Relay Node Placement in Vehicular Delay-Tolerant Networks

Farid Farahmand[§], Isabella Cerutti[†], Ankitkumar N. Patel[‡], Qiong Zhang[††], and Jason P. Jue[‡]

[§]Department of Computer Electronics and Graphics Technology
Central Connecticut State University, New Britain, CT 06050
[†]Scuola Superiore Sant'Anna, Pisa, Italy
[††]Photonics Networking Laboratory
Fujitsu Laboratories of America, Inc., Richardson, TX 75082
[‡]Department of Computer Science
The University of Texas at Dallas, Richardson, TX 75083
email: farahmandfar@ccsu.edu or isabella.cerutti@sssup.it

***Abstract*— Delay-tolerant networking (DTN) is an architecture to enable data communications between isolated or remote regions, where long delays and intermittent connectivity can be tolerated. An emerging class of DTN, called Vehicular DTNs (VDTN), exploits transportation systems as the transport layer to transfer data. In these networks, vehicles (e.g., busses, boats, trains) act as mobile nodes and carry data messages around. Mobile nodes can exchange data messages using devices called *relay nodes*. Relay nodes, placed in strategic positions along vehicle routes, have the capability to download, store, and upload the data messages from/to the mobile nodes.**

An important issue in VDTN is the optimal placement of the relay nodes such that delay-tolerant connectivity in VDTN is ensured at minimum cost. In this paper we show that the problem of optimal relay node placement is an NP-hard problem. Other contributions of this paper are the formulation of the relay node placement problem using ILP and the proposal of heuristic algorithms solving the optimization problem. Using simulation results, we compare the performance of each algorithm under different network constraints, such as node storage capability and network topology.

***Index Terms*— Delay-Tolerant Network, Routing, Wireless Networks, Transit Networks.**

I. Introduction

Delay/Disruption Tolerant Networks (DTNs) are a class of architectural solutions that have been proposed to address the connectivity challenges due to long transmission delays and intermittent connectivity [1]. The key feature of DTNs lies in their ability to store the data in the network nodes until the connectivity required to forward the data becomes available. In general, a DTN architecture can be characterized by (a) sparse connectivity (i.e., end-to-end route between source and destination may not even exist), (b) long or variable delay, (c) asymmetric data rate, and (d) high error rate.

A particular class of DTN architecture is the vehicle-based or *vehicular DTN (VDTN)*. In these networks, also known as transit networks [2], vehicles (e.g., cars, buses, boats) collect and deliver data between static nodes. A unique application of VDTN is to provide asynchronous Internet access to rural and remote regions, e.g., remote villages, which would be otherwise disconnected from Internet services.

A number of projects have been dedicated to study and address the connectivity challenges in rural regions using a VDTN architecture [6]. For example, motivated by the Daknet project [3], the Rural Internet Kiosk project was deployed to provide a low-cost communication system between rural regions [7]. Another example is the Message Ferry (MF) project that aims at developing a data delivery system in remote villages using mobile nodes called message ferries [4]. The DieselNet testbed, developed at the University of Massachusetts, is another attempt to implement a VDTN [8]- [9].

In order to provide the requested delay-tolerant connectivity, each one of the above projects addresses specific network issues and problems under certain service requests and network assumptions such as topology, node architecture, capabilities, mobility patterns, and available knowledge about the network. For example, the Rural Internet Kiosk project focuses on node hardware and software architecture. In MF, the main idea is to design the mobility pattern in order to improve the performance. In the DieselNet project, the developers focus on improving routing mechanisms.

This paper considers a VDTN architecture consisting of vehicles that act as mobile nodes and carry the data between *terminal nodes* located in isolated regions. The data messages between mobile nodes can only be exchanged via store-and-forward devices, called *relay nodes*. Mobile nodes are capable of dropping and picking up data to/from relay nodes encountered along their routes.

The main contribution of this paper is the optimization of the VDTN design. The optimal VDTN design problem aims at finding the *relay node placement* that minimizes the network cost, while providing full connectivity between users located in remote regions. Network cost is considered proportional to the number of relay nodes that need to be installed to offer delay-tolerant connections. To the best of our knowledge, this is the first time such a study has been conducted in the context of VDTN.

The remainder of this paper is structured as follow. In Section II, the network model and the assumptions are described. In Section III, the relay node placement problem is formulated using an integer linear programming formulation. In Section IV, heuristic algorithms solving the relay node placement problem are presented. Section V shows the performance comparison among the proposed heuristic algorithms under different network

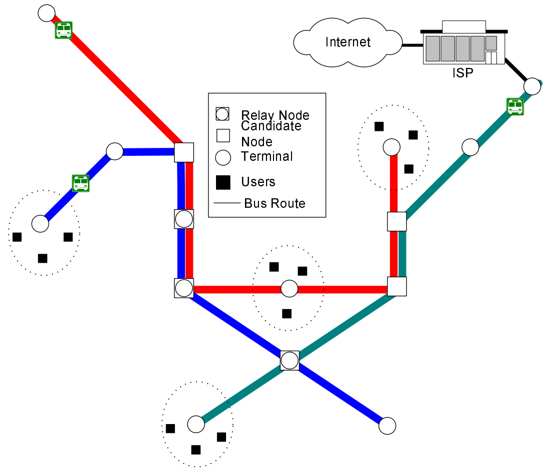


Fig. 1. An example of a VDTN and its components. Isolated regions are circled. A candidate node can be any of the intersecting points or terminal nodes.

constraints, followed by concluding remarks in Section VI.

II. Network Model and Assumptions

The proposed VDTN network utilizes a transportation system as its transport layer to deliver data. Hence, VDTN consists of vehicles (e.g., trains, buses) along predetermined *routes*, as shown in Fig. 1. Each route passes along multiple terminal stations of the transportation system. Two or more routes may intersect at intersection or terminal station.

A *bundle layer* [1] is used as a new layer over the transportation system to aggregate incoming IP data packets into bundle messages, or *messages*, and to provide end-to-end message delivery. Due to the intermittent connectivity and the discrete nature of transfer opportunities in the transportation system, routing large size messages is more suitable than small size IP packets. The bundle layer allows uploading, downloading, and exchanging messages from/to the terminal stations. These operations can take place only when the vehicles approach the terminal nodes. We assume that messages are based on the IP paradigm and the access network connected to the VDTN is an IP network.

Three different types of nodes, namely (a) terminal nodes, (b) relay nodes, and (c) mobile nodes, are deployed in the proposed VDTN architecture.

Terminal nodes are access points to VDTN and they are strategically located at terminal stations to support their surrounding users. Individual end-users are connected to terminal nodes through low-range low-power radio frequency signals, such as the IEEE 802.11 wireless protocol. A key function of terminal nodes is to aggregate the incoming IP data packets and to create messages. Outgoing IP data packets between a terminal node and its surrounding end-users can be routed using commonly known protocols, such as dynamic source routing (DSR) [5].

Mobile nodes are mounted on vehicles (e.g., buses) and act as store-carry-forward devices. It is assumed that message exchange between mobile nodes is not possible due to the short and infrequent contact times among mobile nodes (i.e., the time intervals during which the mobile nodes are within communication range) and the low data-rate that would be reached. In order to store the incoming data messages until successful forwarding, mobile nodes are provided with considerable storage capacity.

Relay nodes are also store-and-forward devices which receive messages from mobile nodes and store them until they can be uploaded on another mobile node. The main function of these nodes is to allow message exchanges between various mobile nodes. Relay nodes are typically stationary and located at terminal nodes shared by different routes, as shown in Fig. 1. Any route intersection is a candidate location for a relay node and hence, it is referred to as *candidate node*. Generally, relay nodes have large storage requirements. Depending on the traffic volume, relay nodes may be equipped with hard drives having several thousand gigabytes of capacity.

The wireless technology, that should be selected to support message exchanges between terminal nodes, relay nodes, and mobile nodes, depends on convergence and economical considerations. For instance, low-powered radio communications based on IEEE 802.11 a/b/g or ultra-wideband (UWB) radio are promising technological solutions for VDTN implementation.

In the next section, the problem of selecting which candidate nodes should act as relay nodes is formalized.

III. Description of the Problem

A VDTN network can be represented as a graph $G(V, E)$. The set of nodes V includes the set of terminal nodes where the traffic is originated, V_t, and the set of terminal nodes where routes intersect (i.e., candidate nodes), V_c. Thus, $V \subseteq V_t \cup V_c$. A directed link i exists from node s to node d, when mobile nodes moving along route i connect s to d, i.e., vehicle route i passes both nodes in sequence. Fig. 2 shows the graph derived for the VDTN network in Fig. 1.

The relay nodes can be placed in any of the candidate nodes in V_c. The relay node placement problem can be defined as follows. Given a VDTN graph and the requested traffic rate (e.g., message rate) between source and destination nodes, the *relay node placement problem* aims at finding the placement of the minimum number of relay nodes, in order to support the requested traffic. The relay placement problem can be described with an integer linear programming formulation, as presented next.

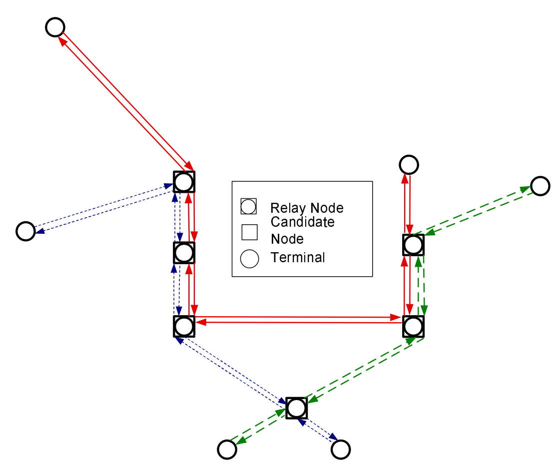


Fig. 2. DTN network graph derived for the transportation system in Fig. 1.

A. Problem Formulation

Given

- $G(V,E)$: VDTN network graph as defined above;
- Λ: the matrix of the traffic requests, Λ_{sd}, expressed in terms of traffic rates, from source node $s \in V_t$ to destination node $d \in V_t$, i.e., $\Lambda \cup_{s,d} \Lambda_{sd}$;
- W: the set of routes in the transportation system;
- π_i: ith route (i.e., sequence of nodes in V);
- C_b: maximum data transportation rate along a route;
- B: maximum transmission bandwidth at a terminal node;
- C_r: maximum storage capacity of a relay node;
- T: large constant, e.g., $T = \sum_{sd} \Lambda_{sd}$.

Variables:

- R_m denotes whether node m is a relay node.

$$R_m = \begin{cases} 1 & \text{if candidate node } m \in V_c \text{ is a relay node} \\ 0 & \text{if } m \in V_t \text{ or if } m \in V_c \text{ is not a relay node;} \end{cases} \tag{1}$$

- $\lambda_{mn}^{sd,i}$ denotes the traffic rate from s to d passing on directed link (m,n) of G, using route i. The end-to-end traffic rate Λ_{sd} may be carried using one route i or multiple routes in parallel or in sequence thanks to the store-and-forward capabilities of relaying nodes.

Objective Function:

$$\text{Minimize: } \sum_{m \in V_c} R_m$$

Constraints:

$$\sum_{i \in W} \sum_{m:(s,m) \in \pi_i} \lambda_{sm}^{sd,i} = \Lambda_{sd} \quad \forall s, d \tag{2}$$

$$\sum_{i \in W} \sum_{m:(m,s) \in \pi_i} \lambda_{ms}^{sd,i} = 0 \quad \forall s, d \tag{3}$$

$$\sum_{i \in W} \sum_{m:(m,d) \in \pi_i} \lambda_{md}^{sd,i} = \Lambda_{sd} \quad \forall s, d \tag{4}$$

$$\sum_{i \in W} \sum_{m:(d,m) \in \pi_i} \lambda_{dm}^{sd,i} = 0 \quad \forall s, d \tag{5}$$

$$\sum_{i \in W} \sum_{m:(m,k) \in \pi_i} \lambda_{mk}^{sd,i} = \sum_{i \in W} \sum_{n:(k,n) \in \pi_i} \lambda_{kn}^{sd,i} \quad \forall s, d \; \forall k \in V \tag{6}$$

$$\sum_{m:(m,k) \in \pi_i} \lambda_{mk}^{sd,i} - \sum_{n:(k,n) \in \pi_i} \lambda_{kn}^{sd,i} \le R_k T \quad \forall s, d \; \forall k \in V \tag{7}$$

Eq. 2-6 are flow conservation constraints, i.e., the requested traffic rate should be routed in the network from s to d, using one or multiple routes and passing through intermediate relay nodes.

Eq. 7 forces to route the traffic on the same route i if node k lacks relaying capabilities. In other words, the combination of Eq. 6 and 7 allows the relaying of traffic on a different routes at relay nodes, while it forces the routing of the traffic on the same route in the intermediate nodes without relaying capabilities.

B. Additional Constraints

The relay placement problem described above can be subject to one or more of the following constraints:

- Constraint on relay node storage capacity. The amount of data stored by relay node m should not be greater than C_r, i.e., $T = C_r$.
- Constraint on route capacity. The amount of data carried by mobile nodes along route i from m to n should not be greater than C_b, i.e.,

$$\sum_{sd} \lambda_{mn}^{sd,i} \le C_b \quad \forall (m,n) \forall i. \tag{8}$$

- Constraint on transmission bandwidth:

$$\sum_{i \in W} \sum_{m:(m,k) \in \pi_i} \lambda_{mk}^{sd,i} \quad \forall m \neq d \in V. \tag{9}$$

In absence of constraints (e.g., when the capacity is large) and when the set of terminal nodes coincides with the set of candidate nodes, the solution of the connected dominating set (CDS) problem [10]–[12] identifies a set of relay locations that assure a feasible route between the source-destination pair of the traffic requests. Finding the smallest CDS on a general graph is NP-hard and thus the complexity of the optimal relay placement problem in the general case is also NP-hard.

When solved, the above presented ILP formulation finds the placement of the minimum number of relay nodes in the DTN. Using this solution, traffic request routing can be further optimized by routing traffic along the minimum latency routes, while meeting the capacity constraints. Under the assumption that message storage time at relay nodes is negligible compared to the transport time of the messages along the routes, the routing can be optimized using a well-known multi-commodity flow formulation that aims at minimizing the average message delivery time (AMDT), i.e., total time spent by a message along the routes.

Next, heuristic algorithms that aim at optimizing the relay node placement and then the average message delivery time are presented.

IV. Heuristic Algorithms

This section presents two heuristic algorithms that attempt to minimize the number of required relay nodes in the network, while guaranteeing end-to-end connectivity between the terminal nodes. Each algorithm aims at minimizing one of the following cost functions subject to minimum number of new relay nodes: (a) minimizing the number of hop counts between source-destination terminal nodes, (b) minimizing the average message delivery time (AMDT) to the destination. We also introduce a heuristic algorithm, which provides an upper bound solution to the relay node placement problem.

A. Minimizing Relay Node and Hop Count (MRH) Algorithm

Minimizing Relay Node and Hop Count (MRH) algorithm maximizes the relay node usage in order to reduce the number of required relay nodes. Hence, no new relay nodes are added unless all existing ones have been fully utilized. The description of the MRH algorithm is given below:

Step 1 : Initialization

a) Given the traffic matrix $\Lambda_{sd} \in \Lambda$ with traffic rates between each $s-d$ node pair, make a copy $Q = \Lambda$.

b) Set $\check{C}_r^i = C_r \; \forall i \in V_c$, where $\check{C}_r^i$ is the available average storage capacity of relay node i.

c) Select the largest traffic request, i.e., $q_{sd}^{max} = \max_{ij} Q_{ij}$.

Step 2 : Find the route (i.e., routing on $G(V,E)$) for q_{sd}^{max} with the *fewest* existing relay nodes.

a) If there are more than one such route use the one with the smallest hop count.

b) If no such route exists, convert another candidate node to a relay node such that a route with the lowest hop count is found.

Step 3 : Verify that the constraint on route capacity is met, i.e., $\check{C}_b^j$ should *not* be greater than its maximum capacity C_b. Verify that the constraint on relay node storage capacity is met, i.e., the total traffic stored by each relay node, $\check{C}_r^i$, should *not* be greater than its maximum capacity, C_r.

Step 4 : Set $q_{sd}^{max} = 0$, update $\check{C}_b^j$ and $\check{C}_r^i$ for all relay nodes and routes utilized along the path. Then return to *Step 2* until all nonzero traffic demands are satisfied.

Step 5 : Count the number of relay nodes carrying traffic.

B. Minimizing Relay Node and Delivery Time (MRD) Algorithm

Minimizing Relay Node and Delivery Time (MRD) algorithm minimizes AMDT while adding the fewest number of new relay nodes. The basic difference between this algorithm and MRH is that, rather than focusing on minimizing the number of relay nodes, messages are routed on the route with smallest AMDT. All steps in MRD algorithm are similar to MRH except *Step 2*, which must be modified as follow:

Step 2 : Find the path on $G(V,E)$ for q_{sd}^{max} requiring the minimum number of new relay nodes.

a) If there are more than one such path, use the one with the smallest hop count.

b) If no such path exists, convert another candidate node to a relay node such that a path with the smallest AMDT is obtained.

C. Minimizing Delivery Time (MDT) Algorithm

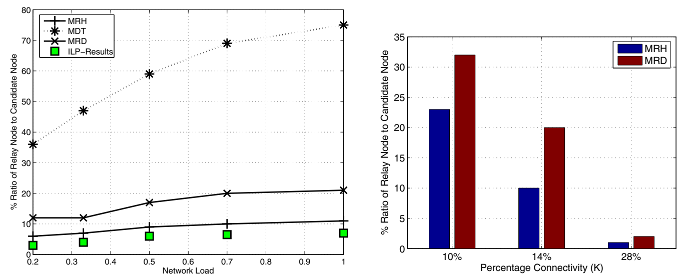
An upper bound solution to the relay node placement problem can be obtained by routing all requests on the route with the smallest AMDT. Hence, Minimizing Delivery Time (MDT) algorithm assumes that all candidate nodes already act as relay nodes. Therefore, routing in MDT results in finding the shortest route between node pairs. If there are more than one such route, the route with the smallest hop count is selected. Once all traffic demands are routed, relay nodes which are not carrying any traffic are removed. We note that MDT provides the lower bound to the average end-to-end message delivery time.

V. Performance Analysis

This section compares the performance of the aforementioned algorithms. Network graphs, as defined in Section III, are randomly generated by uniformly distributing N nodes in a $20 \times 20km^2$ geographical area. The probability of establishing a link between a node pair is uniform. The network connectivity is given by *connectivity factor* (K), indicating the percentage ratio of established links over the number of links in a fully connected network (i.e., $N(N-1)/2$). The number of generated network topologies is sufficient to guarantee that the mean results a confidence interval of 15% or better at 90% confidence level.

For each network topology, a different traffic matrix is generated. Each element in the matrix represents the traffic rate (total number of messages generated in the network per unit time) between a given node pair. Traffic requests are uniformly distributed among all the node pairs. The network load is defined as the total traffic rate offered to the network.

It is assumed that mobile nodes (e.g., buses) move at a speed of 40 km/hr and have a transmission range of 10 m. Also, the average time spent by a message along each given link (including the waiting time and transport time) is known in advance. Unless otherwise indicated, the following assumptions are made: N = 15, K = 14%, unlimited storage capacity at relay nodes and mobile nodes, and unlimited transmission bandwidth.



(a) PRC vs. network load.

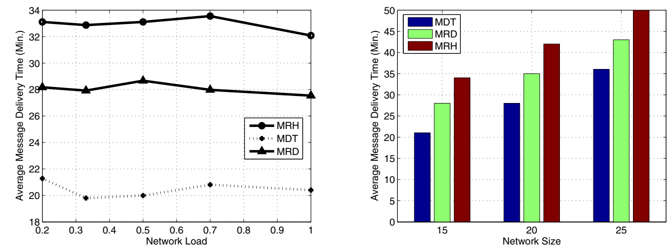
(b) Ratio of relay nodes to candidate nodes for K= 10%, 14%, and 28%, when network load equals 0.8.

Fig. 3. PRC and ratio of relay nodes to candidate nodes.

Fig. 3(a) shows the percentage ratio of relay nodes to candidate nodes (PRC) for each heuristic algorithm versus the network load. When ratio of relay nodes to candidate nodes reaches 100%, each candidate node is selected to be a relay node. This figure suggests that MRH consistently performs better than MRD. This is mainly due to the fact that MRH attempts to find the route with the lowest hop count, resulting in utilizing fewer relay nodes to satisfy all demands. The figure also shows that the results obtained by the ILP and the MDT algorithm represent the lower and upper bounds on the number of relay nodes, respectively.

Fig. 3(b) shows the ratio of relay nodes to candidate nodes for different values of connectivity factor. As K increases (from 10% to 28%), both MRH and MRD algorithms require smaller ratio of relay nodes to candidate nodes. The reason is that the network is more connected and fewer relay nodes are required. The figure suggests that MRH consistently outperforms MRD, regardless of the value of K.

Fig. 4 show the performance of the proposed algorithms in terms of average message delivery time (AMDT) versus the network load and network size, respectively. Note that AMDT also provides an insight into the network throughput, i.e., the amount of messages delivered between nodes in a give time unit. In fact, AMDT is inversely proportional to network throughput.



(a) AMDT vs. network load.

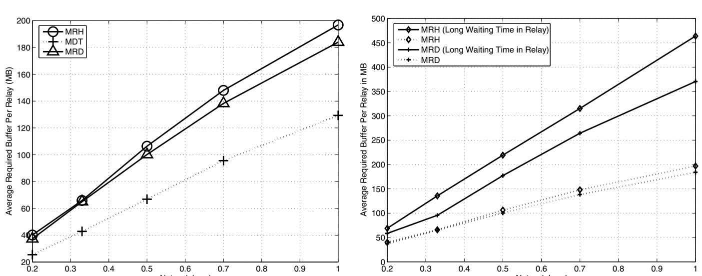
(b) AMDT vs. network size; N changing between 15, 20, and 25.

Fig. 4. AMDT as network load and size varies.

Fig. 4(a) suggests that, in general, as the network load changes, AMDT varies slightly. This is due to the fact that transmission bandwidth is assumed to be infinity. The figure suggests that MRD outperforms MRH in terms of AMDT, while MDT provides the lower bound for AMDT. When the network load reaches its mid-range values (0.5-0.7) AMDT decreases. This counter-intuitive result is due to the fact that as the network load increases, more relay nodes are required to be placed in the network. Once sufficient number of relay nodes are placed, more route options are available, resulting in lower AMDT.

Fig. 4(b) shows AMDT obtained by the algorithms when the total number of nodes in the network increases. The figure suggests that MRD outperforms MRH regardless of the network size.

Fig. 5(a) shows the expected maximum storage capacity required by relay nodes versus the network load. Note that as the network load increases, MRH requires larger buffer capacity. This is mainly due to the fact that the total number of relay nodes placed by MRH algorithm is lower compared to MRD, resulting in large amount of relayed traffic in MRH.



(a) Average maximum buffer size vs. network load.

(b) Expected maximum buffer capacity vs. network load.

Fig. 5. Average maximum buffer size and expected maximum buffer capacity as network load varies.

In general, as the inter-contact time (frequency of vehicles visiting relay nodes) reduces, messages tend to experience longer waiting time at intermediate relay nodes. Fig. 5(b) shows that when inter-contact time becomes longer, MRD outperforms MRH in terms of expected maximum storage capacity at relay nodes; the difference becomes more significant as the network load increases. This is due to the fact that MRH chooses the routes with fewer hops; therefore, messages may experience longer waiting time at relay nodes before being transferred to the mobile nodes.

In summary, MRD and MRH offer different advantages in terms of hardware resources (i.e., storage capacity and number of relay nodes) and message delivery time. While, MRD results in faster message delivery, MRH is more resource efficient.

VI. Conclusion

This paper considered the problem of the optimal relay node placement in vehicular delay tolerant networks (VDTN). The main motivation for this work is the reduction of the network cost in terms of the number of relay nodes required to be installed throughout the network. Two heuristic algorithms aiming at minimizing the number of relay nodes, namely MRH and MRD, were proposed to solve the relay node placement problem. We compared the results obtained from each algorithm and the optimal solution obtained by ILP formulation. The upper bound solution was evaluated using a heuristic that minimizes the delivery time, i.e., MDT. Results obtained on randomly generated VDTN graphs evaluate the algorithm performance in terms of hardware resources (i.e., number and storage capacity of relay nodes) and message delivery time. These results indicate that while MRD results in faster message delivery, MRH is more resource efficient.

In this work we ignored bus storage and transmission bandwidth limitations. Further studies are required to fully understand the benefits of emerging VDTN, in terms of cost effectiveness for the offered performance.

References

[1] V. Cerf et al., “Delay-Tolerant Networking Architecture,” *RFC 4838*, April 2007.

[2] LeBrun, J. Chen-Nee Chuah, D. Ghosal, M. hang, “Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks,” *Vehicular Technology Conference*, 2005.

[3] A. Pentland, R. Fletcher, and A. Hasson, “Daknet: Rethinking connectivity in developing nations,” *IEEE Computer*, Vol. 37, No. 1, pp. 7883, 2004.

[4] W. Zhao and M. H. Ammar, “Message ferrying: proactive routing in highly-partitioned wireless ad hoc networks,” *In proceedings of the IEEE Workshop on Future Trends in Distributed Computing Systems*, Puerto Rico, May 2003.

[5] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” *Mobile Computing*, Kluwer Academic Publisher, Vol. 353, 1996.

[6] Z. Zhang, “Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges,” *IEEE Communications Surveys and Tutorials*, Vol. 8, No. 1, pp. 24-37, 2006.

[7] A. Seth, D. Kroeker, M. Zaharia, S. Guo, S. Keshav, “Low-cost communication for rural internet kiosks using mechanical backhaul,” *Proceedings of the 12th annual international conference*, 2006.

[8] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, “MaxProp: Routing for Vehicle-Based Disruption- Tolerant Networks,” In Proc. *IEEE INFOCOM*, April 2006.

[9] A. Balasubramanian, B. N. Levine, and A. Venkataramani, “DTN Routing as a Resource Allocation Problem,” In Proc. *ACM SIGCOMM*, 2007.

[10] M. S. Savasini, P. Monti, M. Tacca, A. Fumagalli, and H. Waldman, “Regenerator placement with guaranteed connectivity in optical networks,” *ONDM* 2007.

[11] T. Carpenter, D. Shallcross, J. Gannett, J. Jackel, A. Von Lehmen, “Maximizing the transparency advantage in optical networks,” *OFC* 2004.

[12] S. Guha and S. Khuller, “Approximation Algorithms for Connected Dominating Sets,” *Algorithmica*, vol. 20, no. 4, April 1998.