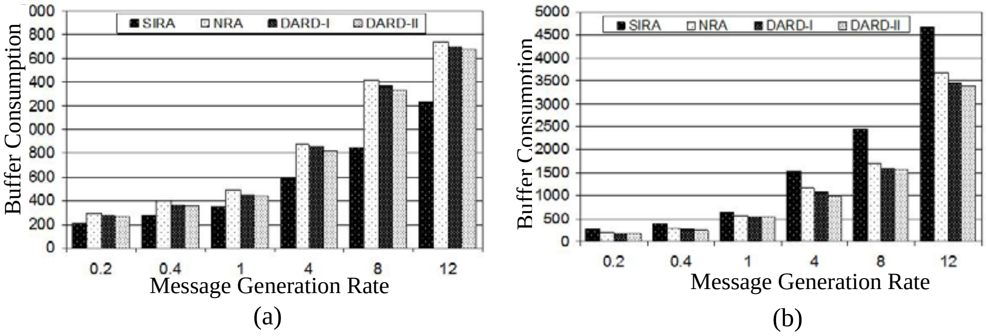


Figure 9. Buffer consumption in (a) nodes and (b) ferries under different traffic load, N = 150, M = 12

Energy Consumption. We now consider the energy consumption for these distinct algorithms. Fig 10 shows the average energy consumption in nodes and ferries, respectively. We can see that in NRA, nodes may need to transmit as much as twice of its own data, which suggests that NRA is not suitable for environments where nodes are constrained in power supplies. Due to the fact that SIRA doesn't use relaying nodes, so less energy is consumed by the nodes. DARD also consumes less energy than NRA. Regarding Fig 10(b), in designed routes by SIRA, the ferries carry more messages which result in increasing the energy consumption. DARD algorithm reduces energy consumption in ferries than NRA by determining the number of optimal ferries and allocating every ferry to the node clusters.

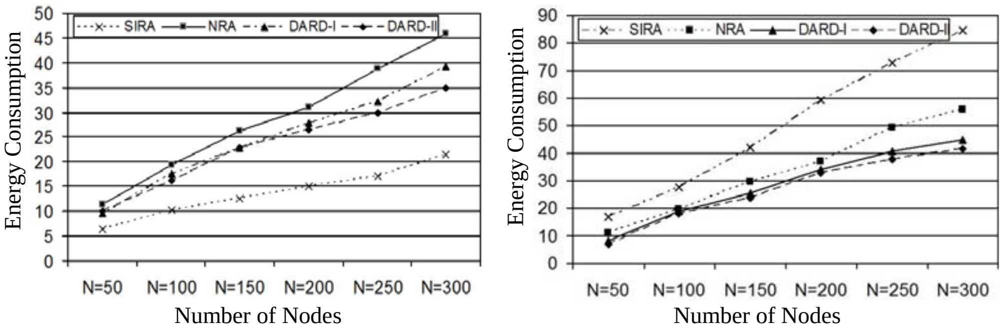


Figure 10. Energy consumption under different number of nodes and 9 ferries (M=9)

7. Conclusions

In this paper, we consider the efficient route design problem for multiple ferries in DTNs. Two designs are proposed, and they represent same trade-off points in the route design space for multiple ferries. According to simulation results, SRT has low buffer requirement in nodes and needs no relay nodes, but it suffers from high delivery delay. Also, in normal traffic, the proposed scheme reduces the message delivery delay and by decreasing buffer consumption in ferries and nodes, it minimizes energy consumption in nodes as well and resulting in increasing ferries efficiency. But, NRA by choosing more relaying nodes than DARD increases the message delivery delay and it also requires more buffer space in ferries and nodes. On the other hand, NRA in high traffic decreases message delay than other algorithms.

References

[1] D.B. Johnson and D.A. Maltz.: Dynamic Source Routing in Ad Hoc Wireless Networks. In Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Chapter 5, Kluwer Academic Publishers, pp. 153-181(1996)

[2] Y.-B. Ko and N.H. Vaidya.: Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM), Dallas, TX, pp. 66-75 (1998)

[3] K. Fall.: A delay-tolerant network architecture for challenged internets. Pages 27 -34, Karlsruhe, Germany, ACM SIGCOMM'03 (2003)

[4] F. Bai and N. Helmy.: Impact of Mobility on Mobility-Assisted Information Diffusion (MAID) Protocols in Ad hoc Networks. IEEE INFOCOM (2005)

[5] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University (2000)

[6] W. Zhao and M. Ammar.: Message Ferrying: Proactive Routing in Highly-Partitioned Wireless Ad Hoc Networks. in Proc. IEEE Workshop on Futrure Trends in Distributed Computing Systems, Puerto Rico(2003)

[7] Z. Huang.: Extensions to the k-Means Algorithms for Clustering Large Data Sets with Categorical Values. Data Mining and Knowledge Discovery, vol. 2, pp. 283-304 (1998)

[8] T. Kohonen.: Self-Organizing Maps. Springer, Berlin, Germany (1997)

[9] D. L. Davies and D. W. Bouldin.: A cluster separation measure. IEEE Trans. Patt. Anal. Machine Intell. vol. PAMI-1, pp. 224–227, (1979)

[10] P. Juang. H. Oki, Y. Wang, M. Martonmi, L. Peh. and D. Rubenstein.: Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet. In ASPWS'O_3 (2002)

[11] A. A. Haqson. R. Fletcher, and A. Pzntland.: DakNet: A road to universal broadband connectivity. wireless Inremet UN ICT Conference Case Study (2003)

[12] Q. Li and D. Rus.: Sending messages to mobile users in disconnected ad-hoc wireless networks. In MOBICOM, (2000)

[13] M. M. B. Tariq, M. Ammar, and E. Zegura.: Message ferry route design for sparse ad hoc networks with mobile nodes. In ACM MobiHoc (2006)

[14] R. Viswanathan, J. Li and M. Chuah.: Message ferrying for constrainted scenarios. Proceedings of the Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks ,WoWMoM'05, (2005)

[15] W. Zhao, M. Ammar, and E. Zegura.: Controlling the Mobility of Multiple Data Transport Ferries in a Delay-Tolerant Network. In Proceedings of IEEE INFOCOMM. (2005)

[16] Z. Zhang and Z. Fei.: Route Design for Multiple Ferries in Delay Tolerant Networks. In Proceedings of IEEE INFOCOMM. (2005)

[17] G. Nemhauser, A. R. Kan, and M. T. (editors). Optimization. (1989)

Authors

Behrouz Jedari was born in Tabriz of Iran. He received his B.S. degree in Software engineering from Azad University of Shabestar, Iran, in 2006, and his M.S. degrees in Software engineering from Azad University of Qazvin, Iran, in 2009. He is a student member of ACIS/IEEE and a member of Young Researchers Club of Tabriz, Iran. His current scientific interests are in general area of Challenged Networks (MANETs,WSNs, DTNs), Knowledge Discovery and Processor Scheduling.

Rouhollah Goudarzi received his B.S. degree in computer engineering from Azad University of Shabestar, Iran, in 2006, and his M.S. degrees from Azad University of Qazvin, Iran, in 2009. His current scientific interests are in Communication and Wireless Sensor Networks.

Mehdi Dehghan received his B.S. degree in computer engineering from Iran University of Science and Technology (IUST), Tehran, Iran, in 1992, and his M.S. and Ph.D. degrees from Amirkabir University of Technology (AUT), Tehran, Iran, in 1995, and 2001, respectively. He is an Assistant Professor of Computer engineering and information technology at Amirkabir University of Technology (AUT). Prior to joining AUT in 2004, he was a Research Scientist at Iran Telecommunication Research Center (ITRC) working in the area of quality-of-service provisioning and network management. His research interests are in wireless networks, pattern recognition, fault-tolerant computing, and distributed systems.