Local clustering for hierarchical ad hoc networks

Satu Elisa Virtanen[1] and Pekka Nikander[2]

[1] Helsinki University of Technology, 02150 Espoo, Finland
satu@tcs.hut.fi
[2] Ericsson Research Nomadiclab, 02420 Jorvas, Finland
pekka.nikander@nomadiclab.com

Abstract. Hierarchical, cluster-based routing greatly reduces routing table sizes compared to host-based routing and the amount of routing related signalling traffic, while reducing path efficiency [8] and generating some management traffic [5], [10], [13]. Proposals for and analysis of cluster-based routing in dynamic networks include [9] and [13]. We present a local clustering method that requires no extra messages, since the required messages are simple enough to be piggybacked on link layer or routing messages. Based on our initial simulations, the method produces intuitively good clusters, thereby minimizing address changes and allowing us to optimize routing traffic.

1 Introduction

When clustering is introduced to an ad hoc routing system, locally computable clustering is a necessity to avoid generation of further control traffic. In the ideal case, each arriving node determines an appropriate cluster by consulting its immediate neighbors, who will not need to communicate further. Clusters can be based on node identifiers or broadcasting formation messages [1], [3]. We propose a cluster selection method that uses the link structure, combining local and relative densities [4], [11]. The method is especially useful when used with link state routing algorithms, such as OLSR [2], which require dense and relatively small networks in order to be efficient [12]. For stable clusters, this gives very good performance. For inter-cluster routing, one may use on-demand routing protocols.

2 Cluster formation and management

We represent an ad hoc network as a dynamic undirected graph, consisting of nodes and edges. In a graph $\mathcal{G} = (V, E)$, a cluster candidate is a set of nodes $\mathcal{C} \subseteq V$, and the set of edges of the subgraph induced by $\mathcal{C}$ is $\mathcal{E} = \{(u, v) \in E \mid u, v \in \mathcal{C}\}$. The *size* of the cluster is the number of nodes included in the cluster, denoted by $|\mathcal{C}|$. The local *density* of a cluster $\mathcal{C}$ is simply $\delta_\ell(\mathcal{C}) = |\mathcal{E}|/\binom{|\mathcal{C}|}{2}$; clusters for which $\delta_\ell(\mathcal{C}) \gg |E|/\binom{|V|}{2}$ can be considered good. The *relative density* of a cluster $\mathcal{C}$ is

$$\delta_r(\mathcal{C}) = \frac{\deg_{\text{int}}(\mathcal{C})}{\deg_{\text{int}}(\mathcal{C}) + \deg_{\text{ext}}(\mathcal{C})}, \text{ where } \begin{cases} \deg_{\text{int}}(\mathcal{C}) = |\mathcal{E}| \ = \ |\{(u, v) \in E \mid u, v \in \mathcal{C}\}|, \\ \deg_{\text{ext}}(\mathcal{C}) = |\{(u, v) \in E \mid u \in \mathcal{C}, v \in V \setminus \mathcal{C}\}|. \end{cases} \tag{1}$$

It is commonly acknowledged that a good graph cluster should have many edges connecting the included nodes to each other, and as few as possible connecting the cluster to the rest of the graph, and hence, high relative density [7], [11]. Hence we want our clusters to be connected, dense, and introvert [6]:

- all of the included nodes are connected to each other by at least one path *within* the cluster,
- each node in the cluster is connected by an edge to many of the cluster members, and
- the cluster members have only few edges pointing to non-members.

The first criterion is met by only considering connected subgraphs; using the below combination of the relative and local densities as a cluster quality measure fulfills the rest [14]:

$$f(\mathcal{C}) = \delta_\ell(\mathcal{C}) \cdot \delta_r(\mathcal{C}) = \frac{2 \deg_{\text{int}}(\mathcal{C})^2}{|\mathcal{C}| \, (|\mathcal{C}| - 1)(\deg_{\text{int}}(\mathcal{C}) + \deg_{\text{ext}}(\mathcal{C}))}. \tag{2}$$

Upon arriving to a new location or waking up from sleep, a new node probes its neighborhood. All nodes that hear the probe, if any, send their cluster identifier together with three integer values: (i) the size $|\mathcal{C}|$, (ii) the internal degree $\deg_{\text{int}}$, and (iii) the external degree $\deg_{\text{ext}}$ of their cluster. These values could be easily embedded into existing messages. Based on the messages received, the arriving node builds its neighbor list and calculates the fitness $f(\mathcal{C})$ that each of the neighboring clusters would obtain if it were to join it. As the node can deduce the current fitness of each of the clusters, it joins the one which "gains" the most (or "suffers" the least) due to the arrival of the new member by sending a join message (typically, a routing protocol message, e.g., an OLSR HELLO [2]). If a node receives no answers to the probe, it starts its own cluster.

Parameters influencing cluster formation, such as a maximum cluster size, can be set to regulate the amount of routing traffic in link-state based routing schemes. Cluster memberships can be updated, for example, at regular intervals or upon the creation or loss of connections. The detection of a new connection should cause a node to change its cluster *only if* the new cluster arrangement is significantly better than the current one. This reduces the amount of routing and address management traffic.

In order to handle cluster fragmentation, one node in each cluster needs to hold *cluster head* status. A node that has no intra-cluster path remaining to its cluster head re-initiates the selection protocol, thereby alerting its neighbors to check whether they are still proper cluster members. If a full link state protocol is used within a cluster, routing information allows trivial partitioning detection. When a hierarchy of clusters is desired, the cluster heads may perform the same type of clustering on the meta-network formed by considering each base-cluster as a single node, and with multiple links between pairs of clusters considered as single links.

3 Evaluation

Based on our simulations, the clusters formed using the above method achieve locality in space and their structure mostly corresponds well to the intuitive global clusterings of the network. One can also approximate $f(\mathcal{C})$ using estimates of the required values $|\mathcal{C}|$, $\deg_{\mathrm{int}}$, and $\deg_{\mathrm{ext}}$. This would allow for a more relaxed control traffic within a cluster, as not all nodes need to be immediately aware of newcomers, departing nodes, or changes in edges. If a link state routing protocol is used within a cluster, its nodes can use link state information to produce the values, and do not need to exchange any extra messages. With the three values and information about new or deleted edges, each node is able to estimate the cluster quality for each candidate cluster. For moderately sized clusters (at most 256 nodes) and 64-bit long cluster identifiers, all of the required information can be fit into 16 bytes. This can be easily included in existing link layer frames, IP layer address resolution or neighbor discovery messages, or routing messages. For example, the information could be carried in Wireless LAN beacon frames or in IPv6 Neighbor Discovery messages.

Acknowledgments The first author was supported by the Academy of Finland (under grants 202205 and 206235) and the Nokia Foundation.

References

1. Z. Cai, M. Lu, and X. Wang. Channel access-based self-organized clustering in ad hoc networks. *IEEE Trans. Mobile Comput.*, 2(2), 2003.
2. T. Clausen and P. Jacquet. Optimized link state routing protocol (OLSR). Tech. Report RFC3626, IETF, Reston, VA, USA, 2003.
3. T.-C. Hou and T.-J. Tsai. An access-based clustering protocol for multihop wireless ad hoc networks. *IEEE J-SAC*, 19(7):1201–1210, 2001.
4. A. Jain, M. Murty, and P. Flynn. Data clustering: a review. *ACM Comp. Surv.*, 31(3):264–323, 1999.
5. F. Kamoun and L. Kleinrock. Stochastic performance evaluation of hierarchical routing for large networks. *Comput. Network*, 3:337–353, 1979.
6. R. Kannan, S. Vempala, and A. Vetta. On clusterings — good, bad and spectral. In *Proc. FOCS*, pp. 367–377, Los Alamitos, CA, USA, 2000. IEEE.
7. J. Kleinberg and S. Lawrence. The structure of the web. *Science*, 294(5548):1849–1850, 2001.
8. L. Kleinrock and F. Kamoun. Hierarchical routing for large networks: Performance evaluation and optimization. *Comput. Network*, 1:155–174, 1977.
9. P. Krishna, N. Vaidya, M. Chatterjee, and D. Pradhan. A cluster-based approach for routing in dynamic networks. *Comput. Comm. Rev.*, 27(2):49–64, 1997.
10. G. S. Lauer. Hierarchical routing design for SURAN. In *Proc. ICC*, pp. 93–102. IEEE, 1986.
11. M. Mihail, C. Gkantsidis, A. Saberi, and E. Zegura. On the semantics of Internet topologies. Tech. Report GIT-CC-02-07, CC, Georgia Institute of Technology, Atlanta, GA, USA, 2002.
12. C. Santivanez, R. Ramanathan, and I. Stavrakakis. Making link-state routing scale for ad hoc networks. In *Proc. MobiHOC*, pp. 22–32, Long Beach, CA, USA, 2001. ACM Press.
13. J. Sucec and I. Marsic. Clustering overhead for hierarchical routing in mobile Ad hoc networks. In *Proc. INFOCOM*, vol. 3, pp. 1698–1706, Los Alamitos, CA, USA, 2002. IEEE.
14. S. Virtanen. Properties of nonuniform random graph models. Research Report A77, Helsinki University of Technology, TCS, Espoo, Finland, 2003.