3A-02

A Research on the Self-optimizing Distributed Algorithm Constructing Rectilinear Steiner Tree in Virtual Grid Networks

Yonghwan Kim† Toshimitsu Masuzawa‡

†Graduate School of Engineering, Nagoya Institute of Technology, Aichi, Japan
‡Graduate School of Information and Technology, Osaka University, Osaka, Japan

1 Introduction

Recently, wireless networks, such as MANETs or Sensor Networks, become popular and important in the distributed systems. In the wireless networks, each node can directly communicate only with nodes within its communication range. If the destination node is outside of the communication range of the source node, the message should be relayed to the destination node. The topology of wireless networks can be changed frequently because of the mobility of a node, node failures, or resource scarcity. Therefore, the key issues of the wireless routing protocols contain adaptability to the network dynamics and reduction of resource consumption.

We suppose one special node (named a *home* node) and several moving nodes (named *target* nodes) in virtual grid networks, which is obtained by virtually dividing a wireless network into a grid of geographical square regions called *cells* of the same size. A single node is selected as a *router* at each cell and inter-cell communication is realized by using the routers. And we assume that the set of paths, which consists of each path to each target node from a home node, are given.

In this paper, we consider maintenance of inter-cell communication paths to each target node from the node. As we mentioned above, the power consumption is one of the important issue in wireless sensor networks, therefore, we consider that our goal is the construction of a rectilinear steiner tree (means the steiner tree on the grid topology, RST) which connects a home node to all target nodes. Steiner tree ensures that the total number of edges between routers from its definition. However, construction of RST in grid networks is known as *NP-hard*[2] problem. This implies that the designing of an distributed algorithm, which can find an optimal solution using local information only, is nearly impossible. Thus, in this paper, we firstly introduce the optimizing protocol with only three nodes, two target nodes and one home node, in a virtual grid network. Moreover, we briefly discuss an optimizing protocol which constructs RST for 4 or more nodes in virtual grid networks.

2 3-nodes RST Protocol

We adopt a routing protocol named *Zigzag* to make a shortest path to each target node from a home node, and we propose some new protocols to combine some parts of two paths for reduction the number of edges contained in RST.

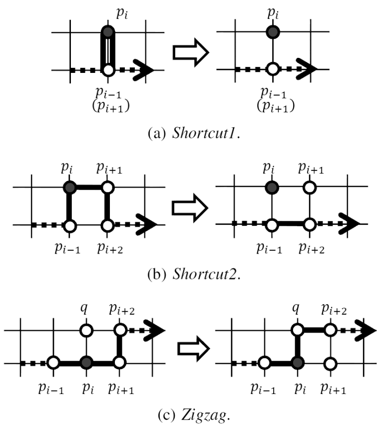

Figure 1: Three local updates to a path in *Zigzag*

2.1 A routing protocol *Zigzag* and its modification

Zigzag is a local-information-based self-optimizing routing protocol in virtual grid networks[1]. Protocol *Zigzag* find a shortest path between two nodes by repeatedly applying local updates to the path until it converges to a shortest path. *Zigzag* detects a redundancy of the recent path locally with making *zigzag-based* path. *Zigzag* defines only three local updates on the node p_i as Fig. 1.

We consider the combining of two paths which are transformed (or been transforming) by *Zigzag*. However, the converged shortest path can be different depending on the initial path, even if the positional relation between two nodes is exactly same. Therefore, we modify the protocol *Zigzag* slightly. If a home node finds some specific starting directions, a home node updates those directions. Table 1 shows the detail rules of starting directions.

Table 1: Rule for fixing the starting directions

2-hop directions from home	modified directions
{UP, RIGHT}(1st quadrant)	{RIGHT, UP}
{LEFT, UP}(2nd quadrant)	{UP, LEFT}
{DOWN, LEFT}(3rd quadrant)	{LEFT DOWN}
{RIGHT, DOWN}(4th quadrant)	{DOWN, RIGHT}

This modification makes it an open possibility to combine two paths. We introduce our protocol in the Section 2.2.

Copyright ©2016 Information Processing Society of Japan.
All Rights Reserved.

2.2 Path Combining Protocol

In this Section, we introduce our protocol to reduce the number of edges combining two paths. In the previous Section, we modified the protocol *Zigzag* thus we can easily find the positional relationship between two paths. We explain how to combine two paths in detail. To easy to understand, first of all, we consider the case when two target nodes are in the neighboring quadrants (e.g., one target node is in the 1st quadrant, and another one is in the 2nd quadrant).

Fig. 2(a) shows an example case when two target nodes are in the neighboring quadrants, 1st and 2nd quadrants. In this case, we can find *U-shaped* path, from $(0, 1)$ to $(1, 1)$, on a home node (the origin). From this U-shaped path, a home node recognizes that two target nodes are in the neighboring quadrants and two paths can be combined. Note that, this recognition might be incorrect because *Zigzag* is not converged yet, but a home node can decide it at the current moment and our protocol can resolve this local miss. Fig. 2(b) presents the situation after combining detected U-shaped path. Total number of edges is less than Fig. 2(a), however we cannnot find more combining points although this is not optimal solution. Therefore, we introduce a new virtual node named *virtual home* node (*vHome*). *vHome* is located on the home node initially, and after combining two path on U-shaped path as Fig. 2(b), *vHome* moves one hop along the combined path. In the case of Fig. 2(b), *vHome* is located on $(0, 1)$. *vHome* operates exactly same as a home node, this changes the *Zigzag* paths on Fig. 2(b). The path to left target changes its starting directions {UP, LEFT} due to the modification of *Zigzag* (Section 2.1). The zigzag part of the path to right target moves left by *Zigzag* protocol.

Fig. 3(a) illustrates the path after *vHome* is located on $(0, 1)$. Because of local updates of *Zigzag* protocol and its modification, our protocol can find the combining point again. A virtual home moves repeatedly and this protocol can be eventually converged as Fig. 3(b).

Our protocol using *vHome* basically updates the path, if it finds U-shaped path. However, we cannnot know whether a protocol *Zigzag* is converged or not using local information only. As we mentioned, our protocol's local update (combining) is sometimes incorrect. However, our protocol allows not only moving forward but also moving backward of *vHome* when it finds U-shaped path.

Finally we introduce one more simple rule to our protocol. By our protocol, the path to *vHome* from a home node is created. When *vHome* is not on the origin, if the direction just after *vHome* is the reverse direction of one just before *vHome*, *vHome* moves backward with one hop because the last hop of common path is redundant trivially. Fig. 4 summarizes the local update's rule of our protocol.

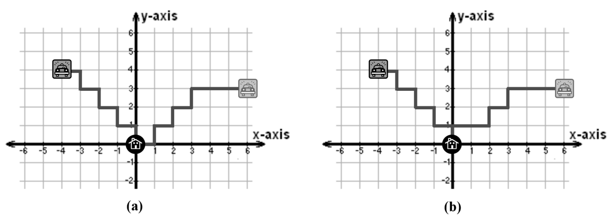

Figure 2: An example case of neighboring quadrants

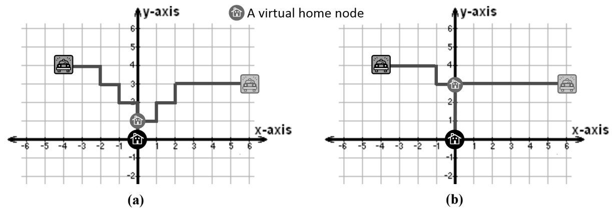

Figure 3: Coordinating using a virtual home

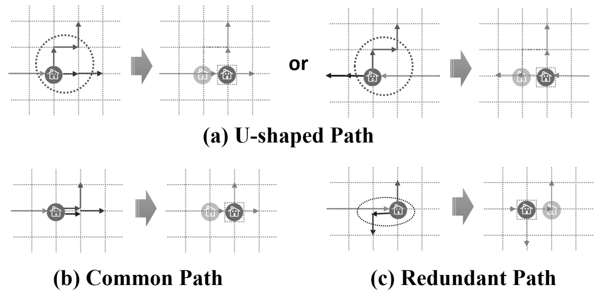

Figure 4: Three rules of our protocol

3 N-nodes RST Protocol

We briefly introduce our distributed algorithm which constructs the RST with 3 nodes. Our proposed protocol can ensure the minimum RST construction using local information only. Unfortunately, an improvement of our protocol which can construct the RST with 4 or more nodes is difficult. Our protocol uses *vHome*, which becomes the steiner point after convergence. A RST with N nodes has at most (N-2) steiner points, this implies the a RST with 3 nodes may have a single steiner point (*vHome*). More *vHome*s are required for constructing the RST with 4 or more nodes. It has 3 or more path (from home to each target) and many U-shaped path can be created and collided. This means the generalized protocol for constructing the RST can deal with these problems. However, selective combining protocol using local information only is impossible, thus, we consider an approximation algorithms which can construct the RST with a reasonable number of edges.

References

[1] S. Takatsu, F. Ooshita, H. Kakugawa, and T. Masuzawa, "Zigzag: Local-information-based self-optimizing routing in virtual grid networks," Proceedings of the 33rd International Conference on Distributed Computing Systems (ICDCS), pp. 358-368, 2013.

[2] F. K. Hwang, "On Steiner Minimal Trees with Rectilinear Distance," SIAM Journal on Applied Mathematics, Vol.30, No.1, pp.104-114, 1976.

Copyright ©2016 Information Processing Society of Japan.
All Rights Reserved.