Graph partitioning algorithm for opportunistic routing in large-scale wireless network

LI Yan-hua (✉), LIU Yuan-an

Wireless Communication Center, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract

Opportunistic routing takes advantage of the broadcast nature of wireless communications by forwarding data through a set of opportunistic paths instead of one ‘best’ path in traditional routing. However, using the global scheduling opportunistic scheme like the existing opportunistic routing protocol (ExOR) would consume considerable transmission latency and energy in large-scale wireless topologies. In this article, a graph partitioning algorithm is proposed, namely, minimum cut with laplacians (MCL), to divide the Ad-hoc network topology into subgraphs with minimized edge cuts across them. Then the existing opportunistic routing can be applied locally in each subgraph. In this way, forwarders in different subgraphs can transmit simultaneously, and each node only needs to maintain a local forwarder list instead of a global one. The simulations show that using MCL scheme in the opportunistic routing can reduce the end-to-end delay by about 49%, and increase the life time of the wireless node by about 39%.

Keywords opportunistic routing, graph partitioning, wireless routing, mobile Ad-hoc network

1 Introduction

Opportunistic routing is a new design trend of wireless routing protocol. It is different from the traditional routing such as dynamic source routing (DSR) and Ad-hoc on-demand distance vector routing (AODV) [1–2] in that, instead of just taking one ‘best’ path to the destination, the source takes multiple opportunistic paths to forward the packets in opportunistic routing. The path a packet takes depends on which forwarders happen to receive it, which is non-deterministic. Also, if a certain wireless forwarder fails or moves out of the radio range during the transmission, other possible paths may take over the transmissions. As a result, opportunistic routing can better cope with lossy, unreliable and varying link qualities typical of wireless networks.

ExOR [3] is one primary opportunistic routing protocol in previous works. It constructs a forwarder list for each source destination pair, in which all of the forwarders are prioritized. At the beginning, the source broadcasts a packet that carries the forwarder list information in the packet header. Many forwarders may hear this transmission. The forwarder with the highest priority that received the packet would forward it. All of the other forwarders in the forwarder list have to wait for their higher-priority nodes’ transmissions, and thus the packet would always be forwarded by the nearest node to its destination.

There are several important issues in designing the opportunistic routing scheduling scheme. For example, how should one schedule the transmissions between different forwarders? Is it necessary for a forwarder to wait for a forwarder far away to transmit? Actually, in a large-scale topology, a pair of nodes may have a long forwarder list. Many of the forwarders are not within the same radio range. It would be more efficient, if they can transmit simultaneously.

Dubois-Ferriere et al. [4] introduces a specific cost function defined with respect to a set of candidate forwarders, and proposes the least-cost opportunistic routing (LCOR) algorithm to identify the best candidate set that minimizes the said cost function. Due to its potentially exponential time complexity, heuristic policies have to be incorporated in the LCOR. In Ref. [5], the end-to-end throughput or capacity of opportunistic routing is studied in multi-rate wireless
networks using a linear programming framework. Based on these works, an optimal forwarder list selection scheme is proposed in Ref. [6], which can minimize the expected number of transmissions needed to opportunistically deliver a packet between two nodes.

This article analyzes the scheduling problem and proposes a local scheduling scheme with graph partitioning method. The graph partition algorithm for partitioning wireless topologies is derived in Sect. 2. In Sect. 3, the performances with and without MCL algorithm in ExOR routing protocol are evaluated and compared by simulations. Finally, conclusions are presented in Sect. 4.

2 MCL

In ExOR, it is assumed that a global prioritized forwarder list is used for opportunistic packet forwarding, and the same (batch mode-based, hop-by-hop, prioritized) packet forwarding mechanisms are employed, as described below. Let \( \{s, u_1, u_2, \ldots, u_d\} \) be an (ordered) forwarder list, where \( s \) is the source, \( d \) is the destination, and \( u_1, u_2, \ldots, u_d \) are intermediate forwarders. The nodes are ordered based on their increasing priorities from left to right: the destination node \( d \) has the highest priority and \( s \) the lowest priority. When \( 1 \leq i < j \leq m \) holds, \( u_i \) has higher priority than \( u_j \). If \( m = 0 \), no intermediate forwarder is used. The priority of the node in the forwarder list is important in coordinating the packet forwarding in the (prioritized) packet forwarding mechanisms used in ExOR. Whereas, in such a way, only one node can transmit at one time, while some other nodes are even not within the same radio range. Particularly, in a large-scale topology, the following problems may arise.

1) Spatial reuse: a large forwarder list will be constructed for a faraway node pair in a large-scale network. Only one of the forwarders in the list can transmit at one time, and others have to wait for a certain back-off time based on the priority they have. Even two forwarders are not in the same radio range, and they cannot transmit simultaneously. Obviously, this would result in heavy delay from the source to the destination.

2) Computation cost: in a large-scale topology, each wireless node has limited energy. The energy consumption at each node includes the computation cost for control information and the transmission cost. In ExOR, the cost for computing the global forwarder list constitutes a big part. The graph partitioning algorithm will be used to divide the wireless topology into small pieces, so that each node just needs to compute the local forwarder list for its subgraph. This would dramatically reduce the energy cost for computing control information and improve the life time of each wireless node.

It has already been known that nodes within the same radio range cannot transmit simultaneously as a result of confliction. The global waiting scheme in ExOR would cost plenty of time to wait for remote nodes’ transmissions, even though they are not within the same radio range. Hence, ExOR protocol would cause higher latency in the large-scale wireless network, in which the source and the destination may be several radio ranges away from each other. In this article, the authors focus on the mobile Ad-hoc multihop networks, though it is believed that the general methodology is applicable to other scenarios. They will design a local scheduling scheme with graph partitioning for this kind of large-scale multihop topologies below.

To solve these problems, one can decompose a large network into several subgraphs. Then, by taking each subgraph as an equivalent node, the original topology is transferred to another graph with fewer ‘nodes’. A new equivalent ‘forwarder list’ would be constructed with fewer ‘forwarders’. Each ‘forwarder’ is actually a subgraph. In each local subgraph, the real nodes are scheduled based on the ExOR opportunistic routing. In the new equivalent graph, the traditional routing is used to find the best ‘path’ for the source destination pair, and each ‘node’ on the path is a subset of real nodes. For example, in Fig. 1(a), the original topology is partitioned into 8 subgraphs, then the topology is transferred that shown in Fig. 1(b). It can be easily found in the best path from \( S \) to \( D \) as \( S \rightarrow C_1 \rightarrow C_i \rightarrow C_s \rightarrow D \). In each subgraph, opportunistic routing is applied to schedule the local transmissions among different forwarders. Obviously, nodes in different subsets would be able to transmit at the same time. The cost for computing the best path in equivalent topology and local forwarder lists for each subset would be less than that for computing the global forwarder list.

With the graph partitioning idea, the authors attempt to use the newly transferred graph (in Fig. 1(b)) to approximate to the original topology (in Fig. 1(a)). It might bring inaccuracy for forwarder list selection. Obviously, the better the connectivity in each sub-topology is, the better the approximation will be. The goal is to partition a graph into \( k \) disjoint sets \( C_1, C_2, \ldots, C_k \), with the maximized sum of the link qualities within all of the subgraphs, such that the connectivity within the subgraphs is optimized. In other words, the authors try to minimize the sum of the link qualities across different subgraphs.
are defined by 

\[ \sum_{i,j=1}^{n} p_{ij} x(i)x(j) \]

\[ = \frac{1}{2} \left[ \sum_{i=1}^{n} d_i x(i)^2 - 2 \sum_{i,j}^{n} p_{ij} x(i)x(j) + \sum_{j=1}^{n} d_j x(j)^2 \right] = \frac{1}{2} \left[ \sum_{i,j}^{n} p_{ij} x(i)x(j) \right] = \frac{1}{2} \sum_{i,j=1}^{n} p_{ij} [x(i) - x(j)]^2 \]

Proof By the definition of \( d_i \) in Eq. (1), one has

\[ x^T L x = x^T D x - x^T P x = \sum_{i=1}^{n} d_i x(i)^2 - \sum_{i,j}^{n} p_{ij} x(i)x(j) \]

Note that, in fact, the sum only runs over all vertices adjacent to \( v_j \), and for all other vertices \( v_j \) the weight \( p_{i,j} \)

is 0. The degree matrix \( D \) is defined as the diagonal matrix with the degrees \( d_1, d_2, ..., d_n \) on the diagonal. Given a subset of vertices \( C \subseteq V \), its complement \( V \setminus C \) is denoted by \( \overline{C} \).

The indicator vector \( x_C := [x(1), x(2), ..., x(n)]^T \in \mathbb{R}^n \) is defined as the vector with entries \( x(i) = 1 \), if \( v_i \in C \) and \( x(i) = 0 \), otherwise. \( |C| \) is used to measure the ‘size’ of a subset \( C \subseteq V \) by the number of vertices.

2.1 The graph laplacians

The graph laplacians matrix is defined as

\[ L := D - P \]

It has one important property as below:

**Lemma 1** [Property of normalized Laplacians \( L \)]

For every \( x = [x(1), x(2), ..., x(n)]^T \in \mathbb{R}^n \), one has

\[ x^T L x = \frac{1}{2} \sum_{i,j=1}^{n} p_{ij} [x(i) - x(j)]^2 \]

2.2 Minimizing the cut of partition

The above-mentioned problem can be restated as follows: one wants to find the way of partitioning the graph such that the edges between different groups have a very low weight (which means that points in different clusters are dissimilar from each other). This section will show how spectral clustering can be derived as an approximation to such graph partitioning problems. Given the partition of \( V \) into \( k \) disjoint subset \( C_1, C_2, ..., C_k \subseteq V \), \( k \leq n \), \( k \) indicator vectors \( x_m := [x_m(1), x_m(2), ..., x_m(n)]^T \), \( m = 1, 2, ..., k \) are defined by

\[ x_m(j) := \begin{cases} \frac{1}{\sqrt{|C_m|}}, & \text{if } v_j \in C_m \\ 0, & \text{otherwise} \end{cases} \]

One has:

\[ \text{cut}(C_1, C_2, ..., C_k) := \sum_{m=1}^{k} \sum_{v \in C_m \setminus \overline{C_m}} p_{ij} \]

Then the matrix \( X \in \mathbb{R}^{n \times k} \) is set as the matrix containing those \( k \) indicator vectors as columns. Note that the columns in \( X \) are orthonormal to each other, that is, \( X^T X = I \). The following can then be obtained:

\[ x_m^T L x_m = \frac{1}{2} \sum_{i,j=1}^{n} p_{ij} [x_m(i) - x_m(j)]^2 \]

\[ = \frac{1}{2} \left[ \sum_{v \in C_m \setminus \overline{C_m}} \frac{P_{ij}}{|C_m|} + \sum_{v \in \overline{C_m} \setminus C_m} \frac{P_{ij}}{|C_m|} \right] = \frac{|V|}{2} \sum_{v \in \overline{C_m} \setminus C_m} \frac{P_{ij}}{|C_m|} \]
where $\text{tr}$ denotes the trace of a matrix. Thus, the problem of under loosened conditions becomes:

$$\min_{C_1, C_2, \ldots, C_k} \text{tr}(X^T L X)$$

subject to $X^T L X = I$, $X$ as in Eq. (4).

For further exploring the problem, the entries of matrix $X$ are allowed to take arbitrary real values. Then, the problem under loosened conditions becomes:

$$\min_{X \in \mathbb{R}^{n \times k}} \text{tr}(X^T L X) \quad \text{s.t.} \quad X^T L X = I$$

This is the standard form of a trace minimization problem, and the Rayleigh-Ritz theorem [8] indicates that the solution is given by choosing $X$ as the matrix that contains the first $k$ eigenvectors corresponding to the $k$ smallest eigenvalues of $L$ as columns. Now, one needs to reconvert the real valued solution matrix to a discrete partition. Standard algorithms [9] are used on the rows of $X$ and the clusters $C_1, C_2, \ldots, C_k$ are obtained.

The authors will predetermine the number of subgraphs $k$ for a topology of $n$ nodes. Let $N_{\text{ngb}}(v_i)$ denote the number of neighbors of the node $v_i$ and $\bar{N}_{\text{ngb}} = \frac{\sum_{i=1}^{n} N_{\text{ngb}}(v_i)}{n}$ the average number of neighbors each node has. Thus the authors use

$$k = n \left( 1 - \frac{1}{ar{N}_{\text{ngb}}} \right)$$

as the number of subgraphs they want to have.

Based on the derivations above, the graph partition algorithm is derived as below:

1. Construct the one-hop probability matrix $P$ for the topology $G$.
2. Compute the graph laplacian $L = D - P$.
3. Compute the first $k$ eigenvectors $u_1, u_2, \ldots, u_k$ of $L$; $k$ is predetermined in Eq. (9).
4. Let the indicator matrix $X$ be the matrix containing the vectors $u_1, u_2, \ldots, u_k$ as columns.
5. Use the $k$-means algorithm on the rows of $X$, and cluster $V$ into $C_1, C_2, \ldots, C_k$.

The MCL algorithm is employed to partition the wireless topology into subgraphs, and ExOR is applied to forward the packet in each subgraph. In this way, one can dramatically improve end-to-end delay and reduce energy cost.

### 3 Performance evaluation and comparison

Extensive simulations are conducted in ns2 [10] to evaluate the performance of the proposed MCL algorithm in comparison to that of the original ExOR protocol in mobile Ad-hoc network. The simulation parameters are listed in Table 1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Batch sizes</th>
<th>Bandwidth</th>
<th>Transmission protocol</th>
<th>Routing protocol</th>
<th>MAC layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting</td>
<td>100</td>
<td>3 Mbit/s</td>
<td>UDP</td>
<td>ExOR</td>
<td>802.11b</td>
</tr>
</tbody>
</table>

The authors tested these two schemes in 6 sets of topologies; each is $1500 \times 1500$ m with different numbers of randomly positioned nodes. The amount of nodes in different topologies sets is 20, 40, 80, 120, 160 and 200, respectively. The radio radius of each node is 300 m. For each topology, the authors randomly generated 20 topologies of different link qualities among the nodes. 10 simulations are conducted with different random seeds for each topology. User datagram protocol (UDP) is used for real-time traffic. Real-time constant bit rate (CBR) flow with 512 B packets is employed in the simulation. The sources and destinations of all flows are randomly selected.

Comparison of performances in end-to-end delay is displayed in Fig. 2. It can be seen that the end-to-end delay grows with the increase of distance (hop counts) between node pairs. The MCL scheme can always obtain lower end-to-end delay than original ExOR scheme for all of the node pairs. The MCL scheme can always obtain lower end-to-end delay than original ExOR scheme for all of the node pairs. And the MCL will gain more benefits than original ExOR when the hop count between node pairs is larger. This is because remote node pairs would have a larger global forwarder lists based on the ExOR scheme. Each node would wait longer to get chance of forwarding. The MCL scheme partitions the topology into subgraphs, and thus nodes in different subgraphs can transmit simultaneously. This would dramatically decrease the end-to-end delay in a large-scale wireless network. It can also be seen that when the hop counts between node pairs are lower (than 10), the performance of the MCL is not much different from the original ExOR. This is because in this kind of topologies, most of the nodes are within the same radio range, thus there is no need for partition or one just needs to partition the graph with smaller $k$. As illustrated in Fig. 2, the largest gain exceeds that of the...
original ExOR by about 49%. In fact, the improvement will be more when the distance between the node pairs is longer.

Then, the life time of wireless nodes is compared in Fig. 3. First, it can be seen that the life time would be longer when the topology size is smaller. This is because in this case the cost for computing the forwarder list and collecting the global information would be less under both of these two schemes. Also, this figure shows that the nodes using the MCL scheme can always have longer life time than that uses the original ExOR scheme. This is because each node with the MCL just needs to do the opportunistic routing locally, which saves considerable energy for the nodes. It is also shown in Fig. 2 that the benefits would be more as the size of topology increases, because more tasks would be taken over locally by the MCL scheme in this case.

4 Conclusions

In this article, a graph partitioning algorithm is proposed, that is, the MCL algorithm for opportunistic routing protocol. It partitions a large-scale topology into $k$ subgraphs and schedules the transmissions in each subgraph. This algorithm enables different forwarders to transmit their data simultaneously in opportunistic routing. In such a way, the end-to-end delay from the source to the destination schedules is significantly improved. The simulations with ns2 demonstrate that the MCL scheme improves performance in end-to-end delay by about 49% and about 39% in node life time in comparison to that of the original ExOR. As the topology size increases, the gain would be much larger.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (60573111), the Project for Supporting Excellent Scholars in the New Century, and the Ministry of Education of China (NCET-04-0112).

References