

논문 2011-48TC-2-2

# 센서네트워크 상의 노드 밀집지역 간 상호연결을 위한 문제

## ( Interconnection Problem among the Dense Areas of Nodes in Sensor Networks )

김 준 모\*

(Joonmo Kim)

### 요 약

본 논문은 ad-hoc 네트워크 또는 센서 네트워크상에서, 주어진 노드들 사이를 상호연결하기 위해 중간노드들을 추가 배치시키는 형태의 상호연결 문제에 대한 연구이다. 이 문제는 NP-hard problem으로 변환된다. 네트워크의 노드들은 응용시스템 또는 지형적인 요인에 의해 일부지역에서는 밀집하여 분포되고, 그 외의 지역에서는 희박하게 분포될 수 있다. 이러한 경우, 노드들이 밀집한 지역의 상호연결을 무시함으로써, 보다 짧은 실행시간 안에 추가노드들의 최적배치에 근접하도록 하는 방법을 만들 수 있다. 그러나 이러한 경우라 하더라도 여전히 NP-hard이므로, 동적프로그래밍을 구현함으로써 다항시간 근사전략(PTAS)을 구성하는 것이 타당하다. 실행결과 등에 대한 분석은 목적함수를 적절하게 정의함으로써 가능해진다. 목적함수는 노드 밀집지역을 추상화시킴에 의해 발생하게 되는 문제점에 대처할 수 있도록 정의되어야 한다.

### Abstract

This paper deals with the interconnection problem in ad-hoc networks or sensor networks, where relay nodes are deployed additionally to form connections between given nodes. This problem can be reduced to a NP-hard problem. The nodes of the networks, by applications or geographic factors, can be deployed densely in some areas while sparsely in others. For such a case one can make an approximation scheme, which gives shorter execution time, for the additional node deployments by ignoring the interconnections inside the dense area of nodes. However, the case is still a NP-hard, so it is proper to establish a polynomial time approximation scheme (PTAS) by implementing a dynamic programming. The analysis can be made possible by an elaboration on making the definition of the objective function. The objective function should be defined to be able to deal with the requirement incurred by the substitution of the dense area with its abstraction.

**Keywords :** Sensor Networks, Graph Interconnection, NP-hard Problem, Approximation Algorithm, Steiner Tree Problem

## I. Introduction

This paper deals with an interconnection problem between nodes in ad-hoc networks or sensor networks. The nodes of networks, by application or geographic factors, can be deployed densely in some

areas while sparsely in others. Over the sparsely deployed network areas, to make the nodes get interconnected with the shortest distance, it is necessary to know the special locations called steiner points<sup>[1]</sup>: locally interconnecting through which the global interconnection over the network will have the shortest total length. On the other hand, over the densely deployed areas, there is no need to be concerned about the interconnections since it is assumed that the nodes in there are given to be

\* 정회원, 단국대학교 컴퓨터학부  
(Member, Computer Science & Engineering,  
Dankook University)  
접수일자: 2010년12월23일, 수정완료일: 2011년1월27일

interconnected. Therefore, by abstracting and simplifying the dense areas, one may construct the approximation scheme effectively that performs the overall interconnections efficiently. As a first thought, one may expect constructing an interconnection scheme by abstracting the dense areas to two dimensional shapes such as circles or rectangles. However, at this point, one has a problem that it is hard to define or make use of the objective function, which is necessary in analyzing the approximation ratio for the execution results. In analyzing and evaluating the interconnection between nodes, one needs to define and apply the objective function that represents the accumulation of inter node distances over the interconnections. Meanwhile, if the area of two dimensional shape is intermixed in the representation or evaluation of interconnections, it becomes hard to define the objective function, or the evaluation for the execution results becomes complicated or meaningless. So, one needs to devise a gadget that helps to define the objective function. By substituting the dense areas with gadgets, explained in what follows, one is able to omit two dimensional area from the problem instance. After the substitution, one may be able to abstract and define the problem as follows in this section, and construct its approximation scheme in the next sections.

A *gadget* is a two dimensional shape that covers the dense area most appropriately, or it may represent a node of a point. For the purpose of the proof, one should confine a gadget to be a point, a line, a triangle or a rectangle. On the other hand, one may know, from what follows in the next sections, that it can be expanded to be an arbitrary convex polygon. Note that the dense area, which is close to the concave polygon form, can be partitioned into multiple smaller convex polygons. Inside of a gadget is where there is no steiner point, and abstracts the dense area of a network. It can be easily seen that the errors, coming from the abstraction of the dense areas into gadgets, should be reduced into an acceptable level by modifying the gadget into a

convex polygon with more number of sides. The other remaining errors can reasonably be ignored. Now one may define the interconnection problem for the graph represented by the gadgets as: the problem is to construct the set of interconnection lines that has the minimum total length to interconnect  $n$  *gadgets*,  $g_1, g_2, \dots, g_n$ . One may see analogous problems in [2~3]. The case that  $g_i$  is just one node, which is a point in the abstraction, for all  $i = 1, 2, \dots, n$  is just the Steiner Minimum Tree problem<sup>[4]</sup>, which is NP-hard. Thus the proposed problem is also NP-hard<sup>[5~6]</sup>. For this problem, this paper provides the scheme for constructing its algorithm and its analysis in terms of the execution time and the optimality of the result.

The notations and definitions in the following sections are borrowed from [2~3, 7~8]. Next section is for the definitions in constructing the approximation scheme. Section III shows that moving the given problem instance from the real coordinate into the integer one is a necessary and also can be an acceptable modification for the approximation scheme. Section IV gives component ideas and the main theorem. Section V is for the skeleton of the associated dynamic programming and its run time analysis. Section VI is for the proof of the main theorem, and Section VII is the conclusion.

## II. Total Length and Other Definitions

Our objective is to find the interconnection that has the almost minimum sum of the interconnecting edge lengths for the problem instance. For this, first of all, one needs to have definitions and a proposition as follows:  $s$  is a side (line segment) of the perimeter of a gadget,  $|s|$  is the length of  $s$ ,  $\ell$  is an optimally interconnecting edges between two gadgets,  $|\ell|$  is the length of  $\ell$ ,  $\tilde{\ell}$  is an interconnecting edge between two gadgets formed by the approximation,  $|\tilde{\ell}|$  is the length of  $\tilde{\ell}$ ,  $T$  is a graph formed by  $s$ 's and  $\ell$ 's,  $|T|$  is  $\sum_{\text{for all } s} |s| + \sum_{\text{for all } \ell} |\ell|$ ,  $\mathcal{L}$  is a graph of

all  $\ell$ 's,  $|\mathcal{L}|$  is  $\sum_{\text{for all } \ell} |\ell|$ . Likewise,  $T_a$  is a graph formed by  $s$ 's and  $\tilde{\ell}$ 's;  $|T_a|$ ,  $\mathcal{L}_a$  and  $|\mathcal{L}_a|$  are defined analogously. As well,  $c(> 0)$  is a constant, and  $\epsilon_1(> 0)$  and  $\epsilon_2(> 0)$  are two arbitrarily small constants.

It may happen that  $|\mathcal{L}|$  comes to be almost zero, and then it becomes meaningless to define the approximation ratio in terms of  $|\mathcal{L}|$ . To avoid this situation, one needs to approximate in terms of  $|T|$ . Then the following proposition is in needed.

Proposition 1 Under the reasonable assumption that

$$\sum_{\text{for all } s} |s| \leq c \sum_{\text{for all } \ell} |\ell|, (1 + \epsilon_1)\text{-approximation for } T$$

implies  $(1 + \epsilon_2)$ -approximation for the  $\mathcal{L}$ .

Proof:

$$\begin{aligned} |T_a| &= |\mathcal{L}_a| + \sum_{\text{for all } s} |s| \\ &\leq (1 + \epsilon_1)|T| \\ &= (1 + \epsilon_1)(|\mathcal{L}| + \sum_{\text{for all } s} |s|), \end{aligned}$$

$$\begin{aligned} |\mathcal{L}_a| &\leq (1 + \epsilon_1)|\mathcal{L}| + \epsilon_1 \sum_{\text{for all } s} |s| \\ &\leq (1 + \epsilon_1(1 + c))|\mathcal{L}| \\ &= (1 + \epsilon_2)|\mathcal{L}| \quad \blacksquare \end{aligned}$$

### III. Shifting into the Integer Coordinates

In order to run the program, the instance of our problem should be shifted into the integer coordinates. The shift will move each end point into the nearest integer position. As well, the steiner

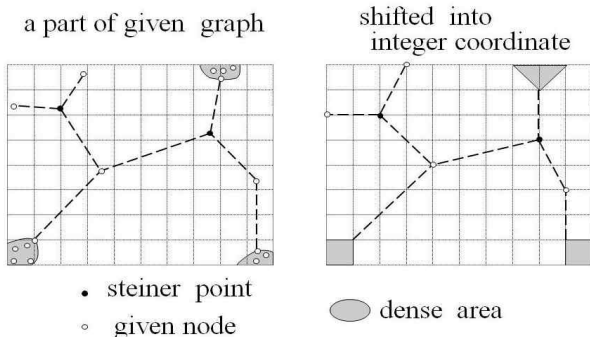


그림 1.  $T$ 에서  $T^\dagger$ 로의 이동  
Fig. 1. Shift from  $T$  to  $T^\dagger$ .

points, which aids the minimum connection between given nodes, should also lie at the integer positions. With more definitions that  $T^\dagger$  is  $T$  on integer coordinate, and  $T_a^\dagger$  is  $T_a$  on integer coordinate, one may show that the shifts are acceptable.

Proposition 2  $(1 + \epsilon)$ -approximation over  $T^\dagger$  implies  $(1 + \hat{\epsilon})$ -approximation over  $T$ .

Proof: By the shift and the assumption, one may have the following inequality:  $|T| - |T^\dagger| \leq 2(6n - 3)$ .

The term  $(6n - 3)$  is the number of edges of the graph. Note that the maximum number of points for the problem instance is  $(5n - 2)$  that is the sum of the maximum number,  $4n$ , of points of gadgets and the number,  $n - 2$ , of steiner points. Considering the tree with  $(5n - 2)$  nodes, there are  $(5n - 3)$  edges. But,  $T$  may have  $n$  more edges than the tree due to the shape of gadgets such as triangles or rectangles. The term 2 is for the maximum length increase by

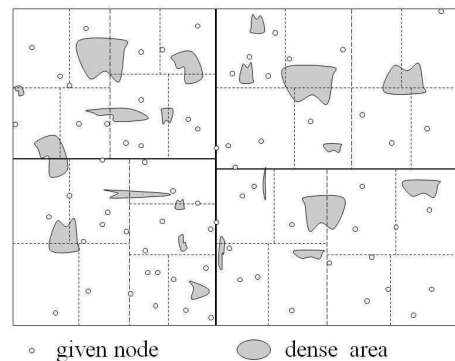


그림 2. 2:3-타일링  
Fig. 2. 2:3-tiling.

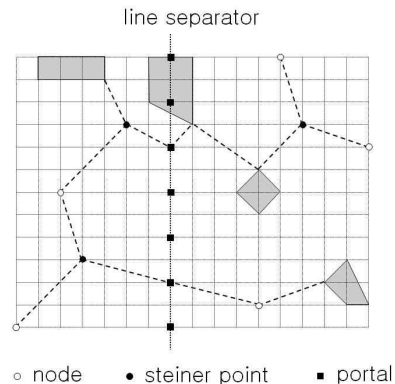


그림 3. m-light 그래프  
Fig. 3. M-light graph.

the shift of an edge: each of both ends of an edge can increase its length unit distance at the most.

Likewise,  $|T_a| - |T_a^\dagger| \leq 2n^2$ , where  $n^2$  is an upper bound for the number of edges, supposing the approximated-graph is a complete network. This condition,  $|T_a^\dagger| \leq (1 + \epsilon) \cdot |T^\dagger|$ , will be shown to be satisfied by the PTAS and then,

$$\begin{aligned} |T_a| - 2n^2 &\leq (1 + \epsilon)(|T| + 2(6n - 3)) \\ |T_a| &\leq (1 + \epsilon)|T| + 2(1 + \epsilon)(6n - 3) + 2n^2 \\ &= (1 + \epsilon + \frac{2(1 + \epsilon)(6n - 3) + 2n^2}{|T|})|T| \\ &= (1 + \epsilon + \frac{2(1 + \epsilon)(6n - 3) + 2n^2}{n^3})|T| \\ &\leq (1 + \hat{\epsilon})|T| \end{aligned}$$

where  $\hat{\epsilon}$  can be chosen appropriately. ■

The term  $n^3$  has the key role in this proof, and it can be acquired by setting the unit length of the integer coordinate short enough:  $T$  gets bigger as the unit of length gets shorter.

#### IV. Partitioning, Portals and the Structure Theorem

The first stage of the scheme is to partition the given problem instance so as to form a dynamic programming. A rectangle, denoted as  $R$ , in partitioning the problem instance is an axis-aligned rectangle. The size of the rectangle is the length of its longer edge. The bounding box of the problem instance is the smallest rectangle enclosing them. A line-separator, denoted as  $l^{sep}$ , of a rectangle is a straight line segment, which is parallel to the rectangle's shorter edge, and which partitions the rectangle into two sub-rectangles. Each of the sub-rectangles should occupy at least  $(1/3)_{rd}$  the area. For example, if the rectangle's width  $W$  is greater than its height, then  $l^{sep}$  is any vertical line in the middle  $W/3_{rd}$  of the rectangle. Now a recursive partition, on which the dynamic programming may run, of a rectangle is defined as follows.

**Definition 1 (2:3-tiling)** A 2:3-tiling of a rectangle  $R$  is a binary tree (hierarchy) of sub-rectangles of  $R$ .  $R$  is at the root. If the size of any sub-rectangle is  $\leq 1$ , then it's the terminal of the tree. Otherwise, the root contains  $l^{sep}$  for  $R$ , and has two sub-trees that are 2:3-tiling of the two rectangles, into which  $l^{sep}$  divides  $R$  recursively.

Note that the rectangles at depth  $d$  in the tiling form a partition of the root rectangle. The set of all rectangles at depth  $d+1$  is a refinement of this partition obtained by putting  $l^{sep}$  through each depth  $d$  rectangle of size  $> 1$ . The area of any depth  $d$  rectangle is at most  $\left(\frac{2}{3}\right)^d$  times the total area. The following proposition is therefore immediate.

**Proposition 3** If a rectangle has width  $W$  and height  $H$ , then its 2:3-tiling has depth at most  $\log_{1.5} W + \log_{1.5} H + 2$ .

**Definition 2 (portals)** A portal in a 2:3-tiling is any point that lies on the edges of rectangles in the tiling. If  $m$  is any positive integer then a set of portals  $P$  is called  $m$ -regular for the tiling if there are exactly  $m$  equidistant portals on  $l^{sep}$  of each rectangle of the tiling. The two crossing points of  $l^{sep}$  and  $R$  are also portals. In other words,  $l^{sep}$  can be divided into  $m-1$  equal parts by the portals on it.

**Definition 3 (m-light)** Let  $S$  be a 2:3-tiling of the bounding box,  $P$  be an  $m$ -regular set of portals on  $S$ , and  $m \in \mathbb{Z}^+$ . Then the associated  $T_a^\dagger$  is  $m$ -light with respect to  $S$  if the followings are true: (i) in each rectangle of tiling  $S$ , all  $\tilde{\ell}$ 's cross  $l^{sep}$  of that rectangle at most  $m$  times (ii) each  $\tilde{\ell}$  crosses  $l^{sep}$  only at portals in  $P$ .

**Theorem (Structure Theorem)** The following is true for each  $\epsilon > 0$ . Every set of gadgets in the problem has a  $(1 + \epsilon)$ -approximate  $T_a^\dagger$  and an associated 2:3-tiling  $S$  of the bounding box such that the  $T_a^\dagger$  is  $m$ -light for  $S$ , where  $m = O\left(\frac{\log n}{\epsilon}\right)$ .

### V. The Polynomial Time Dynamic Programming (DP)

By Proposition 2 and the proof of the Structure Theorem in the next section,  $\mu$ -light can be defined as a  $T_a^\dagger$  that has  $\mu$ -light property and the approximation ratio  $(1 + \epsilon)$ . One can build  $\mu$ -light up to the root of the tiling with the DP. The Structure Theorem guarantees the existence of  $\mu$ -light and its tiling  $S$ , where the number of the portals is  $m = O(\frac{\log n}{\epsilon})$ . By Proposition 3, the depth of  $S$  is at most  $O(\log n)$ . Now, the description for the DP that finds both  $S$  and  $\tilde{\ell}$ 's for a  $\mu$ -light comes next in this section. The execution time will be shown to be a polynomial time of  $poly(n) \cdot 2^{O(m)} = n^{O(1/\epsilon)}$ .

The work of the DP is bottom-up approach, but it is easy to see the procedure from the final stage to the start. The final result of the DP is the rectangle, which is the bounding box for the given problem instance, and which contains the  $\mu$ -light that can be the almost optimal solution for the problem.

Right before the final rectangle is reached, many combinatorial cases must be checked. All  $l^{sep}$ 's that could divide the final rectangle into two sub-rectangles according to the 2:3-tiling are considered one by one as in Fig. 4. Along such a  $l^{sep}$ , there are combinations of choices of portals that can be represented by multi-sets. Then, for each choice of

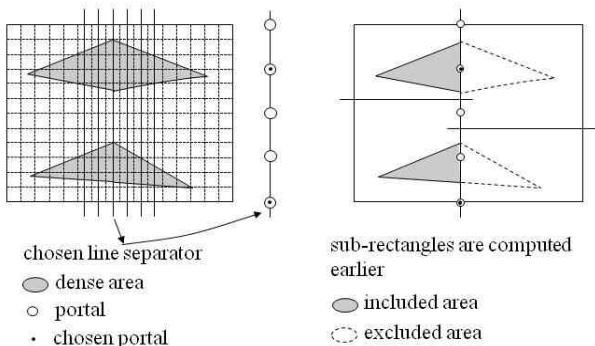


그림 4. line-separator를 사용한 분할  
Fig. 4. Partitioning with a line-separator.

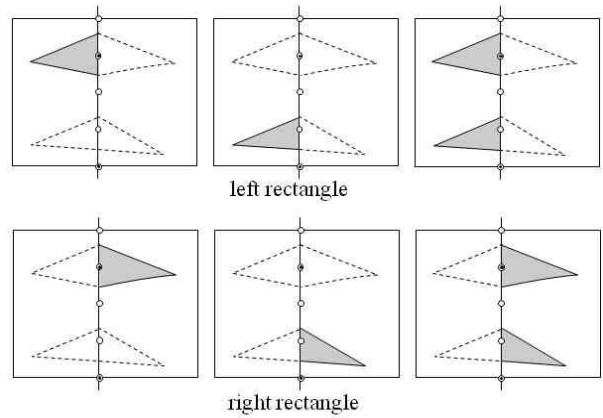


그림 5. 포함/제외에 의한 조합  
Fig. 5. Combinations from inclusion/exclusion.

$l^{sep}$  and its associated multi-set of portals, there is one more level of choices for  $s$ 's. The choices of  $s$ 's are determined by their inclusions/exclusions within each of the sub-rectangles as in Fig. 5. For each of the many combinatorial cases, it has its own minimum cost  $\mu$ -light. Such a minimum cost  $\mu$ -light comes from the concatenation of the two smaller  $\mu$ -light from each of the sub-rectangles.

So a rectangle holds as many minimum cost  $\mu$ -light's as the number of the combinations.

The same observations hold repeatedly in each of the sub-rectangle for finding its own minimum cost  $\mu$ -light until the work reaches the bottom most rectangles, where there are limited number of  $s$ 's, and so the brute-force algorithm can find the minimum cost  $\mu$ -light for each of the combinatorial cases inside the smallest rectangle. Out of all the minimum cost  $\mu$ -light's from the cases of combinations in the root rectangle, the minimum one from them all will be chosen to be the approximated solution for the problem, the final solution.

Now it is to be shown that the number of entries of the lookup table for this DP is a polynomial, and the run time for each of the entries is in a poly time. An entry can be indexed by the triple: (a) a rectangle, (b) a multi-set of  $i (\leq 4m)$  portals along the perimeter of the rectangle, and (c) a set of the  $s$ 's inside the rectangle.

For (a), the number of distinct rectangles is at

most  $\binom{2(3n-3)}{4}$  since the maximum number of points are  $2(3n-3)$  from the Proof of Proposition 2. For (b), each rectangle has 4 sides, which are parts of the  $l^{sep}$ 's of its ancestors. The  $m$  portals on  $l^{sep}$  are evenly spaced, so they are determined once a  $l^{sep}$  is chosen. But the number of choices for a  $l^{sep}$  is at most the number of pairs of points, which is  $\binom{n}{2}$ . This accounts for the factor:  $O((n^2)^4) = O(n^8)$ . Once the set of  $4m$  portals on the four sides is identified, the number of ways of choosing a multi-set of size  $i$  out of them is at most  $2^{4m+i}$ . For (c), if  $j (\leq 4m)$   $s$ 's exist in a rectangle, there may exist  $2^{4m}$  possible sets at the most. However, if more than  $m$   $s$ 's pass the  $l^{sep}$  of the rectangle, the number of possible sets of the  $s$ 's inside the rectangle will be artificially set to 1 by using the bridge described in the next section. Hence the upper bound of the size of the lookup table is:

$$\sum_{i=1}^{4m} n^4 \times n^8 \times 2^{4m+i} \times 2^{4m}, \text{ which is}$$

$$O(n^{12} 2^{12m}) = n^{O(1/\epsilon)}.$$

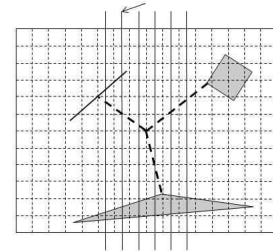
The running time over the lookup table is now to be considered. The bottom level rectangles has limited number of  $s$ 's of  $O(m)$  and so run the brute-force algorithm for each in  $2^{O(m)}$  time. For each of the upper level rectangles, the minimum value is computed by the comparison operations over the sub-rectangles, so the time can be bounded by a polynomial. Therefore, the running time of the DP is upper bounded by  $2^{O(m)} \times poly(n) \times table\ size$ , which is  $n^{O(1/\epsilon)}$ . Note that the DP is organized so that all the m-light graphs can be reached. Then, if the existence of  $\mathfrak{M}$ -light in any chosen rectangle is ensured, the m-light graph with the minimum length will turn out to be a  $\mathfrak{M}$ -light. The proof of the Structure Theorem in the next section shows the existence of  $\mathfrak{M}$ -light in a rectangle.

## VI. Proof of the Structure Theorem

The Structure Theorem shows the existence of the  $\mathfrak{M}$ -light whose difference from  $T^\dagger$ , which is the mathematical optimality, is arbitrarily small. Once the existence of  $\mathfrak{M}$ -light is shown, the minimum cost m-light graph, which will be found by the DP, is naturally a  $\mathfrak{M}$ -light. The proof of the Structure Theorem can be stated as follows. For each rectangle over the 2:3-tiling of the problem instance, one may choose a  $l^{sep}$  that (virtually) crosses with the edges of  $T^\dagger$  less times than the other  $l^{sep}$ 's. Let the points, at which the edges of  $T^\dagger$  cross with the chosen  $l^{sep}$ , be target-points. Then the  $\mathfrak{M}$ -light whose edges and points cross at the nearest portals to the target-points will be shown to be within the expected approximation bound.

The analytic sum of the lengths between the target-points and their nearest portals, through which the edges of  $\mathfrak{M}$ -light is passing, may represent the estimation of the length difference between the optimal structure  $T^\dagger$  and the m-light structure  $\mathfrak{M}$ -light. The  $|T^\dagger|$  inside the rectangle can be represented by the minimum possible length estimation, which can be expressed by Lemma 1 that is illustrated in Fig. 6. Let a unit-band in a rectangle be the sub-rectangle with the unit length of width and the height of length  $L$ .

Lemma 1 When a  $l^{sep}$  inside the unit-band crosses  $k(\in N \cup \{0\})$  times with  $\tilde{\ell}$ 's or  $s$ 's, the minimum



— line separator    - - - interconnect line     dense area

그림 6. line-separators를 포함한 6개의 단위 밴드  
Fig. 6. Six unit bands with line-separators.

possible  $|T^\dagger|$  in the unit-band is  $\frac{k}{2}$ .

Proof: The accumulated length of the line segments,  $\tilde{\ell}$ 's or  $s$ 's, inside the unit-band gets to be the shortest when all the line segments touch perpendicular to  $l^{sep}$ . By the  $k$  crossings, the minimum cost in the unit-band is  $\frac{k}{2}$ . ■

For each rectangle over the 2:3-titling, in its middle  $\frac{1}{3}$  area, one  $l^{sep}$  can be chosen such that the  $l^{sep}$  crosses  $k$  times with  $T^\dagger$  while other  $l^{sep}$ 's in the same area cross at least  $k$  times. As a result, the minimum  $|T^\dagger|$  in the middle  $(1/3)_{rd}$  area could be estimated as  $\frac{k}{2} \cdot \frac{1}{3}L$ . Along the chosen  $l^{sep}$ , a  $\mathcal{M}$ -light can be built up. The minimum length  $\mathcal{M}$ -light graphs that has less or equal length than that of  $\mathcal{M}$ -light will be found by the DP.

Proof: (Structure Theorem) One needs to figure out how much more length  $\mathcal{M}$ -light has than  $T^\dagger$ . By the number  $m$  of crossings between  $T^\dagger$  and a  $l^{sep}$ , there are two cases to be considered as in Fig. 7.

Case I For a rectangle, there is a  $l^{sep}$ , which is crossed  $k$  ( $\leq m$ ) times by  $T^\dagger$ , where  $k$  is the minimum number of the crossings that a  $l^{sep}$  in the rectangle may have.

Case II For a rectangle, all the  $l^{sep}$ 's cross with  $T^\dagger$  more than  $m$  times.

In Case I, the maximum length difference between

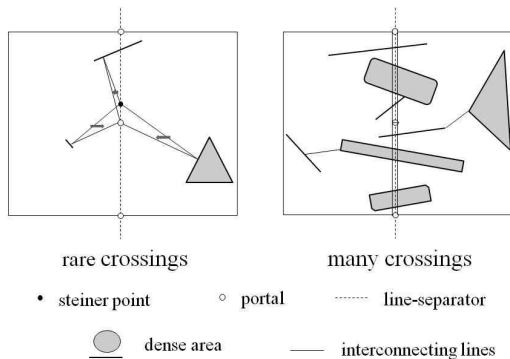


그림 7. 교차  
Fig. 7. Crossings.

$T^\dagger$  and  $\mathcal{M}$ -light in a rectangle can be  $3k(\frac{L}{m})$  when all the crossing points of  $\mathcal{M}$ -light at the portals are steiner points of degree 3, and each of the associated edges increase the length up to  $\frac{L}{m}$  to reach the nearest target-points. This is due to the properties of steiner tree interconnections<sup>[1,4]</sup>. Thus, the ratio of the length difference to  $T^\dagger$  in the rectangle turns out to

$$\text{be: } \frac{\frac{3k}{m}L}{\frac{k}{2} \cdot \frac{1}{3}L} = \frac{18}{m}. \text{ Now one may let } m = O\left(\frac{\log n}{\epsilon}\right)$$

so that the ratio of the length of  $\mathcal{M}$ -light to  $|T^\dagger|$  is  $\leq (1 + \frac{18}{m}) = (1 + \hat{\epsilon})$ .

In Case II, the estimated minimum  $|T^\dagger|$  in the  $(1/3)_{rd}$  area of a rectangle is  $\frac{m}{2} \cdot \frac{1}{3}L$  since  $k = m$ .

Let  $br$  be a bridge inside a sub-rectangle. A bridge is a line segment that lies in parallel with and tightly close to (, but not touching to) the  $l^{sep}$  in the rectangle. Note that a rectangle should contain two  $br$ 's because each of the two sub-rectangles may have one  $br$  respectively. The  $\tilde{\ell}$ 's,  $s$ 's and steiner points of a  $\mathcal{M}$ -light in one sub-rectangle will shown to be connected to those in the other sub-rectangle with the intentional use of  $br$ 's for the analysis purpose. In fact, a  $br$  is one of  $\tilde{\ell}$ 's. Because the density of the  $\tilde{\ell}$ 's in this case is determined to be high, two  $br$ 's of length  $L$  are added to form a  $\mathcal{M}$ -light, instead of considering the details, which would involve exponential time complexity, of the crossings. Actually, all  $\tilde{\ell}$ 's and  $s$ 's are assumed to be touched with a  $br$ , instead of passing through portals. Later the two  $br$ 's are connected with each other through the portal by an infinitesimally short line segment whose lengths can be ignored. So, the ratio of the length difference to the  $T^\dagger$  in the rectangle is:  $\frac{2L}{\frac{m}{2} \cdot \frac{1}{3}L} = \frac{12}{m}$ . As a result, the length

difference from the two cases is at most  $\frac{18}{m}$ . Since the tiling has depth of  $O(\log n)$ , the approximation ratio is

$$\begin{aligned} & \left(1 + \frac{18}{m}\right)^{O(\log n)} \\ &= \left(1 + \frac{18}{m}\right)^{m \left(\frac{1}{m} O(\log n)\right)} \\ &= e^{O(\epsilon)} \leq (1 + \epsilon) \quad \blacksquare \end{aligned}$$

## VII. Conclusion

The purpose of the scheme is to find the location, called steiner point, so as to form the interconnection that has the shortest total length. One may design a polynomial time algorithm from the scheme. As the algorithm can handle more portals, execution time will increase within a polynomial bound, giving the output with a shorter total length. After having the result of the interconnection algorithm; if there are any two given nodes, which should be connected directly, and which are located far each other beyond the transmission radius of the sensor node, one needs to have another consideration about putting multiple relay nodes between them. This may lead to additional scheme and algorithm with its own analysis, and thus make an independent research later. In implementing the algorithms, the schemes may give the guide for the possible pair of the expected precision of the results and the run time for it, as well as the procedures of the algorithms.

## References

- [1] E.N. Gilbert and H.O. Pollak, "Steiner minimal trees," SIAM Journal on Applied Mathematics, Vol. 16, pp.1-29, 1968.
- [2] X. Cheng, J.-M. Kim and B. Lu, "A Polynomial Time Approximation Scheme for the Problem of Interconnecting Highways," Journal of Combinatorial Optimization, Vol 5, issue 3, pp. 327-343, 2001.
- [3] DING-ZHU DU, FRANK K. HWANG and

GUOLIANG XUE. "Interconnecting Highways," SIAM Discrete Mathematics, Vol. 12, pp. 252-261, 1999.

- [4] Z.A. Melzak, "On the problem of Steiner," Canadian Mathematics Bulletin, Vol. 4, pp. 143-148, 1961.
- [5] Ding-Zhu Du, Ker-I Ko, Theory of Computational Complexity, Wiley Inter-Science 2000.
- [6] Michael R. Garey, David S. Johnson, Computers and Intractability A Guide to the Theory of NP-Completeness, Freeman, 1978.
- [7] S. Arora, "Polynomial-time approximation schemes for Euclidean TSP and other geometric problems," Proc. 37th IEEE Symp. on Foundations of Computer Science, pp. 2-12, 1996.
- [8] S. Arora, "Nearly linear time approximation schemes for Euclidean TSP and other geometric problems," Proc. 38th IEEE Symp. on Foundations of Computer Science, pp. 554-563, 1997.

## 저 자 소 개



김 준 모(정회원)

1989년 서울대학교 컴퓨터공학과 학사

2001년 University of Minnesota 전산학 박사

2002년~2004년 한국정보보호진흥원 연구원

2004년~현재 단국대학교 컴퓨터학부 부교수

<주관심분야 : Approximations for NP-hard problems>