TR UCSB 2007-03*

Approximation Algorithms for the Minimum-Length Corridor and Related Problems

Arturo Gonzalez-Gutierrez† and Teofilo F. Gonzalez

Department of Computer Science

University of California

Santa Barbara, CA, 93106

E-mail: {aglez,teo}@cs.ucsb.edu

May 14, 2007

Abstract

Given a rectangular boundary partitioned into rectangles, the Minimum-Length Corridor (MLC-R) problem consists of finding a corridor of least total length. A corridor is a set of connected line segments, each of which must lie along the line segments that form the rectangular boundary and/or the boundary of the rectangles, and must include at least one point from the boundary of every rectangle and from the rectangular boundary. The MLC-R problem has been shown to be NP-hard. In this paper we present the first polynomial time constant ratio approximation algorithm for the MLC-R and MLC$_n$ problems. The MLC$_n$ problem is a generalization of the the MLC-R problem where the rectangles are rectilinear k-gons, for $k \leq n$. We also present a polynomial time constant ratio approximation algorithm for the Group Traveling Salesperson Problem (GTSP) for a rectangle partitioned into rectilinear k-gons as in the MLC$_n$ problem.

*This Tech. Rep. replaces TR 2006-12.

†On leave under the PROMEP program from the Universidad Autonoma de Queretaro. Supported in part by a Grant from UC MEXUS-CONACyT.

1 Introduction

An instance I of the Minimum Length Corridor (MLC-R) problem consists of a pair (F, R), where F is a rectangle partitioned into a set R of rectangles[1] (or rooms) $R_1, R_2, \ldots, R_r$. A *corridor* $T(I)$ for instance I is a set of connected line segments, each of which lies along the line segments that form F and/or the boundary of the rooms, that must include at least one point from every room and from rectangle F. The objective of the MLC-R problem is to construct a minimum edge-length corridor. It is simple to see that the line segments in an optimal corridor do not form any loops, i.e., no two points have more than one path between them along an optimal corridor. A generalization of the MLC-R problem where the rooms are rectilinear polygons is called the MLC problem. The MLC problem becomes the MLC$_n$ problem when every room is a rectilinear k-gon, for $k \leq n$. The MLC problem was initially defined by Naoki Katoh [8] and subsequently Eppstein [11] introduced the MLC-R problem. Experimental evaluations of several heuristics for the MLC problem are discussed in [15]. The question as to whether or not the decision version of these problems are NP-complete is raised in the above three references. Mitchell [17] raised the question as to whether or not the Group Steiner Tree problem (a problem related to our MLC problem) for a set of points in 2D space has a polynomial time constant ratio approximation algorithm.

Recently Gonzalez-Gutierrez and Gonzalez [12], and independently Bodlaender, et al. [5], proved that the MLC problem is strongly NP-complete. The reductions in these two papers are different. Gonzalez-Gutierrez and Gonzalez [12] also showed that the decision version of the MLC-R problem is strongly NP-complete as well as some of its variants. In virtue of these results, attention has shifted to the corresponding approximation problems.

Bodlaender, et al. [5] consider several restricted versions of the MLC problem. One restricted version of the MLC problem is called the *geographic clustering* problem. In this case, there is a square with side length q which can be enclosed by each room, and the perimeter of each room is bounded above by cq, where $c \geq 4$ is a constant. Clearly, not all instances of the MLC-R problem are geographic clustering instances. It is not known whether the decision version of the geometric clustering problem is NP-complete. A PTAS for the geographic clustering version of the MLC and related problems is presented in [5].

[1] Throughout this paper we assume that all the rectangles (and rectilinear polygons) consist only of horizontal and vertical line edges.

Another restricted version of the MLC problem is when each room has a size ρ_i defined as the smallest enclosing square of the room and each room R_i has perimeter at most $4\rho_i$. A room with this property is said to be a room with *square perimeter*. A room R_i is called α-fat if for every square Q whose boundary intersects R_i and its center is inside R_i, the intersection area of Q and R_i is at least $\frac{\alpha}{4}$ times the area Q. In general $\alpha \in [0, 1]$. For square rooms α is equal to one, but for rectangular rooms α tends to zero. It is not known whether the MLC problem, when all the rooms have square perimeter and are α-fat, is an NP-complete problem; however, there is a polynomial time approximation algorithm with approximation ratio $\frac{16}{\alpha} - 1$ [5]. In the case when all the rooms are squares the approximation ratio is 15. Clearly, $\frac{16}{\alpha} - 1$ is not bounded by a constant for the MLC-R.

In this paper we present a polynomial time approximation algorithm for the MLC-R problem with approximation ratio 30. This is the first constant ratio approximation algorithm for the MLC-R problem. The approximation ratio is twice the one for the case when all the rectangles are squares [5]. We also present a polynomial time constant ratio approximation algorithm for the MLC_n problem, when n is a constant.

In Section 2 we discuss preliminary results and related problems. In Section 3 we present our parameterized approximation algorithm for the p-MLC-R problem, a restricted version of the MLC-R problem used to approximate the MLC-R problem. In Section 4 we discuss some properties for which the parameterized algorithm is not a constant ratio approximation algorithm for the p-MLC-R problem. In Section 5 we present our polynomial time constant ratio approximation algorithm for the p-MLC-R problem. In Section 6 we discuss results for a related problem, conclusions, and open problems. In particular we discuss a polynomial time constant ratio approximation algorithm for the MLC_n problem when n is a constant. We also discuss our approximation algorithm for the Group Traveling Salesperson Problem (GTSP) for a rectangle partitioned into rectilinear k-gons as in the MLC_n problem. An application for the MLC problem is when laying optical fiber in metropolitan areas and every block (or set of blocks) is connected through its own gateway. The objective is to find a minimum-length corridor interconnecting all the blocks in the area. Our problems have also applications in VLSI and floorplanning when laying wires for clock signals or power, and when laying wires for an electrical network or optical fibers for data communications. There has been recent research activity for related problems arising in public and private transportation as well as in spatial databases for trip planning queries. Section 6 discusses our problems under other objective functions.

2 Preliminaries and Related Problems

Consider the restricted version of the MLC problem when all the possible corridors must include the access point p located on the edges of the rectangle F. We called this problem the *p-access* point version of the problem or simply the p-MLC and p-MLC-R problems. The decision versions of these problems are shown to be NP-complete in [12]. The thick line segments represent an optimal corridor for the p-MLC-R problem instance given in Figure 1. The solution to any instance of the MLC (resp. MLC-R) problem can be obtained by finding a solution to the p-MLC (resp. p-MLC-R) problem at each intersection point p located along the edges of rectangle F, and then selecting the best solution. Based on this observation we state the following theorem. The same type of theorem can be established for the MLC and related problems.

Theorem 2.1 *Any polynomial time constant ratio approximation algorithm for the p-MLC-R problem is also a polynomial time constant ratio approximation algorithm for the MLC-R problem. The approximation ratio for both algorithms is the same.*

Proof: By the above discussion.

□

As a result of this theorem, we have reduced our problem to finding an approximation to the p-MLC-R problem. Gonzalez-Gutierrez and Gonzalez [12] discuss other versions of the MLC and MLC-R problems. One

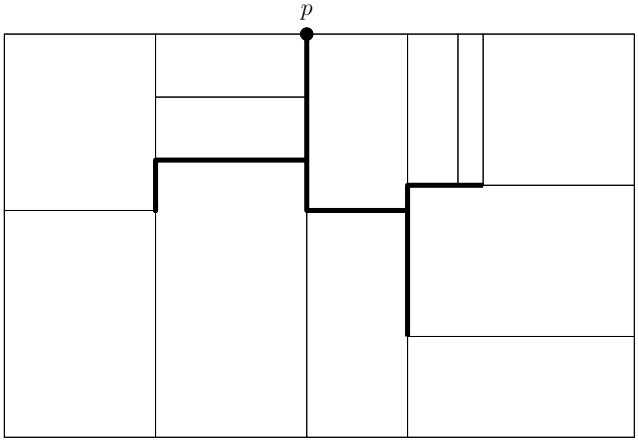


Figure 1: Optimal Corridor for a p-MLC-R problem instance

of these generalizations is when rectangle F is a rectilinear polygon.

We may further generalize the MLC problem to graphs. In this case we are given a connected undirected edge-weighted graph and the objective function is to find a tree with least total edge-weight such that every cycle in the graph has at least one of its vertices in the tree. We call this problem the *network MLC problem* (N-MLC). In graph theoretic terms the set of vertices that break all the cycles in a graph is called the *feedback node set* (FNS). One may redefine the N-MLC problem by requiring that the vertices in the corridor form a feedback node set for the graph. Thus, this problem can also be referred to as the *tree feedback node set* (TFNS) problem, and is formally defined as follows:

INPUT: A connected undirected edge-weighted graph $G = (V, E, w)$ where $w : E \rightarrow \mathbb{R}^+$ is an edge-weight function.

OUTPUT: A tree $T = (V^{'}, E^{'})$, where $E^{'} \subseteq E$, and $V^{'} \subseteq V$ is a feedback node set (i.e., every cycle in G includes at least one vertex in $V^{'}$) and the total edge-weight $\sum_{e \in E'} w(e)$ is minimized.

The TFNS problem restricted to planar and k-outerplanar graphs has been studied under the name of *face cover tree* problem. Bodlaender, et al. [5] present a subexponential time algorithm for some versions of the TFNS problem. A similar problem, the *tree vertex cover* (TVC) problem is discussed in [1]. The difference between the TFNS and TVC problems is that instead of the set $V^{'}$ being a feedback node set for G, it must be a vertex cover for G. A polynomial time approximation algorithm with performance ratio of $\frac{34}{9} < 3.78$ for the TVC problem is given in [1]. This algorithm has two main steps: Find a set of edges (with near-optimal total weight) whose vertices form a vertex cover, and then select a set of edges (with near optimal weight) such that when added to the previous set results in a tree. The first step of this algorithm uses the approximation algorithm for the weighted vertex cover problem and the second one uses an approximation algorithm for the Steiner tree problem. A similar strategy for our MLC problems replaces the approximation algorithm for the weighted vertex cover by a constant ratio approximation algorithm for the weighted FNS problem [2, 4, 6]. However, we can show that even for the p-MLC-R problem this approach and several variations do not generate provably good solutions (see Figure 5(b)).

The *Group Steiner Tree* (GST) problem may be viewed as a generalization of the MLC problem. Reich and Widmayer [18] introduced the GST problem, motivated by applications in VLSI design. The GST problem is defined by Reich and Widmayer as follows.

INPUT: A connected undirected edge-weighted graph $G = (V, E, w)$, where $w : E \rightarrow \mathbb{R}^+$ is an edge-weight function; a non-empty subset S, $S \subseteq V$, of *terminals*; and a partition $\{S_1, S_2, \ldots, S_k\}$ of S.

OUTPUT: A tree $T(S) = (V', E')$, where $E' \subseteq E$ and $V' \subseteq V$, such that at least one terminal from each set S_i is in the tree $T(S)$ and the total edge-length $\sum_{e \in E'} w(e)$ is minimized.

For an instance of the GST problem in which S is partitioned into k subsets, Ihler [14] gives a heuristic which has a performance bound of $(k-1) \cdot OPT$, where OPT is the value of an optimal solution. Bateman et al. [3] give the first known heuristic with a *sub-linear* performance bound of $(1 + \ln(\frac{k}{2})) \cdot \sqrt{k} \cdot OPT$. Helvig, Robins, and Zelikovsky [13] give a polynomial-time $O(k^\epsilon)$-approximation algorithm for any fixed $\epsilon > 0$. The graph Steiner tree (ST) problem is a special case of the GST problem where each set S_i is a single vertex. Karp [16] proved that the decision version of the ST problem is NP-complete. Since the GST problem includes the ST problem, the decision version of the GST problem is also NP-complete. One may reduce the MLC problem to the GST problem [12]. However, none of these approximation algorithms for the GST are constant ratio approximation algorithms for the MLC problem because k, which corresponds to the total number r of rectilinear polygons inside rectangle F, is not bounded by a constant.

Slavik [19, 20, 21] defined a more general version of GST where $\{S_1, S_2, \ldots, S_k\}$ is not necessarily a partition of S, but each $S_i \subseteq S$ and $\cup_i S_i = S$, i.e., a vertex in S may be in more than one set S_i. This version of the GST problem is called by Slavik [20, 21] the *Tree (Errand) Cover* (TEC) problem. Slavik [20, 21] developed an approximation algorithm for the TEC problem with approximation ratio 2ρ, when each errand can be performed in at most ρ locations. Safra and Schwartz [19] established inapproximability results for the 2D version of the TEC problem when each set is connected, but again, these results do not seem to carry over to the MLC problem. The TEC and GST (as defined above) are computationally equivalent problems. In this paper we refer to the GST problem when the groups intersect as the TEC problem.

It is convenient to transform the geometric representation of the p-MLC-R problem into a graph representation. A vertical (resp. horizontal) line segment in the instance I of the p-MLC-R problem is called *maximal* if it intersects horizontal (respectively, vertical) line segments at its two endpoints, but not in between. We assume without loss of generality that p is an intersection point between two or three maximal line segments, and is located on the rectangular boundary F. Every instance I of the p-MLC-R problem is represented by the graph $G(I) = (V, E, w)$, where the set V of vertices represents all the intersection points of maximal

line segments (the intersection of two or more maximal line segments at the same point is represented by a single vertex), the set E of edges represents all the maximal line segments, and the weight of an edge ($w(e)$) corresponds to the length of the maximal line segment represented by the edge. We use $V(R_i)$ to denote all the vertices of rectangle R_i. In this paper we use the geometric and graph representation of the p-MLC-R problem interchangeably, and mix the two notations. Note that $|V(R_i)| \geq 4$ (see Figure 2 where $V(R_5)$ has 9 vertices). We use $C(R_i)$ to denote the set of vertices that corresponds to the corners of R_i. Every vertex is a non-corner point of at most one rectangle (there are five ($|V(R_5)| - 4$) vertices in $V(R_5)$ that are non-corner points of rectangle R_5 (Figure 2)).

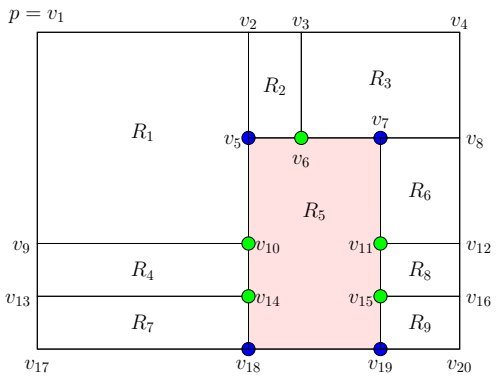


Figure 2: Instance of the p-MLC-R problem.

The instance of the TEC problem corresponding to the instance (F, R, p) of the p-MLC-R problem is defined for the metric graph $G(V, E, w)$ constructed from (F, R, p) with an errand E_i for each rectangle R_i located at all the vertices $V(R_i)$, plus the errand E_0 located at vertex p. Clearly every feasible solution to the p-MLC-R problem instance is also a feasible solution for the TEC problem instance and vice versa. Furthermore, the objective function value of every feasible solution to both problem instances is identical.

Let I be any instance of the p-MLC-R problem. Let $T(I)$ be any corridor for instance I and $t(I)$ be its edge-length. Let $OPT(I)$ be an optimal corridor for instance I and let $opt(I)$ be its edge-length. An approach to generate suboptimal solution for the p-MLC-R problem is to construct an instance of the TEC problem and then invoke an approximation algorithm for the TEC problem. The solution generated by the algorithm for the TEC problem is the solution to the p-MLC-R problem. Currently one uses Slavik's [20, 21] approximation algorithm for the TEC problem. A direct application of this approach to the p-MLC-R problem generates a corridor whose total edge-length is at most $2\rho \cdot OPT(I)$, where $\rho = max_i\{|V(R_i)|\}$. Unfortunately, this simple

approach does not produce a constant factor approximation for the p-MLC-R problem since, as we pointed out before, $|V(R_i)|$ is not bounded above by any constant. In the next section we discuss our parameterized approximation algorithm which is based on the above approach.

Another related problem studied also by Slavik [20, 21] is the *Errand Scheduling* (ES) problem. In this case the problem is to find a *shortest partial tour* visiting a subset of vertices of the given metric graph G such that each *errand* is contained in at least one set $S_i \subset S$ associated to vertex i that is in the partial tour. When each vertex represents a unique errand, the ES problem is an instance of the well-known *Traveling Salesperson Problem (TSP)*. Therefore the ES problem is NP-hard. The ES problem has also been referred to as the *group TSP (g-TSP)*. It is interesting to point out that a solution to the TEC problem can be transformed to a solution to the ES problem, by simply walking around the tree following a pre-order traversal, starting and ending at the root of the tree. This transformation guarantees a solution to the ES problem with weight 2 times the total weight of the solution to the TEC problem. Slavik [20, 21] presents an algorithm that approximates ES within $O(\log m)$ where m is the total number of errands, for the restricted case where the graph G is a weighted tree. Slavik also shows that for metric general weighted graphs the ES problem can be approximated to within $\frac{3\rho}{2}$ when each errand can be performed in at most ρ nodes. In Section 6 we discuss how to apply our results to the g-TSP restricted to rectilinear k-gons as in the MLC_n.

For the Euclidean Group TSP we are given a set of points P in the plane grouped in pairwise disjoint regions, each region is connected and contains at least one point in P. The problem is to find a tour visiting at least one point in P from each region. The tour may visit any point in the plane. Several approximation algorithms have been developed for this problem when the regions are α-fat, which under some conditions are equivalent to the α-fat regions defined before. For this problem, α-fat is defined in terms of disks rather than squares. The approximation algorithm by de Berg et al. [7] has an approximation ratio $12000\alpha^3$, where $\alpha \geq 1$. This ratio is 93 when the regions are squares. As it turns out, the algorithm in [7] has the same general structure as the one of our approximation algorithms. Elbassioni et al. [10] developed a different algorithm with a $9.1\alpha + 1$ approximation ratio. Bodlaender et al. [5] algorithm for the MLC problem with α-fat rooms discussed before is a modified version of Elbassioni et al. [10] algorithm. Dumitrescu and Mitchell [9] present a polynomial time constant ratio approximation algorithm for the TSP with neighborhoods on a set of n connected regions of the same diameter.

As we have seen, our problems are restricted versions of more general ones reported in the literature. But previous results for those problems do not establish NP-completeness results, inapproximability results, nor constant ratio approximation algorithms for our problem.

3 Parameterized Approximation Algorithm

To establish a constant ratio approximation for the p-MLC-R problem, we need to refine our previous strategy that uses an approximate solution to the TEC problem. The idea is to restrict the solution space by limiting in each rectangle R_i the possible vertices, from which at least one must be part of the corridor. Consider the p-MLC-R$_S$ problem where S is a selector function that defines uniformly in each rectangle R_i a set of points from which at least one must be included by every corridor. The points selected for each rectangle R_i are called the *critical points of* R_i. The objective of the p-MLC-R$_S$ problem is to find a minimum edge-length corridor that includes for each rectangle R_i at least one of its critical points. Let $k_S \geq 1$ be the maximum number of critical points selected by S from each rectangle R_i.

Given S and an instance I of the p-MCL-R problem, we use I_S to denote the instance of the corresponding p-MLC-R$_S$ problem. The instance of the TEC problem, denoted by J_S, is constructed from an instance I_S of the p-MLC-R$_S$ problem using the same approach as the one used for the p-MLC-R problem, but limiting the errands from each rectangle to the critical points of the rectangle. Clearly every feasible solution to the p-MLC-R$_S$ problem instance I_S is also a feasible solution to the instance J_S of the TEC problem, and vice versa. Furthermore, the objective function value of every feasible solution to both problems is identical. Slavik's algorithm applied to the instance J_S of the TEC problem constructed from I_S generates a solution to the TEC problem from which we construct a corridor $T(I_S)$ with edge-length $t(I_S)$ of the p-MLC-R problem. We call our approach the parameterized algorithm $Alg(S)$, where S is the parameter. Let $OPT(I_S)$ be an optimal corridor for I_S and let $opt(I_S)$ be its edge-length. Theorem 3.1 establishes the approximation ratio for our parameterized algorithm $Alg(S)$. To use the following theorem one needs to prove that $opt(I_S) \leq r_S \cdot opt(I)$ for every instance I of the p-MLC-R problem, where $r_S \geq 1$. Below we discuss how to establish this bound.

Theorem 3.1 *Parameterized algorithm $Alg(S)$ generates for every instance I of the p-MLC-R problem a corridor $T(I)$ with length $t(I)$ at most $2k_S \cdot r_S$ times the length $opt(I)$ of an optimal corridor $OPT(I)$, provided*

that $opt(I_S) \leq r_S \cdot opt(I)$.

Proof: Applying Slavik's approximation algorithm [20, 21] we generate a corridor $T(I_S)$ with length $t(I_S) \leq 2 \cdot k_S \cdot opt(I_S)$. Clearly $T(I_S)$ is also a corridor for I, so the solution generated $T(I)$ is simply $T(I_S)$. By the condition of the theorem, $opt(I_S) \leq r_S \cdot opt(I)$. It then follows that the length of the corridor generated by our parameterized algorithm $Alg(S)$ is $t(I) \leq 2k_S \cdot r_S \cdot opt(I)$.

□

Instead of proving that $opt(I_S) \leq r_S \cdot opt(I)$, for every instance I of the p-MLC-R problem, one establishes a stronger result which is normally easier to prove. That is, prove that for every corridor $T(I)$ with edge length $t(I)$ there is a corridor for I_S denoted by $T(I_S)$ with edge-length $t(I_S) \leq r_S \cdot t(I)$. Applying this to $T(I) = OPT(I)$ we know that there is a corridor $T(I_S)$ such that $t(I_S) \leq r_S \cdot opt(I)$. Since $opt(I_S) \leq t(I_S)$, we know that $opt(I_S) \leq r_S \cdot opt(I)$. For the above approach to yield a constant ratio approximation algorithm we need both k_S and r_S to be constants. For example, when S selects from each rectangle R_i its four corner points, k_S is equal to four. However in order for our parameterized algorithm $Alg(S(4C))$, when $S(4C)$ selects the four corners from each rectangle R_i, to be a constant ratio approximation algorithm for the p-MLC-R problem, we need to show that $opt(I_S) \leq r_S \cdot opt(I)$, for some constant r_S. We discuss several selector functions in the next section. For all of these selector functions, r_S cannot be bounded by a constant, but fortunately r_S can be bounded by a constant for certain selector functions discussed in Section 5 (with k_S bounded by a constant).

4 Selector Functions

In this section we restrict each set of vertices $V(R_i)$ by considering several different basic selector functions S. We show that when incorporating them into our parameterized algorithm $Alg(S)$ do not result in constant ratio approximations. These facts point us in the direction of a method for selecting a set of points where each point is referred to as a *special point*. Special points will turn out to be very important when combined with other critical points to generate constant ratio approximation algorithms (Section 5).

4.1 Selecting the four corners

Consider the function that selects for each rectangle R_i its four corners $C(R_i)$, i.e. the four critical points for R_i are its corner points. We call this selector function $S(4C)$. Clearly, $k_{S(4C)} = 4$. We now show that $r_{S(4C)}$ cannot be bounded above by a constant. Thus the resulting parameterized algorithm $Alg(S(4C))$ is not a constant ratio approximation algorithm. To prove this we give a family of problem instances with parameter j such that $r_{S(4C)}$ is proportional to j, and j can be made arbitrarily large.

Consider the j-layer family of instances $I(j)$ of the p-MLC-R problem represented in the Figure 3. The rectangle F has width 4 and height $\epsilon << 4$. The point p is the top-right corner of F. The rectangle F is partitioned into layers of rectangles. Layer $i = 1$ is formed by two rectangles of size $(\frac{\epsilon}{2} - \delta) \times 2$, above three rectangles with (very tinny) height δ and width 1, 2, and 1, respectively (see the bottom part of Figure 3). The rectangle in layer 1 with height δ and width 2 is colored gray. Layer $i + 1$ consists of two copies of layer i scaled by 50% laid side by side and placed on top of layer i (see Figure 3). In other words, layer 2 has four rectangles of size $(\frac{\epsilon}{4} - \frac{\delta}{2}) \times 1$, above two sets of three rectangles each with (very tinny) height $\frac{\delta}{2}$ and width $\frac{1}{2}$, 1, and $\frac{1}{2}$, respectively. The two rectangles in layer 2 with height $\frac{\delta}{2}$ and width 1 are colored gray. In general, layer $i \geq 1$ has 2^i rectangles of size $(\frac{\epsilon}{2^i} - \frac{\delta}{2^{i-1}}) \times \frac{4}{2^i}$ above 2^{i-1} sets of three rectangles each of (very tinny) height $\frac{\delta}{2^{i-1}}$ and width $\frac{1}{2^{i-1}}$, $\frac{1}{2^{i-2}}$, and $\frac{1}{2^{i-1}}$, respectively. The 2^{i-1} rectangles in layer i with height $\frac{\delta}{2^{i-1}}$ and width $\frac{1}{2^{i-2}}$ are colored gray.

An optimal solution $OPT(I(j))$ of the j-layer family of problem instances $I(j)$ is given by the thick black lines in Figure 3. The total edge-length of the optimal corridor $OPT(I(j))$ is given by

$$opt(I(j)) = 4 + 2(\epsilon - \delta) + 2^0(\frac{\epsilon}{2^0} - \frac{\delta}{2^0}) + 2^1(\frac{\epsilon}{2^1} - \frac{\delta}{2^1}) + 2^2(\frac{\epsilon}{2^2} - \frac{\delta}{2^2}) + \ldots + 2^{j-1}(\frac{\epsilon}{2^{j-1}} - \frac{\delta}{2^{j-1}}).$$

Thus, $opt(I(j)) = 4 + (j + 2)(\epsilon - \delta)$.

It is simple to show that in an optimal solution for $I_{S(4C)}(j)$, denoted by $OPT(I_{S(4C)}(j))$, there must be a segment of length 1 that is needed to connect a corner point of the gray rectangle in layer 1, there are two segments of length $\frac{1}{2}$ needed to connect a corner point of the two gray rectangles in layer 2, and so on. Furthermore all of these segments must be distinct. Therefore, the total length of $OPT(I_{S(4C)}(j))$ is $opt(I_{S(4C)}(j)) > j$ and the ratio $\frac{opt(I_{S(4C)}(j))}{opt(I(j))}$ is greater than $\frac{j}{4+(j+2)(\epsilon-\delta)}$. Making δ and ϵ approach zero, the ratio is proportional to $\frac{j}{4}$, and j can be made arbitrarily large.

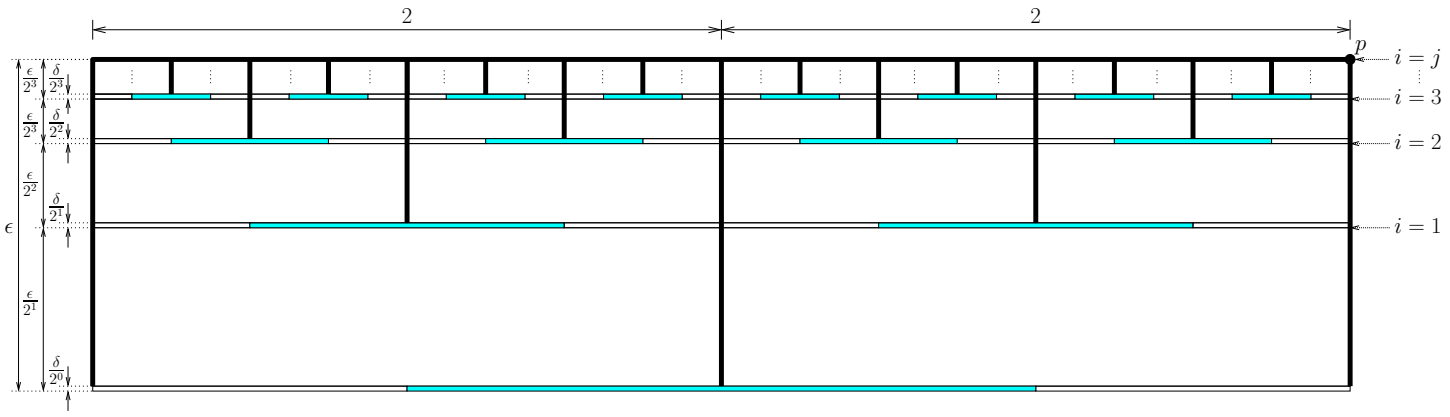


Figure 3: Optimal solution $OPT(I(j))$ for the family of problem instances $I(j)$.

Let $S(F4C)$ be a function that selects fewer corners than $S(4C)$ for each rectangle. It is simple to see that by using the same example, our parameterized algorithm $Alg(S(F4C))$ has an approximation ratio that cannot be bounded above by any constant.

4.2 Selecting k points at random

Randomization is a powerful technique to generate near optimal solution for some problems. Lets apply it to restrict the sets $V(R_i)$. Consider the selector function $S(Rk)$ that selects at most $k \geq 1$ critical points randomly among the vertices of each rectangle. If $k = 7$ the example given in Figure 3 will have $opt(I_{S(Rk)}) = opt(I)$ because every rectangle has at most 7 vertices. One can show that $opt(I_{S(Rk)}) = opt(I)$ holds even when $k = 5$.

We show that the approach of selecting k critical points randomly does not result in an algorithm with an expected approximation ratio bounded above by a constant. Consider the following modification to the problem instance given in Figure 3. We stack k identical rectangles inside the two rectangles placed to the left and right of all the gray rectangles of Figure 3. Figure 4 shows the rectangles added to the left and right of a gray rectangle. The gray rectangles have $2k + 3$ vertices. By selecting k points from each gray rectangle at random (all points being equally likely to be selected), the probability of selecting the middle vertex (see Figure 4) from each gray rectangle is $\frac{\binom{2\cdot k+2}{k-1}}{\binom{2\cdot k+3}{k}} = \frac{k}{2\cdot k+3} < \frac{1}{2}$. So, we expect less than 50% of the gray rectangles to have their middle vertex as a critical point. Therefore, the expected value of $opt(I_{S(Rk)}) \geq \frac{1}{2}j$ and the expected ratio $\frac{opt(I_{S(Rk)})}{opt(I)}$ is at least $\frac{j}{8+(2j+4)(\epsilon-\delta)}$. Making δ and ϵ approach zero, the ratio is proportional to $\frac{j}{8}$, and j can be made arbitrarily large. So our parameterized algorithm $Alg(S(Rk))$ does not generate solutions with

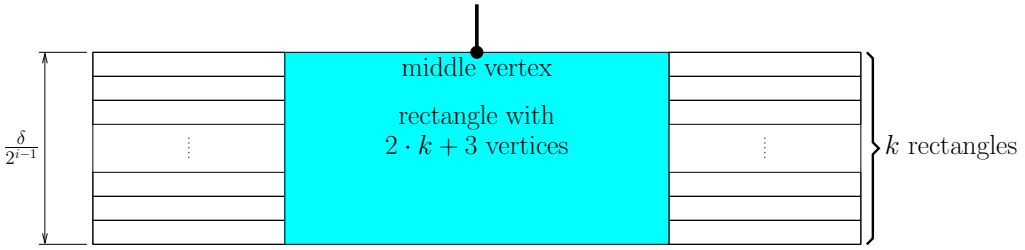


Figure 4: Adding two stacks of k rectangles to the left and right of the gray rectangles, for $k \in \{1, 2, \ldots\}$.

an expected approximation ratio bounded above by a constant.

4.3 Selecting a set of special points

By observation we know that what really helps in the instance given in Figure 3 is for the selector function S to include the middle point as a critical point for every gray rectangle. We call these points *special points*, and formally define them below. For the definition of special point, assume that "rectangle" R_0, which is just point p, is included in R, and p is said to be a corner of R_0. Depending on the selector function S, a subset of the corner points $C(R_i)$ are called the *fixed points* $F(R_i)$ of rectangle R_i. Now, critical points include fixed points and special points.

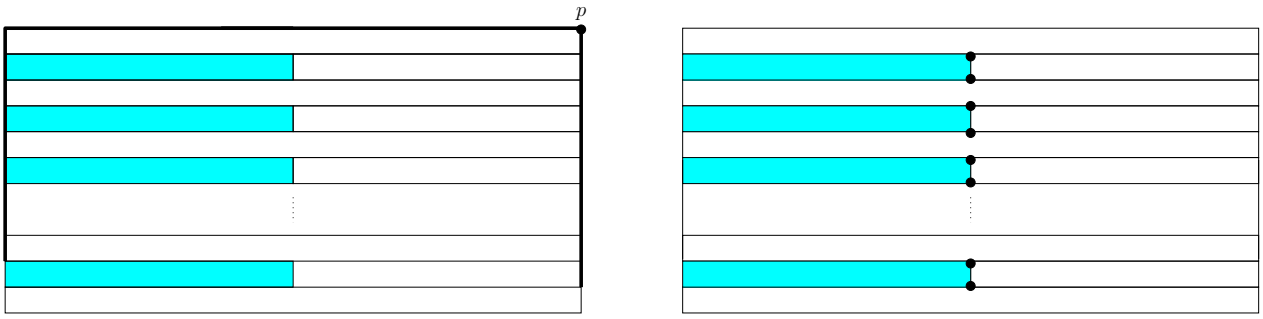
The middle point of each gray rectangle in Figure 3 has the minimum *connectivity distance property*. By this we mean, in very general terms, that if given all partial corridors that do not include a point from rectangle R_i, but include points from all rectangles, then a *special point* is a vertex in R_i, that is not in $F(R_i)$, such that the maximum edge-length needed to connect it to each one of the partial corridors is least possible. Computing this value is in general time consuming. Also, this definition does not work for all problem instances as the set of partial corridors that included vertices from all the rectangles except from R_i may be empty. In what follows we define special points precisely for all problem instances in a way that is computationally easy to identify a set of special points for each rectangle.

Given that we have selected a set $F(R_i)$ of fixed points for each room (rectangle) R_i, we define a special point as follows. Let $u \in V(R_i)$ and let T_u be a tree of shortest paths from u to all other vertices $(\bigcup V(R_j))$ along the edges of rectangle F and the edges of the rooms. Let $SP(u, v)$ be the length of the (shortest) path from vertex u to vertex v along T_u. Let $FP(u, R_j)$ be the length of the (shortest) path from point $u \in V(R_i)$

to the "farthest" vertex of rectangle R_j along T_u, for $i \neq j$, i.e., $FP(u, R_j) = max\{SP(u, v)|v \in V(R_j)\}$. In other words, the edge-length needed to connect vertex u to the corridor through a connection of room R_j to the corridor is at most $FP(u, R_j)$.

We define the *connectivity distance* $CD(u, R)$ of vertex u in room R_i as $min\{FP(u, R_j)|R_j \in R, i \neq j\}$. If $F(R_i) \neq V(R_i)$ we define the connectivity distance $CD(R_i, R)$ of room R_i as $min\{CD(u, R)|u \in V(R_i)\backslash F(R_i)\}$. In other words, $CD(R_i, R)$ is the minimum edge-length needed to connect a vertex in $V(R_i)\backslash F(R_i)$ to the corridor through the connection of another room to the corridor. The special point of R_i is a vertex $u \in V(R_i)\backslash F(R_i)$ such that $CD(u, R) = CD(R_i, R)$. Notice that there may be more than one point for which this condition holds. When $F(R_i) = V(R_i)$ then there is no special point. It is important to remember that for the definition of special point, R_0 which is simply p is included in R.

Consider now the selector function $S(+)$ that selects one special point for each room R_i. The special point for R_i is referred to as SpP_i. In this case, $F(R_i)$ is the empty set. Then for the problem instance given in Figure 3, $opt(I(j)) = opt(I_{S(+)}(j))$. However, as we shall see below, this property does not hold in general. Figure 5 (a) shows an unscaled problem instance I of the p-MLC-R problem where the height of F is very small compared to its width. An optimal solution is represented by the thick black lines in the figure. The



(a) Optimal corridor with length $opt(I)$

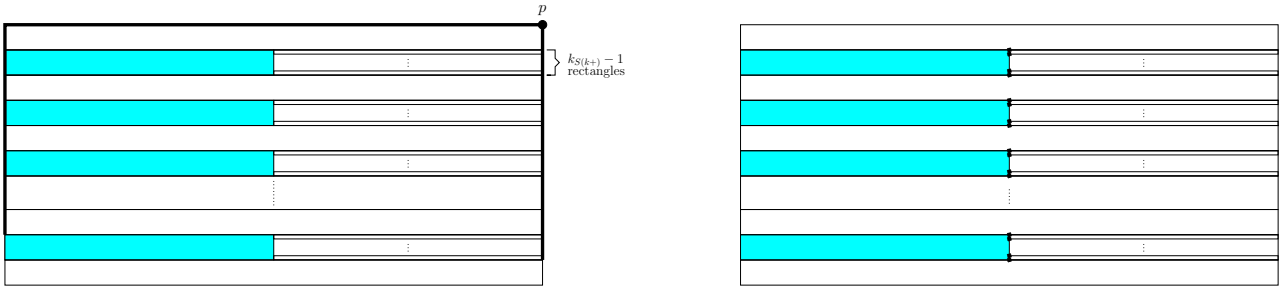
(b) Two candidates of special points for the gray rectangles

Figure 5: Instance I of the p-MLC-R problem.

special point of each gray rectangle is either its top-right or bottom-right corner. An optimal solution for the instance $I_{S(+)}$ must include a segment from a right-most corner of each gray rectangle to the left or right-hand side of rectangle F. Since the height of F is very small with respect to its width, it then follows that the value of $opt(I_{S(+)})$ can be made arbitrarily large compared to $opt(I)$ by adding pairs of gray and white rectangles

following the pattern as in Figure 5. Therefore, $r_{S(+)}$ is not bounded above by any constant. Thus, the restriction to $S(+)$ does not result in constant ratio approximations.

What if the selector function includes at most a given constant number of special points for each rectangle? Let $S(k+)$ and $k_{S(k+)}$ be such selector and constant. In this case, $F(R_i) = \emptyset$. When $|V(R_i)| \leq k_{S(k+)}$, we select all the vertices in $V(R_i)$ as special points, but when $|V(R_i)| > k_{S(k+)}$, we select for each rectangle R_i a subset of $k_{S(k+)}$ vertices with least value of $CD(u, R)$, for $u \in V(R_i)$. Consider the instance given in Figure 6). This instance has a stack of $k_{S(k+)} - 1$ rectangles, for $k_{S(k+)} \in \{2, 3, \ldots\}$, in every white rectangle to the



(a) Optimal corridor with length $opt(I)$

(b) $k_{S(k+)}$ special points for the gray rectangles.

Figure 6: Instance I of the p-MLC-R problem.

right of each gray rectangle in Figure 5. All the $k_{S(k+)}$ special points of the gray rectangles will be on their right-hand side. An optimal solution for $I_{S(k+)}$ must include a segment from a right-most side of each gray rectangle to the left or right-hand side of rectangle F. It follows that $opt(I_{S(k+)})$ can be made arbitrarily large compared to $opt(I)$. Thus $r_{S(k+)}$ is not bounded above by a constant, and our scheme does not result a constant ratio approximation algorithm.

4.4 Selecting two adjacent corners and one special point

The case for $S(2AC+)$ having as critical points two adjacent corners and a special point does not result in a constant ratio approximation , i.e. $k_{S(2AC+)}$ is 3, but $r_{S(2AC+)}$ is not bounded above by a constant. Consider $S(2AC+) = \{TR, BR, SpP\}$, i.e. the top-right and bottom-right corners, and a special point. In this case, $F(R_i) = \{TR_i, BR_i\}$, where TR_i and BR_i are the top-right and bottom-right corners of R_i, respectively. Consider the instances of Figures 5(a) and 6(a) where the critical points of the gray rectangles are on their right-hand side and an optimal solution for each gray rectangle must include a segment from the right-hand side

of each gray rectangle to the left or right-side of rectangle F. Since the height of F is very small compared to its width, it follows that the value of $opt(I_{S(2AC+)})$ can be made arbitrarily large compared to $opt(I)$, and the approximation ratio $r_{S(2A+)}$ is not bounded by a constant. Examples showing that $r_{S(2AC+)}$ is not bounded by a constant for other pairs of adjacent corners and a special point are given in Figures 5(a) and 6(a) when rotated $90, 180$ and 270 degrees. Clearly, the case for S consisting of fewer corners and one special point does not result in constant ratio approximations.

5 Constant Ratio Approximation Algorithms

All the selector functions discussed in the previous section do not result in constant ratio approximations for the parameterized algorithm, but the ones we consider in this section generate constant ratio approximations. These are the first two provably constant ratio approximation algorithms for the p-MLC-R problem and, by Theorem 2.1, for the MLC-R problem.

This approach is motivated by two facts. First, the problem instance given in Figure 3 suggests the selection of at least one special point. Second, the instance given in Figure 5 suggests to include at least two corners of each rectangle R_i. In Section 4.4 we show that two adjacent corners and one special point are not enough to make our parameterized algorithm a constant ratio approximation algorithm, therefore we consider at least two opposite corners and one special point.

We have analyzed the most promising two approaches. The first approach selects two opposite corners and one special point. The second approach selects the four corners of the rectangle and one special point. As it turns out the analysis for the first approach is considerably easier. In Subsection 5.2 we discuss and prove that the first approach results in a constant ratio approximation. In Subsection 5.3 we briefly sketch the analysis for the second approach. In Subsection 5.1 we define new terms that are used extensively throughout this section.

5.1 Preliminaries and Definitions

Let $T(I)$ be a corridor for instance I of the p-MLC-R problem. Now, overlap the corridor $T(I)$ with the rectangles in R. All the rectangles that do not have a critical point along the corridor are called *no-critical-point-exposed* rectangles (*ncpe rectangles*). The remaining rectangles are called *critical-point exposed* rectangles

(*cpe rectangles*). Figure 7 shows an instance I of the p-MLC-R problem with a corridor $T(I)$ and all the ncpe rectangles

We say that the *edges* of the ncpe rectangle n_i are the line segments consisting of $\{BR_i\} \cup P_i^r$, $\{BR_i\} \cup P_i^b$, $\{TL_i\} \cup P_i^l$, or $\{TL_i\} \cup P_i^t$, where TL_i and BR_i are the top-left and bottom-right corners of R_i, respectively; and P_i^r, P_i^b, P_i^l and P_i^t are the right, bottom, left, and top side of R_i, respectively. Note that P_i^r, P_i^b, P_i^l and P_i^t do not include the corners of R_i.

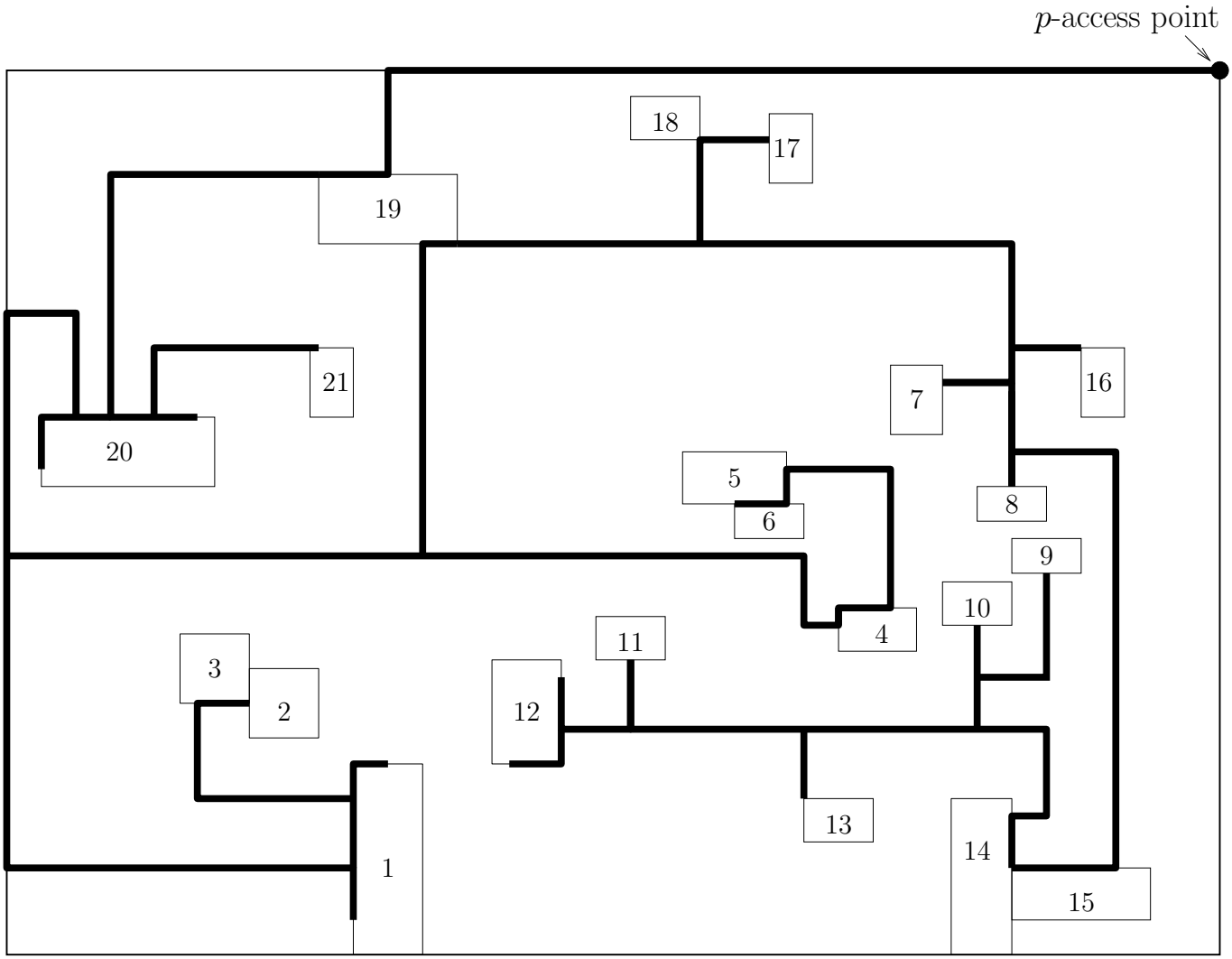


Figure 7: Corridor and ncpe rectangles.

Let τ be the counter-clockwise tour of corridor $T(I)$. This tour starts at the access point p moving in the counter- clockwise direction with respect to p on the exterior of F. Then τ traverses the edges of the corridor without crossing it and ends again at the access point p as shown in Figure 8. Note that there is also

a clockwise tour that traverse the corridor in the opposite direction of τ.

In Figure 9(a), τ visits first all the rectangles located below and to the right of $T(I)$ and then the ones above and to the left of $T(I)$. As corridor becomes complex, we cannot have a consistent pattern of "above" before "below" or vice versa. Consider the tour of Figure 9(b) beginning on the top part of the access point in the direction of the arrow. Rectangle R_i is visited before rectangle R_j. The first point where R_i is visited by τ is w_i and the first point where R_j is visited by τ is w_j. The labels in Figure 7 denote the order in which the ncpe's are visited when traversing the counter-clockwise tour given in Figure 8.

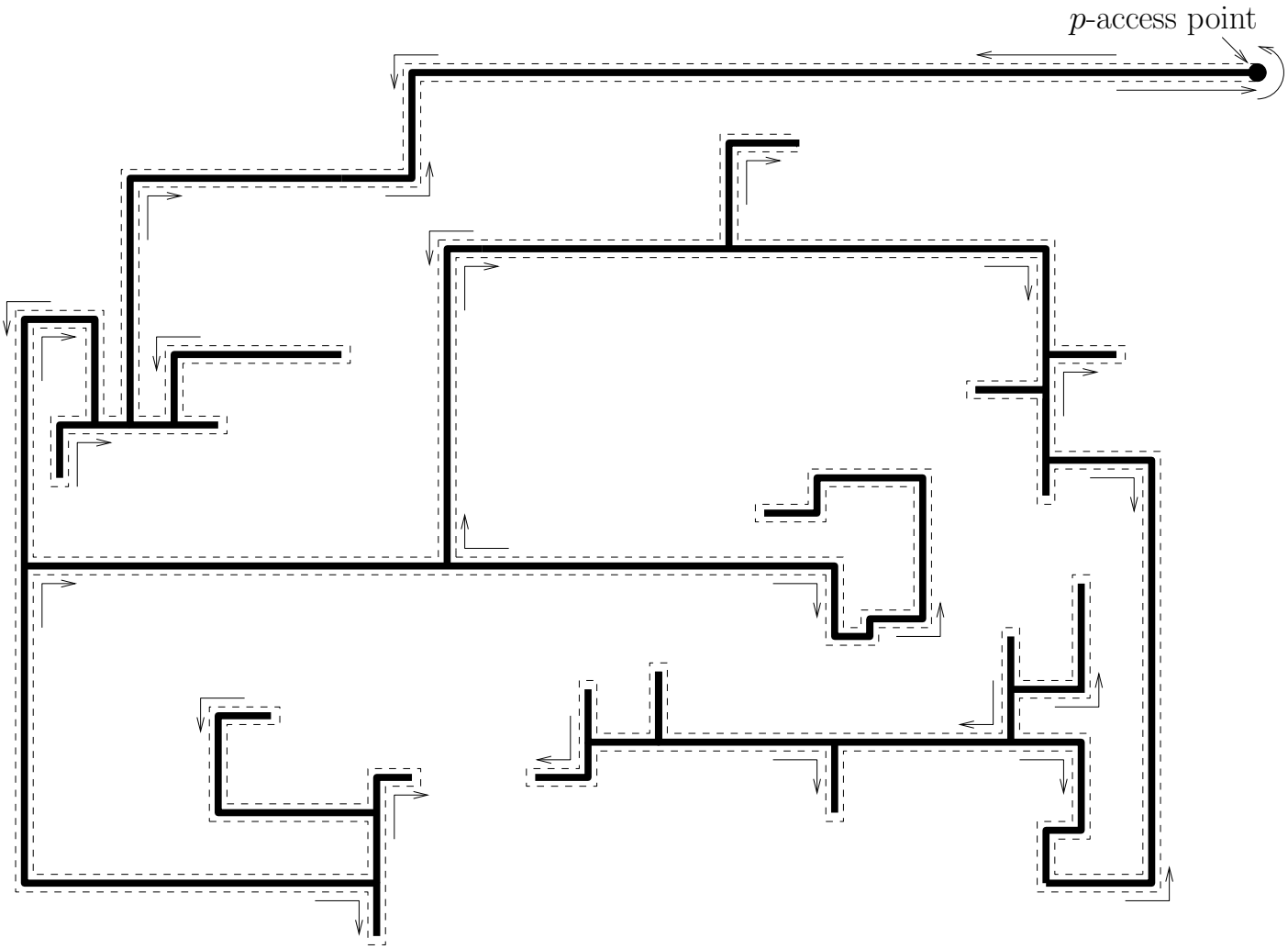


Figure 8: Pre-order traversal tour of the corridor $T(I)$.

Lets mark the tour τ at the first point where each ncpe rectangle is visited for the first time. The order in which the ncpe rectangles are first visited is called the *canonical order* of the ncpe rectangles. Rename the

ncpe rectangles according to this canonical order as $VR = \{n_1, n_2, \ldots, n_q\}$. For example, in Figure 9(b), n_1 is R_i and n_2 is R_j; and in Figure 7, the ncpe rectangles are labeled in the order they are visited by tour τ. For $1 \leq i \leq q$, we define the *reaching point* Y_i' of the ncpe rectangle n_i as the first point in the ncpe rectangle n_i visited by the tour τ; and the *leaving point* X_i' of the ncpe rectangle n_i as the last point in the ncpe rectangle n_i visited by τ before it visits the reaching point Y_{i+1}' of rectangle n_{i+1} (see Figure 10). For convenience we add the points X_0' and Y_{q+1}' which correspond to the access point p.

Let $\tau(Z_1', Z_2')$ be the path along tour τ from the point Z_1' to the point Z_2', where $Z_1', Z_2' \in \{X_0', X_1', \ldots, X_q', Y_1', \ldots, Y_{q+1}'\}$ and Z_1' appears before Z_2' in the tour τ. For $0 \leq i \leq q$, let l_i be the length of the path $\tau(X_i', Y_{i+1}')$. Let h_j be the length of the path $\tau(Y_j', X_j')$, for $1 \leq j \leq q$. Figure 9(a) and 10 show line segments of length l_i and h_j for a set of ncpe rectangles.

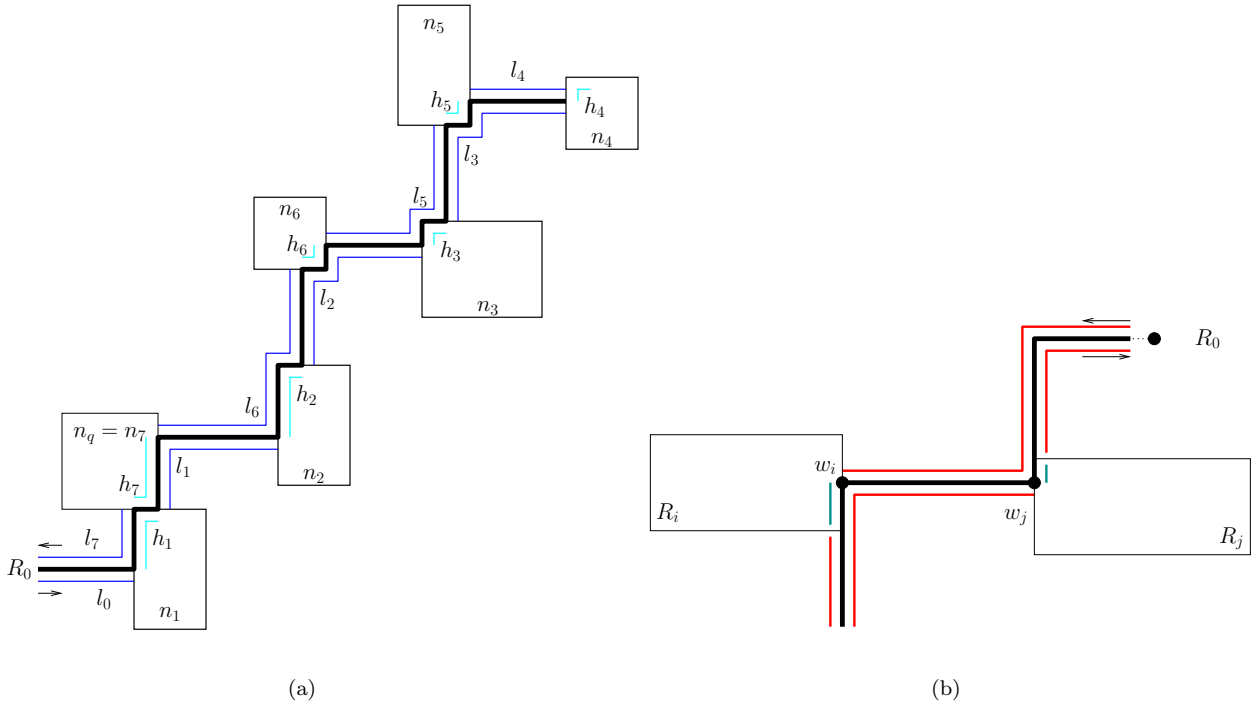
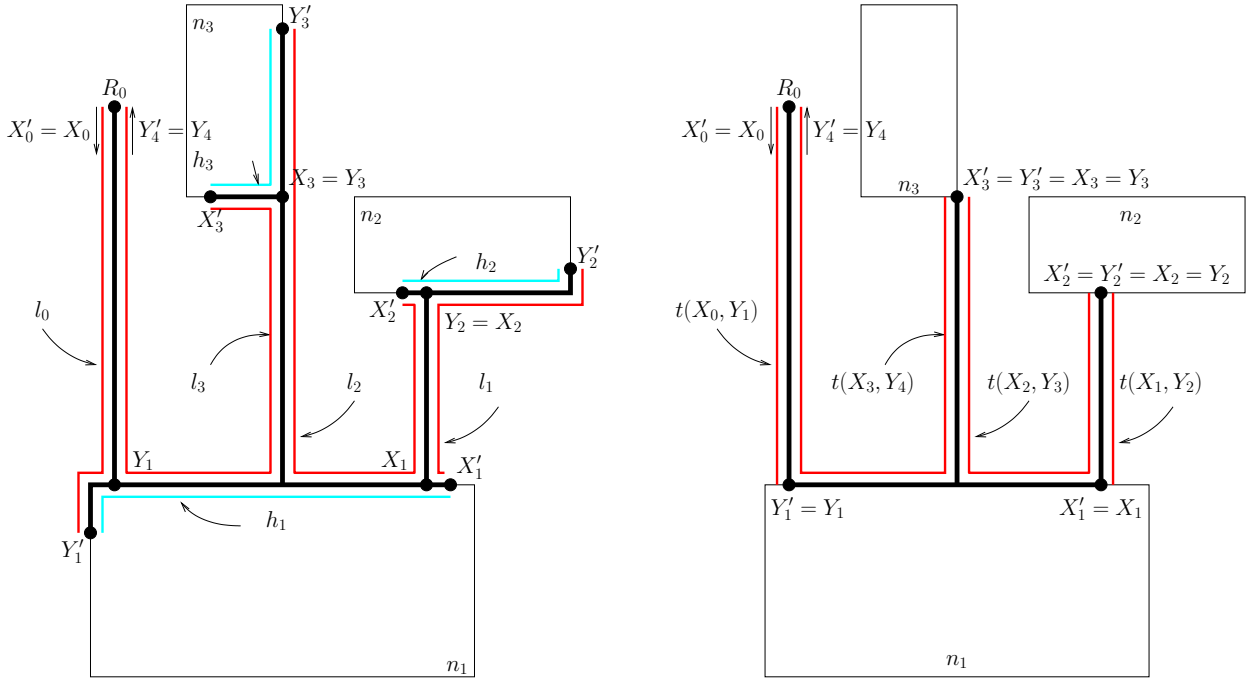


Figure 9: The line segments of length l_i and h_j for a set of ncpe rectangles.

By using the shortest path along the corridor $T(I)$ from n_i to n_{i+1}, we define the *exit point* X_i of ncpe n_i and the *entry point* Y_{i+1} of n_{i+1} as the intersection of the shortest path with the edges of n_i and n_{i+1}, respectively. For convenience we also add the points X_0 and Y_{q+1} which correspond to the access point p. For $i < j$ we define $T(X_i, Y_j)$ as the shortest path along the corridor T from the exit point X_i to the entry

point Y_j, and $t(X_i, Y_j)$ denotes its total length. It is simple to prove that $t(X_i, Y_{i+1}) \leq l_i$ for $0 \leq i \leq q$, and $t(Y_i, X_i) \leq h_i$ for $1 \leq i \leq q$ (see Figure 10).



(a) X_i and X_i', Y_i and Y_i' do not match, except when $i \in \{0, 4\}$.

(b) X_i matches X_i', and Y_i matches Y_i', for $0 \leq i \leq 4$.

Figure 10: Examples of corridors for three ncpe rectangles.

5.2 Selecting two opposite corners and one special point

In this subsection we analyze the parameterized algorithm $Alg(S)$ when each set $V(R_i)$ is restricted to two opposite corners and one special point by the selector function $S(2OC+)$. Without loss of generality, we select the top-right and bottom-left corners as the two opposite corners. Clearly $k_{S(2OC+)} = 3$, and we prove that $r_{S(2OC+)} = 5$. By Theorem 3.1 this results in a parameterized algorithm having the approximation ratio 30. Before we present our analysis, we need to define several terms.

We establish in Theorem 5.1 that the parameterized algorithm with $S(2OC+)$ generates a solution with length at most $30 \cdot opt(I)$. We take a top-down approach to prove this theorem. In Theorem 5.2, which we establish below, we show that given any corridor $T(I)$ with edge-length $t(I)$, there is a corridor $T(I_{S(2OC+)})$ with length $t(I_{S(2OC+)}) < 5 \cdot t(I)$. Applying the theorem to $OPT(I)$ we know there is a corridor $T(I_{S(2OC+)})$ with length $t(I_{S(2OC+)}) < 5 \cdot opt(I)$. Since $opt(I_{S(2OC+)}) \leq t(I_{S(2OC+)})$ it follows that $opt(I_{S(2OC+)}) < 5 \cdot opt(I)$.

Applying Theorem 3.1, the result follows.

Theorem 5.1 *For every instance I of the p-MLC-R problem, the parameterized algorithm $Alg(S(2OC+))$ generates, in polynomial time with respect to the number of rectangles r, a corridor with length at most $30 \cdot opt(I)$.*

Proof: By the above discussion. The time complexity of the algorithm is the one for Slavik's algorithm [20, 21], which is bounded by a polynomial in terms of the number of rectangles.

□

Now, applying Theorem 2.1 we know that selecting the best corridor generated by the parameterized algorithm $Alg(S(2OC+))$, when executing it with every point p being a vertex located along the boundary of F, results in a constant ratio approximation algorithm for the MLC-R problem. We call this algorithm the general parameterized algorithm $ALG(S(2OC+))$.

Corollary 5.1 *The general parameterized algorithm $ALG(S(2OC+))$ generates a corridor with length at most $30 \cdot ov(I)$ for every instance I of the MLC-R problem, where $ov(I)$ is the edge-length of a minimum length corridor for problem instance I.*

Proof: By the above discussion.

□

Our proof technique is simple, but it involves quite a few details. The idea is to show that for each ncpe rectangle n_i one can connect one of its critical points to the corridor by adding line segments of length at most $l_{i-1} + h_i + l_i$. We establish this claim in Lemma 5.1 which make use of the subsequent lemmas. In Theorem 5.2 we show this is enough in order to establish our approximation bound.

Theorem 5.2 *Given any corridor $T(I)$ with length $t(I)$ for any instance I of the p-MLC-R problem, there is a corridor for instance $I_{S(2OC+)}$ of the p-MLC-$R_{S(2OC+)}$ problem with edge-length $t(I_{S(2OC+)}) \leq 5 \cdot t(I)$.*

Proof: By Lemma 5.1, which we establish below, the corridor $T(I)$ can be extended to reach a critical point of each ncpe rectangle by adding a set of line segments of length at most $l_0 + \sum_{j=1}^{q} h_j + 2\sum_{j=1}^{q-1} l_j + l_q < 4t(I)$, since $\sum_{j=0}^{q} l_j + \sum_{j=1}^{q} h_j = |\tau| = 2t(I)$. Therefore, the total length of the additional line segments plus the length of the corridor T is less than $5t(I)$. So we have constructed a corridor $T(I_{S(2OC+)})$ with edge-length

$t(I_{S(2OC+)}) < 5t(I)$.

□

To establish our main result we need to prove Lemma 5.1.

Lemma 5.1 *For $1 \leq i \leq q$, it is possible to connect at least one of the critical points of every ncpe rectangle $n_i \in VR$ to the corridor, by adding line segments of length at most $l_{i-1} + h_i + l_i$.*

Proof: By definition of special points, $CD(SpP_1, R) = CD(n_1, R) \leq l_0$ and $CD(SpP_q, R) = CD(n_q, R) \leq l_q$. Therefore, joining SpP_1 (SpP_q) to its nearest point in the corridor requires line segments of length at most l_0 (respectively l_q).

In Lemmas 5.5, 5.6, and 5.8 below we show that every ncpe rectangle n_i for $2 \leq i \leq q-1$ has the property that it can be joined to the corridor by connecting one of its special points to the nearest point in the corridor by adding line segments of length at most $l_{i-1} + h_i + l_i$. The lemma follows from Lemmas 5.5, 5.6, and 5.8.

□

The idea behind the proof of the remaining lemmas is to consider the paths $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$. These paths will be labeled type-1 or type-2 depending on some properties that we define below. If $T(X_{i-1}, Y_i)$ (resp. $T(X_{i-1}, Y_{i+1})$) is type-1 then in Lemma 5.5 (resp. 5.6) we show that a critical point in n_i can be connected to the corridor by adding line segment of length at most l_{i-1} (resp. $l_{i-1} + h_i + l_i$). For the remaining case, we characterize the form of paths $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ when they are of type-2 in Lemma 5.7. Then in Lemma 5.8 we use this characterization to show that the path $T(Y_i, Y_{i+1})$ is type-1, and we show that a critical point of n_i can be connected to the corridor by adding line segments of length at most $l_{i-1} + h_i + l_i$. In order to prove these lemmas we need to introduce additional notation and establish (preliminary) Lemmas 5.2-5.4.

First, we define a special type of turns in paths as the relevant combinatorial structure to be exploited, in order to establish our approximation bounds. This is called an *inversion of direction* (or simply iod-subpaths) of a path $T(X_i, Y_j)$. Iod-subpaths are vertical or horizontal. We then define the beam of an iod-subpath as the set of points that are "visible" from an iod-subpath. In Lemma 5.2 we show that if a path has a vertical and a horizontal iod-subpath, then it has two adjacent ones whose beams intersect. When this condition holds, Lemma 5.3 shows that there is at least one rectangle in R that is completely inside the region of the adjacent

iod-subpaths. This latter property is used to establish that from either end of the path that includes the adjacent iod-subpaths, or from a point which we call the bifurcation point, one can reach any point in one of the rectangles located completely inside the region of the iod-subpath by using line segments of length at most $t(X_i, Y_j)$. Then we use this fact to show that the length of the path from a vertex of n_i to a rectangle in R is at most $l_i + h_i + l_{i+1}$, and by using the definition of a special point, we show the special point of n_i can be connected to the corridor by using line segments of length at most $l_i + h_i + l_{i+1}$.

Before we formulate Lemmas 5.5-5.7 we need to introduce additional notation and establish preliminaries results. As we traverse the path $T(X_i, Y_j)$ from X_i to Y_j we identify a sequence of alternating horizontal and vertical line segments, $s_1, s_2, \ldots, s_l$, such that the first endpoint of s_1 visited is X_i; the last endpoint of s_{k-1} visited coincides with the first endpoint of s_k, for $1 < k \leq l$; and the last endpoint of s_l visited is Y_j. The length of each of these segments is greater than zero, except when $X_i = Y_j$.

The line segments s_{k-1} and s_k, for $1 < k < l$, in path $T(X_i, Y_j)$ are said to be *adjacent*. For the path $T(X_i, Y_j)$ we say that the line segment s_1 is *adjacent* to the edge of ncpe n_i where X_i is located if s_1 intersects the edge, and s_1 and the edge are perpendicular. We define adjacency for s_l and an edge of n_j similarly.

An *inversion of direction subpath* (or *iod-subpath*) of a path $T(X_i, Y_j)$ and the ncpe's n_i and n_j, consists of a line segment s_k of the path and on each of its two ends there is either an adjacent edge of ncpe n_i or n_j, or an adjacent line segment s_{k-1} or s_{k+1}, and adjacent segments and/or portions of the edges must be on the same side of s_k (i.e., the segments and/or portions of the edges are on the same side of the line that completely includes s_k). The line segment s_k of an iod-subpath is called the *central segment* and the two adjacent line segments and/or edges are called the *end segments.* If the end segments are vertical then it is a vertical iod-subpath, otherwise it is a horizontal iod-subpath. Figure 11 shows a vertical (horizontal) iod-subpath labeled p_v (p_h).

The subpath $P'(p_v, p_h)$ between the two iod-subpaths p_v and p_h of a path $T(X_i, Y_j)$ consists of all the line segments in the path between the farthest (with respect to the distance along the path) endpoints of their central line segments (path from x_0 to x_{k+1} in Figure 11). Two iod-subpaths of a path are said to be *adjacent* if the subpath $P'(p_v, p_h)$ between them does not contain another iod-subpath.

A vertical iod-subpath p_v is said to be to the left of a horizontal iod-subpath p_h, if its central segment is completely located to the left of the vertical line that completely includes the central line segment of p_h.

A vertical iod-subpath with portions of its two vertical end segments above its central line segment is called an *up-vertical* iod-subpath. If portions of its two vertical end segments are below its central line segment, the vertical iod-subpath is called *down-vertical* iod-subpath. We define *right-horizontal* or *left-horizontal* iod-subpath similarly. The *beam* of an iod-subpath consists of all the points visited by the central line segment as we move it perpendicularly and in the direction of the end segments of the iod-subpath until it reaches the rectangular boundary F.

We say that point x precedes the point y along a path $P'(p_v, p_h)$, denoted by $x \prec y$, if x is visited before y when traversing the path $P'(p_v, p_h)$ from p_v to p_h. We now establish a fundamental lemma which will enable us to simplify the proofs of subsequent lemmas.

Lemma 5.2 *If a path (a, b) contains vertical and horizontal iod-subpaths, then it contains a vertical iod-subpath adjacent to a horizontal iod-subpath, and their beams intersect.*

Proof: Suppose we traverse the path from a to b. Since the path contains vertical and horizontal iod-subpaths, let p_v^{first} be the first vertical iod-subpath, and p_h^{first} be the first horizontal iod-subpath. Without loss of generality, assume that p_v^{first} precedes p_h^{first}. Let p_v^{last} be the last vertical iod-subpath that precedes p_h^{first}. Assume without loss of generality that p_v^{last} is an up-vertical iod-subpath, and that as we traverse the path from a to b we will exit p_v^{last} through its right end. Consider the path $P'(p_v, p_h)$. Since there are no iod-subpaths between p_v^{last} and p_h^{first} it must be that as we traverse the path from p_v^{last} to p_h^{first} the horizontal segments are traversed left to right and the vertical ones are traversed in the upwards direction, as otherwise there would be a vertical or horizontal iod-subpath before p_h^{first}, which contradicts our earlier assumptions. This means that p_h^{first} is a left-horizontal iod-subpath and the beams of p_v^{last} and p_h^{first} intersect.

□

A pair of adjacent vertical p_v and horizontal p_h iod-subpaths whose beams intersect are said to be in *canonical form* if p_v is an up-vertical iod-subpath located to the left of p_h. A proof similar to Lemma 5.2 can be used to show that as we traverse the path $P'(p_v, p_h)$ from p_v to p_h one only moves up and to the right (see Figure 11). Let L be the left-most vertical line traced by the beam of p_v, and let T be the top-most horizontal line traced by the beam of p_h. The area delineated by $L, P'(p_v, p_h)$ and T is called the region of the canonical iod-subpaths (p_v-p_h) and it is specified by $r(p_v, p_h)$ (see Figure 11).

We say that a rectangle $R_i \in R$ is *contained* by $r(p_v, p_h)$ if it is completely inside $r(p_v, p_h)$. A rectangle $Z_b \in R$ ($Z_r \in R$) is said to be *bottom (right) contained* if Z_b (Z_r) is contained by $r(p_v, p_h)$ and its bottom (right) edge is completely contained by a horizontal (vertical) line segment of the path $P'(p_v, p_h)$.

Traverse the path $P'(p_v, p_h)$ from p_v to p_h. Label the first point x_0; label each corner point visited as $c_0, x_1, c_1, x_2, \ldots, c_{k-1}, x_k, c_k$; and label the last point x_{k+1} (see Figure 11). Let $CP = \{c_0, c_1, \ldots, c_k\}$ and let $XP = \{x_1, x_2, \ldots, x_k\}$. For $0 \leq i \leq k$, let $Z_i \in R$ be the rectangle that has its bottom-right corner on the corner point $c_i \in CP$.

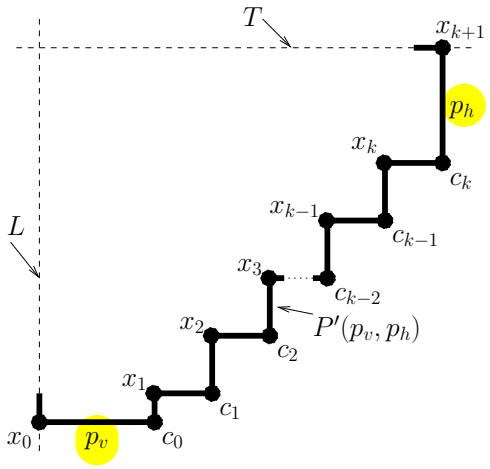


Figure 11: Pair of adjacent iod-subpaths.

Lemma 5.3 *Given any pair of adjacent vertical (p_v) and horizontal (p_h) iod-subpath in canonical form, there is a bottom-side contained rectangle $Z_b \in R$ and a right-side contained rectangle $Z_r \in R$.*

Proof: Consider first the case when Z_0 has its top-edge above line T. It follows that Z_k has its left edge inside the region $r(p_v, p_h)$, and Z_k is a right-side contained rectangle (i.e., Z_r). If Z_k is bottom-side contained then Z_k is Z_b and Z_r, and the lemma follows (See Figure 12(a)).

Otherwise, Z_k is not a bottom-side contained rectangle, and its bottom side extends to the left of the corner point x_k (see Figure 12(b)). Let k' be the largest positive integer such that the bottom side of Z_l for $l \leq k'$ is completely contained by a horizontal segment of $P'(p_v, p_h)$. Clearly such an l exists as the property holds for Z_0 but does not for Z_k. Since $Z_{k'+1}$ extends to the left of the vertical line with the x-coordinate value of $c_{k'}$, it follows that $Z_{k'}$ is in the region $r(p_v, p_h)$. Thus, $Z_{k'}$ is right-side and bottom-right contained (See Figure 12(b)). Therefore both types of contained rectangles exist in $r(p_v, p_h)$.

Similar arguments can be used for the case when the top-edge of Z_0 is not above line T.

□

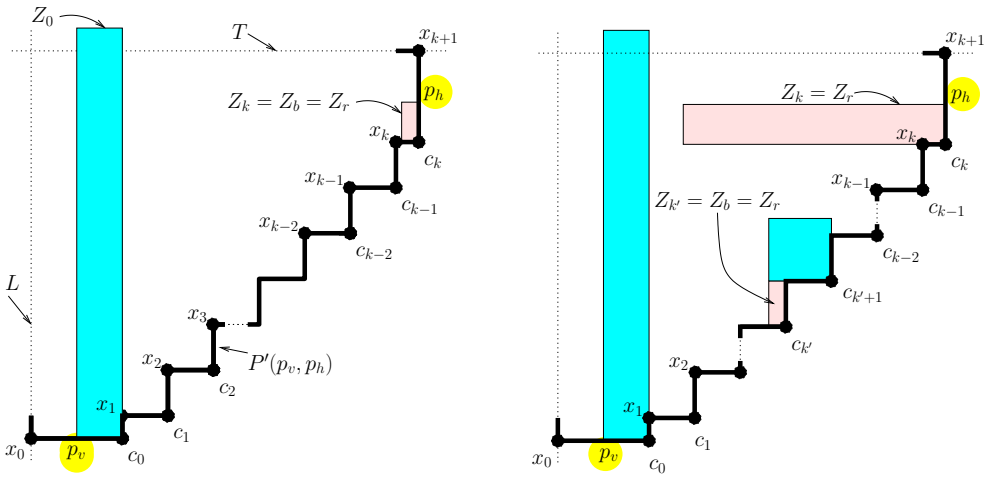


Figure 12: Bottom-side and right-side contained rectangles Z_b and Z_r.

Lemma 5.4 *For a pair of adjacent vertical p_v and horizontal p_h iod-subpaths in canonical form, and three points $x_i \prec X \prec x_{i+1}$ along $P'(p_v, p_h)$, where $x_i, x_{i+1} \in XP$, there exists both a bottom-side contained rectangle Z_b and a right-side contained rectangle Z_r either (a) along the path from p_v to x_{i+1}, or (b) along the path from x_i to p_h. Furthermore, these rectangles have their bottom-right corner point coincide with a point in CP.*

Proof: Consider the corner point $c_i \in CP$ such that $x_i \prec c_i \prec x_{i+1}$. It is easy to see that if Z_i has its top and left edges bounded by horizontal and vertical lines overlapping the points x_{i+1} and x_i, respectively, then Z_i is vertically and horizontally contained, and it is along both paths from p_v to x_{i+1} and from x_i to p_h (see Figure 13(a)). Assume now that the top edge of rectangle Z_i is above the horizontal line overlapping the point x_{i+1}. It then follows that the right edge of rectangle Z_i and the path from x_{i+1} to p_h produce an up iod-subpath [2] (see Figure 13(b)). Therefore, both horizontal and vertical iod-subpaths exist along the path from x_{i+1} to p_h. By Lemma 5.3 we know that there exist both a bottom-side and a right-side contained rectangles. Furthermore, these rectangles are on the path from x_{i+1} to p_h and consequently on the path from

[2]Note that since Z_i is not an ncpe, the right edge of Z_i and the path from x_{i+1} to p_h is not technically an iod-subpath. However, iod-subpaths may also be defined with cpe rectangles and have the same properties which are needed to prove the lemma.

x_i to p_h. Therefore lemma follows. The remaining case is when the left edge of rectangle Z_i is on the left side of the vertical line overlapping the point x_i (see Figure 13(c)). Similar arguments to the previous case can be used to show that there exist both horizontally and vertically contained rectangles on the path from p_v to x_i and consequently on the path from p_v to x_{i+1}, and the lemma follows.

□

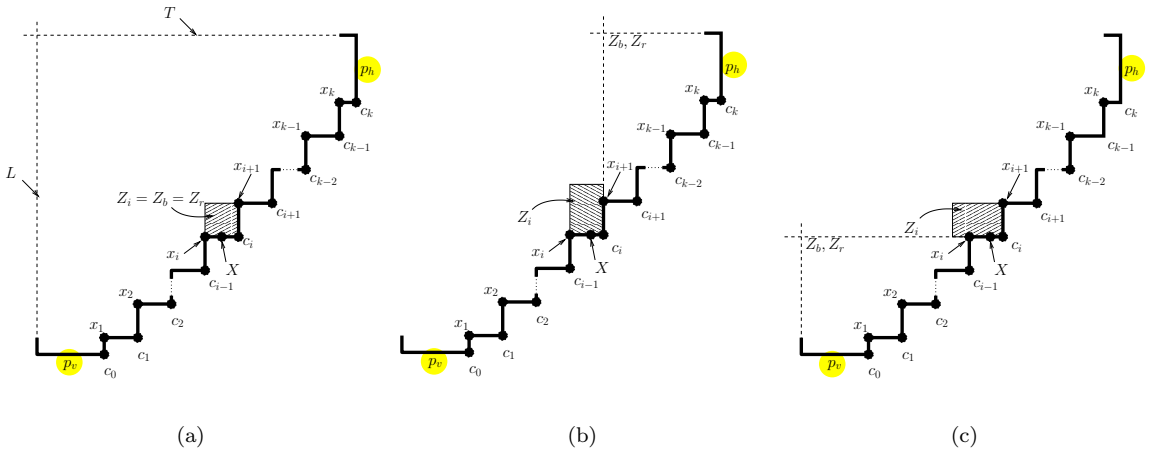


Figure 13: Bottom-side and right-side contained rectangles with respect to the points $x_i \prec X \prec x_{i+1}$.

Consider the ncpe rectangles n_{i-1}, n_i, and n_{i+1}. By reflexion (with respect to the axis x and y, intersecting the central point of n_i) and rotations (of $90, 180$ and 270 degrees with respect to the axis that is perpendicular to the plane passing through the central point of n_i), we know we only need to consider the case when the exit point X_{i-1} either belongs to P^r_{i-1} or is the BR_{i-1} corner. Therefore, in what follows when we consider a path starting at X_{i-1}, we assume that X_{i-1} is located at $P^r_{i-1} \cup BR_{i-1}$ A path is said to be of *type-1* if it contains vertical and horizontal iod-subpaths, and *type-2* otherwise. Lemmas 5.5, 5.6, and 5.8 show that a critical point of n_i can be connected to the corridor by adding line segments of length at most $l_{i-1} + h_i + l_i$. These are the three lemmas needed to establish Lemma 5.1 and our main result.

Lemma 5.5 *If path $T(X_{i-1}, Y_i)$ is type-1, then a critical point of n_i can be connected to the corridor by adding line segments of length at most l_{i-1}.*

Proof: Since path $T(X_{i-1}, Y_i)$ is type-1 then by Lemma 5.2 we know it has a pair of adjacent vertical and horizontal iod-subpaths (p_v and p_h) whose beams intersect and are in canonical form. By Lemma 5.3, the region $r(p_v, p_h)$ has a bottom-side contained rectangle $Z_b \in R$ and a right-side contained rectangle $Z_r \in R$ (see

Figure 14). We define $FP_c(x_i, Z_j)$ as the length of the path along a restricted set of edges given by c from x_i to the farthest point in rectangle Z_j. Let c be the restricted set of edges $T(X_{i-1}, Y_i)$ plus the side of Z_r. By definition $FP(Y_i, Z_r) \leq FP_c(Y_i, Z_r)$. By projecting the bottom edge of Z_r to the corridor $T(X_{i-1}, Y_i)$, we know that $FP_c(Y_i, Z_r) \leq t(X_{i-1}, Y_i)$ which earlier we established it is at most l_{i-1}. Therefore, $FP(Y_i, Z_r) \leq l_{i-1}$. By the definition of a special point, the special point of n_i can be connected to its closest point of the corridor by segments of length at most $FP(Y_i, Z_r) \leq l_{i-1}$ (see figure 14).

□

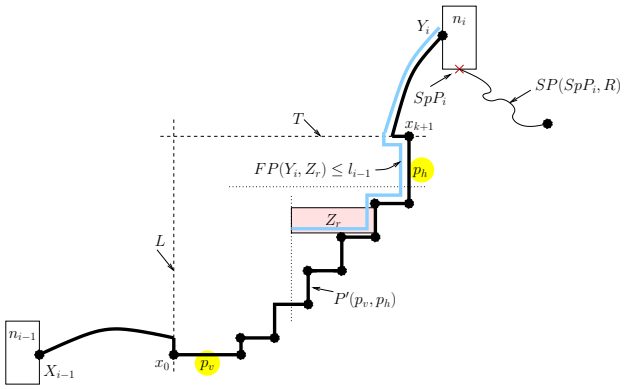


Figure 14: $T(X_{i-1}, Y_i)$ type-1.

Lemma 5.6 *If path $T(X_{i-1}, Y_{i+1})$ is type-1, then a critical point in n_i can be connected to the corridor by adding line segments of length at most $l_{i-1} + h_i + l_i$.*

Proof: Let X be a point along the path $T(X_{i-1}, Y_i)$ that is the bifurcation point of the paths $T(X_{i-1}, Y_{i+1})$ and $T(X_{i-1}, Y_i)$. Since $T(X_{i-1}, Y_{i+1})$ is type-1, then applying similar arguments as in the proof of Lemma 5.5, we know that the path $T(X_{i-1}, Y_{i+1})$ has a pair of adjacent vertical and horizontal iod-subpaths (p_v and p_h) whose beams intersect. By Lemma 5.3, the region $r(p_v, p_h)$ has a bottom-side contained rectangle $Z_b \in R$ and a right-side contained rectangle $Z_r \in R$ (see Figure 15 and 16). If the bifurcation point is located in the portion of the path $T(X_{i-1}, Y_{i+1})$ that is not included in the path from x_0 to x_{k+1}, then a proof similar to the one for Lemma 5.5 can be used to show that the special point of n_i can be connected to its closest point in the corridor by segments of length at most $FP(X_i, Z_b) \leq l_i$ (Figure 15), or $FP(X_i, Z_r) \leq l_{i-1}$ (Figure 16). Otherwise, X is in the path from x_0 to x_{k+1}. Suppose that X is in between x_j and x_{j+1} in XP. By Lemma 5.4 we know that there is a bottom-side contained rectangle $Z_b \in R$ and a right-side contained rectangle $Z_r \in R$ in the path from

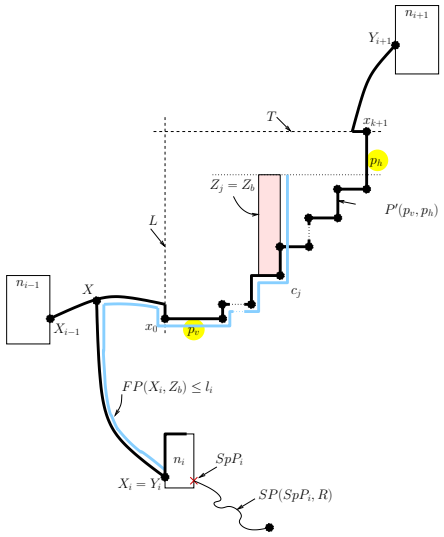


Figure 15: Path $T(X_{i-1}, Y_{i+1})$ of type-1: X is on the left side of subpath $P'(p_v, p_h)$.

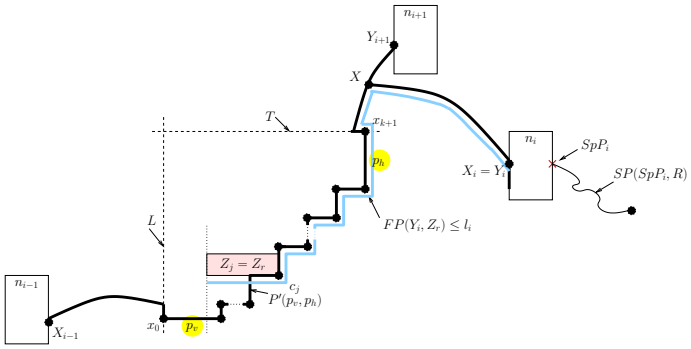


Figure 16: Path $T(X_{i-1}, Y_{i+1})$ of type-1: X is on the right side of subpath $P'(p_v, p_h)$.

x_0 to x_{j+1}, or in the path from x_j to x_{k+1}. If there is a bottom-side contained rectangle $Z_j = Z_b$ ($Z_j = Z_r$) whose bottom-side (right-side) is between a pair of corners $c_{j+1}, \ldots, c_k$ ($c_0, \ldots, c_{j-1}$), then arguments similar to the ones for Lemma 5.5 can be used to show that $FP(X_i, Z_b) \leq l_i$ ($FP(X_i, Z_r) \leq l_{i-1}$), and therefore the special point of n_i can be connected to its closest point in the corridor by a set of line segments of length at most $FP(X_i, Z_b) \leq l_i$ ($FP(X_i, Z_r) \leq l_{i-1}$). One can show that the only remaining case is when the rectangle Z_j (the one with bottom-right corner at c_j) is both a bottom and right-side contained rectangle. In this case we can easily show that $FP(X_i, Z_j) \leq l_{i-1} + h_i + l_i$. Therefore the special point of n_i can be connected to its closest point in the corridor by a set of line segments of length at most $FP(X_i, Z_j) \leq l_{i-1} + h_i + l_i$ (see Figure 17). This concludes the proof of the lemma.

□

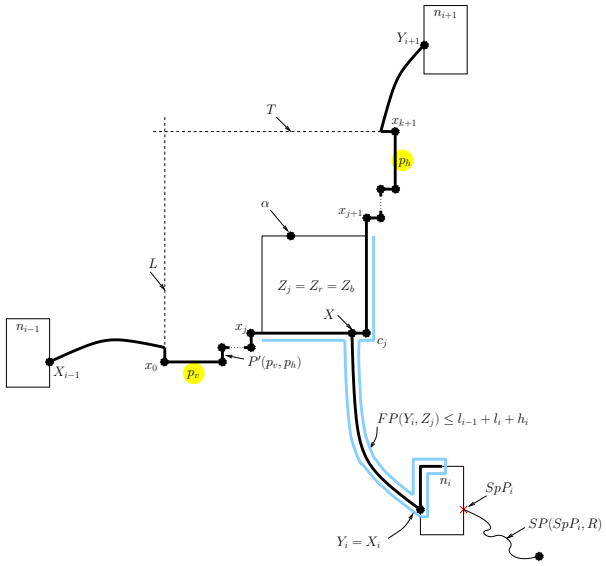


Figure 17: Path $T(X_{i-1}, Y_{i+1})$ of type-1: X is along the subpath $P'(p_v, p_h)$.

The remaining case is when both $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ are paths of type-2. By definition, path $T(X_{i-1}, Y_j)$ for $j \in \{i, i+1\}$ is type-2 and it does not have both a vertical and a horizontal iod-subpath. We call such path type-V, type-H, or type-N, depending on whether it has only vertical iod-subpaths, only horizontal iod-subpaths, or no iod-subpaths, respectively. A path type-V is called type-VMR if when traversing each segment of the path from X_{i-1} to Y_j it takes us up, down or right. A path type-H is called type-HMD if when traversing each segment of the path from X_{i-1} to Y_j it takes us left, right or down. A path type-N is called type-ND if when traversing each segment of the path from X_{i-1} to Y_j it takes us right or down and $X_{i-1} \neq Y_i$; and type-ND$_1$, if $X_{i-1} = Y_i$.

Lemma 5.7 *A type-2 path $T(X_{i-1}, Y_j)$, for $j \in \{i, i+1\}$, is of one of the following forms:*

1. *Path $T(X_{i-1}, Y_j)$ is of type-VMR, and the entry point $Y_j \in P_j^l \cup \{TL_j\}$.*
2. *Path $T(X_{i-1}, Y_j)$ is of type-HMD, $X_{i-1} = BR_{i-1}$, and the entry point $Y_j \in P_j^t \cup \{TL_j\}$.*
3. *Path $T(X_{i-1}, Y_j)$ is of type-ND, $X_{i-1} = BR_{i-1}$ and the entry point $Y_j = TL_j$.*
4. *Path $T(X_{i-1}, Y_j)$ is of type-ND$_1$, consists of only one point, $X_{i-1} = Y_j$ and $X_{i-1} \in P_{i-1}^r \cup \{BR_{i-1}\}$.*

Proof: There are four cases depending on the type of path $T(X_{i-1}, Y_j)$.

Case 1: Path $T(X_{i-1}, Y_j)$ is type-V.

By definition the path has vertical iod-subpaths. While traversing the path from X_{i-1} to Y_j, if a horizontal segment is traversed in the left direction, then there will be a horizontal iod-subpath contradicting that $T(X_{i-1}, Y_j)$ is a type-2 path. So as we traverse $T(X_{i-1}, Y_j)$ from X_{i-1}, horizontal segments are traversed from left to right and the vertical segments are traversed in either direction. Therefore, $Y_j \in P_j^l \cup P_j^t \cup P_j^b \cup \{TL_j, BR_j\}$. However, if $Y_j \in P_j^t \cup P_j^b \cup \{BR_j\}$, then there is a horizontal iod-subpath, contradicting that the path $T(X_{i-1}, Y_j)$ is type-2. It must then be that $Y_j \in P_j^l \cup \{TL_j\}$. Therefore, the path $T(X_{i-1}, Y_j)$ is type-VMR.

Case 2: Path $T(X_{i-1}, Y_j)$ is type-H.

Since the path has at least one horizontal iod-subpath and the path is type-2, it must be that $X_{i-1} = BR_{i-1}$ and as we traverse the path $T(X_{i-1}, Y_j)$ from X_{i-1} to Y_j, all its vertical segments are traversed in the downward direction. The horizontal segments of $T(X_{i-1}, Y_j)$ are traversed in either direction. Therefore, Y_j can only belong to $P_j^t \cup P_j^l \cup P_j^r \cup \{TL_j, BR_j\}$. However if $Y_j \in P_j^l \cup P_j^r \cup \{BR_j\}$, then there is also a vertical iod-subpath, contradicting that $T(X_{i-1}, Y_j)$ is type-2. It must then be that $Y_j \in P_j^t \cup \{TL_j\}$, and the path $T(X_{i-1}, Y_j)$ is type-HMD.

Case 3: Path $T(X_{i-1}, Y_j)$ is type-N and $X_{i-1} \neq Y_j$.

Since the path has no iod-subpaths, it must be that $X_{i-1} = BR_{i-1}$, and as we traverse the path $T(X_{i-1}, Y_j)$ from X_{i-1} to Y_j, all its vertical segments are traversed in the downward direction. It cannot be that a horizontal line segment of $T(X_{i-1}, Y_j)$ is traversed from right to left because that would mean there is a horizontal iod-subpath, contradicting that $T(X_{i-1}, Y_j)$ is a type-N path. Thus, as we traverse the path $T(X_{i-1}, Y_j)$ from X_{i-1} to Y_j, the segments are traversed in the downward and rightward direction. If $Y_j \in P_j^t \cup P_j^l$, a horizontal or vertical iod-subpath is formed, contradicting that $T(X_{i-1}, Y_j)$ is type-N. It must then be that $Y_j = TL_j$. Therefore, the path $T(X_{i-1}, Y_j)$ is type-ND.

Case 4: Path $T(X_{i-1}, Y_j)$ is type-N and $X_{i-1} = Y_j$.

If $X_{i-1} \in P_{i-1}^r$ then $Y_j = TL_j$, and if $X_{i-1} = BR_{i-1}$ then $Y_j \in P_j^l \cup \{TL_j\}$. So the path is type-$ND_1$.

□

Lemma 5.8 *If paths $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ are type-2, then a critical point of n_i can be connected to the corridor by adding line segments of length at most $l_{i-1} + h_i + l_i$.*

Proof: Since paths $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ are type-2 then by Lemma 5.7 they can only be type VMR, HMD, ND, or ND_1. On the path $T(X_{i-1}, Y_{i+1})$, let X be the bifurcation point where the paths $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ break into two, or where $T(X_{i-1}, Y_i)$ ends (i.e., $X = Y_i$) or where $T(X_{i-1}, Y_{i+1})$ ends (i.e., $X = Y_{i+1}$). First we show that the path $T(Y_i, Y_{i+1})$ is type-1. There are four cases depending on the type of path $T(X_{i-1}, X)$.

Case 1: Path $T(X_{i-1}, X)$ is type-VMR. Clearly both $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ are also type-VMR paths. Therefore, the paths $T(X, Y_i)$ and $T(X, Y_{i+1})$ are type-VMR, type-ND, or type-ND_1. There are three cases depending on X: (a) $X = Y_i$, (b) $X = Y_{i+1}$, and (c) $X \neq Y_i$ and $X \neq Y_{i+1}$. For case (a) ($X = Y_i$) when $Y_i = TL_i$ the first two[3] (or three) segments of $T(X, Y_{i+1})$ form a horizontal iod-subpath with P_i^t (see Figure 18(a)-(c)). When $Y_i \in P_i^l$ the first two (or four) segments of $T(X, Y_{i+1})$ form a horizontal iod-subpath with P_i^t (see Figure 18(d)-(e)). Applying similar arguments for case (b) ($X = Y_{i+1}$), we know that $T(X, Y_{i+1})$ has a horizontal iod-subpath. For case (c) (when $X \neq Y_i$ and $X \neq Y_{i+1}$), the first two segments of $T(X, Y_i)$ and the first two segments of $T(X, Y_{i+1})$ form a horizontal iod-subpath in $T(Y_i, Y_{i+1})$ (see Figure 19). In either of the three cases if $T(X, Y_i)$ or $T(X, Y_{i+1})$ is a path type-VMR, then from our previous arguments we know there is a horizontal iod-subpath in $T(Y_i, Y_{i+1})$. Therefore, path $T(Y_i, Y_{i+1})$ is type-1. On the other hand, if both $T(X, Y_i)$ and $T(X, Y_{i+1})$ are type-ND, then the first two segments in the path $T(X, Y_i)$ and the first two segments in the path $T(X, Y_{i+1})$ form a horizontal and vertical iod-subpaths (see Figure 20). Therefore path $T(Y_i, Y_{i+1})$ is type-1.

Case 2: Path $T(X_{i-1}, X)$ is type-HMD. Then both $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ are type-HMD. Therefore, the paths $T(X, Y_i)$ and $T(X, Y_{i+1})$ are type-HMD, type-ND, or type-ND_1. There are three cases depending on X: (a) $X = Y_i$, (b) $X = Y_{i+1}$, and (c) $X \neq Y_i$ and $X \neq Y_{i+1}$. For case (a) ($X = Y_i$) when $Y_i = TL_i$ the first two or three segments of $T(X, Y_{i+1})$ form a vertical iod path with P_i^l (see Figure 21(a)-(c)). When $Y_i \in P_i^t$, the first three or four segments of $T(X, Y_{i+1})$ form also a vertical iod-subpath with P_i^l

[3]In the proof of this lemma we state that there are several segments in a path. We do not give a proof of this because it is straightforward when we use the edge of the ncpe rectangle at the end of the path.

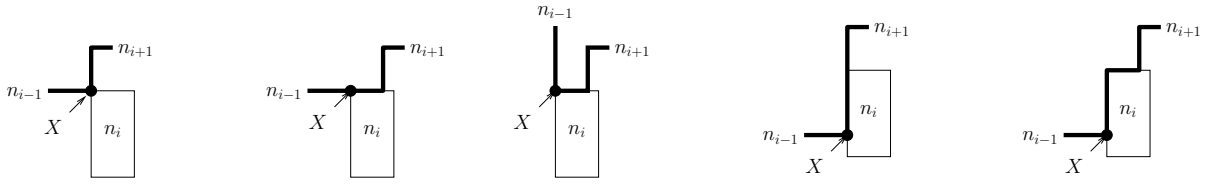


Figure 18: Connection of ncpe rectangle n_i along the path $T(X_{i-1}, Y_{i+1})$ of type-VMR: (a)-(c) $X = Y_i = TL_i$, (d)-(e) $X = Y_i \in P_i^l$.

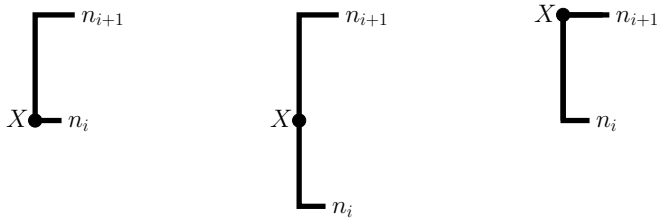


Figure 19: Bifurcation point X of paths $T(X_{i-1}, Y_{i+1})$ and $T(X_{i-1}, Y_i)$ along the path $T(X_{i-1}, Y_{i+1})$ of type-VMR.

(see Figure 21(d)-(e)). Applying similar arguments for case (b) ($X = Y_{i+1}$), we know that $T(X, Y_{i+1})$ has a horizontal iod-subpath. For case (c) (when $X \neq Y_i$ and $X \neq Y_{i+1}$), the first two segments of $T(X, Y_i)$ and the first two segments of $T(X, Y_{i+1})$ form a vertical iod-subpath in $T(Y_i, Y_{i+1})$ (see Figure 22). In either of the three cases if $T(X, Y_i)$ or $T(X, Y_{i+1})$ is a path type-HMD, then from our previous arguments we know there is a vertical iod-subpath in $T(Y_i, Y_{i+1})$. Therefore, $T(Y_i, Y_{i+1})$ is type-1. On the other hand, if both $T(X, Y_i)$ and $T(X, Y_{i+1})$ are type-ND, then the first two segments in path $T(X, Y_i)$ and the first two segments in path $T(X, Y_{i+1})$ form a vertical and horizontal iod-subpaths (see Figure 20). Thus $T(Y_i, Y_{i+1})$ has a vertical and horizontal iod-subpaths and therefore it is type-1.

Case 3: Path $T(X_{i-1}, X)$ is type-ND. Then the paths $T(X_{i-1}, Y_i)$ and $T(X_{i-1}, Y_{i+1})$ are types VMR, HMD,

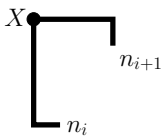


Figure 20: Bifurcation point X of paths $T(X_{i-1}, Y_{i+1})$ and $T(X_{i-1}, Y_i)$, both of type-ND.

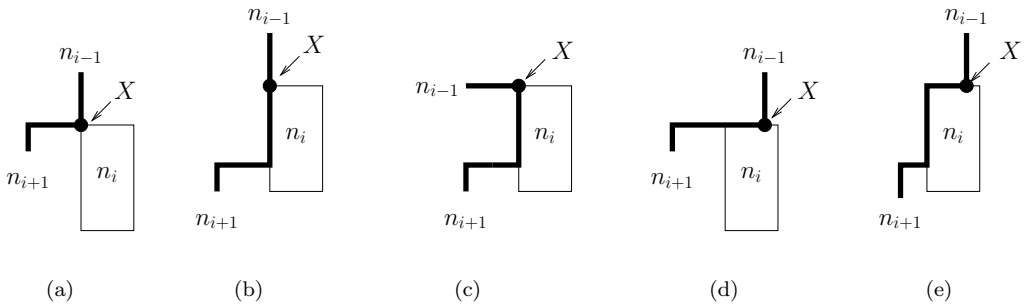


Figure 21: Connection of ncpe rectangle n_i along the path $T(X_{i-1}, Y_{i+1})$ of type-HMD: (a)-(c) $X = Y_i = TL_i$, (d)-(e) $X = Y_i \in P_i^t$.

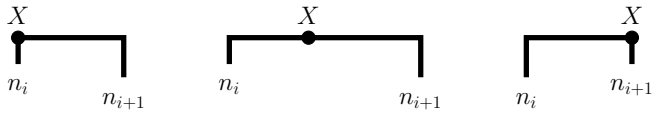


Figure 22: Bifurcation point X of paths $T(X_{i-1}, Y_{i+1})$ and $T(X_{i-1}, Y_i)$ along the path $T(X_{i-1}, Y_{i+1})$ of type-HMD.

or ND. Therefore, paths $T(X, Y_i)$ and $T(X, Y_{i+1})$ are type VMR, HMD, ND, or ND_1. If one of these subpaths is type-VMR and the other is type-HMD, then clearly, $T(Y_i, Y_{i+1})$ has a vertical and horizontal iod-subpaths and it is type-1. So either both of the paths $T(X, Y_i)$ and $T(X, Y_{i+1})$ are either VMR, ND, or one is ND_1, or they are HMD, ND, or one is ND_1. In either case, arguments similar to the ones for Cases 1 and 2 can be used to establish that the path $T(Y_i, Y_{i+1})$ is type-1.

Case 4: Path $T(X_{i-1}, X)$ is type-ND_1. Since n_i is visited after n_{i-1} but before n_{i+1}, one can show that $X_{i-1} = X = Y_i$. Since these two ncpe rectangles share a point, it must be that at least one of the rectangles has a corner point at point X. But this corner point cannot be the TR or BL corner as otherwise n_{i-1} or n_i would not be an ncpe rectangle. So it has to be that one corner point is a TL corner or it is a BR corner. Since $X_{i-1} \in P_{i-1}^r \cup \{BR_{i-1}\}$, then X_i belongs to P_i^t or corresponds to the TL_i corner. In either case, it must be that the path $T(X, Y_{i+1})$ is type-VMR, containing a vertical iod-subpath. But a horizontal iod-subpath is formed by the top-edge of n_i. Therefore path $T(Y_i, Y_{i+1})$ is type-1.

In all the cases, there exist both horizontal and vertical iod-subpaths along the path $T(Y_i, Y_{i+1})$. Now, by Lemmas 5.2 and 5.3 we know that there exist both horizontally and vertically contained rectangles along $T(Y_i, Y_{i+1})$. Therefore, using arguments similar to those of Lemma 5.5 and the fact that $T(Y_i, Y_{i+1})$ is type-1,

we know that a critical point of n_i can be connected to the corridor by adding line segments of length at most $l_{i-1} + h_i + l_i$.

□

5.3 Selecting the four corners and one special point

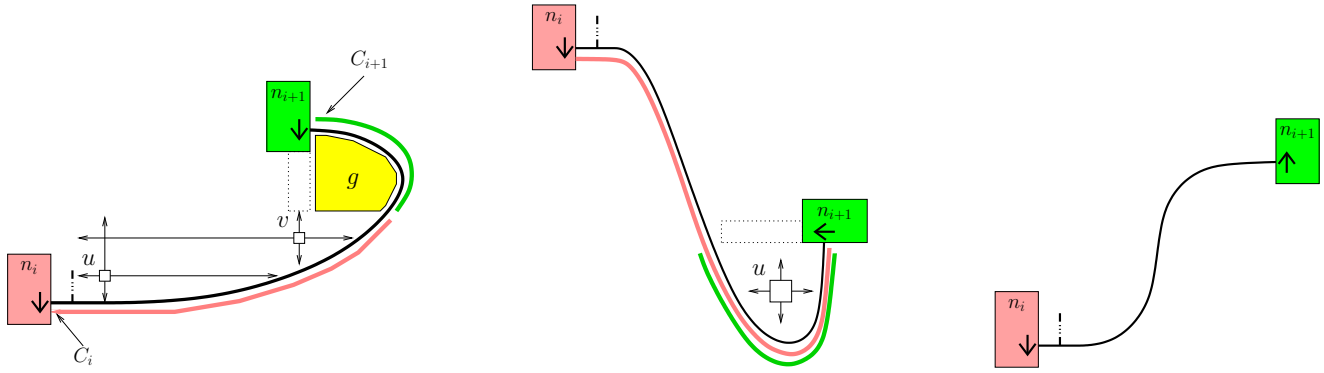
The critical points in $S(4C+)$ for each rectangle $R_i \in R$ are its four corners and a special point. For this case $k_{S(4C+)} = 5$ and we can show that $r_{S(4C+)} = 3$. Therefore, the approximation ratio of the parametrized algorithm is 30 as in the case of $S(2OC+)$.

We now briefly discuss our proof strategy to show that given any corridor $T(I)$ there is a corridor $T(I_{S(4C+)})$ such that $t(I_{S(4C+)}) \leq 3 \cdot t(I)$. As in the case of $S(2OC+)$, given any corridor $T(I)$ we identify all the ncpe rectangles ad establish an ordering $(n_1, n_2, \ldots, n_q)$ between them. Assume there exists at least one ncpe rectangle ($q > 1$), otherwise $t(I_{S(4C+)}) = t(I)$ and the result follows. For each ncpe we find a shortest path from one of its critical points to the corridor $T(I)$. We then select this path to connect a critical point to the corridor $T(I)$. Clearly after deleting some edges for removing any cycle that may have been created, corridor $T(I)$ plus a subset of these connections give a corridor $T(I_{S(4C+)})$. Next we need to show that the sum of the length of the segments introduced is at most $2 \cdot t(I)$. This is the part that is more complex than the one for the selector function $S(2OC+)$.

We characterize the region between every pair of adjacent ncpe's. The region between two adjacent ncpe's is said to be of type 0, 1 or 2 (see Figure 23). The region between ncpe rectangles n_i and n_{i+1} is type-2 (see Figure 23 (a)), if the distance along the corridor between n_i and n_{i+1}, which we call in Section 5.1 l_i, is larger than the edge-length needed to connect both a critical point from n_i and one from n_{i+1} to the corridor. This is the most desirable case. If this were to be the case for every pair of adjacent ncpe's then the proof of the approximation ratio would be simple to establish. In fact we would even be able to establish a better ratio. However, this is not always the case. The region between ncpe n_i and n_{i+1} is type-1 (see Figure 23 (b)), if l_i is larger than the edge-length needed to connect either a critical point from n_i to the corridor, or one from n_{i+1} to the corridor, but not both. If this were the case for every pair of adjacent ncpe's the proof would also be simple. The main problem is when the region between ncpe n_i and n_{i+1} is type-0 (see Figure 23 (c)). In this case, the edge-length needed to connect a critical point of either n_i or n_{i+1} cannot be bounded above by

l_i. This is when the proof is complex because we need to consider a sequence of ncpe rectangles, not just three as in the previous subsection.

Now suppose that there is a sequence of type-0 adjacent ncpe rectangles as shown in Figure 24. The connection of the first ncpe rectangle n_1 to the corridor has already been accounted for in l_0. Now we need to charge the connection of a critical point of each ncpe rectangles $n_2, \ldots, n_9$ to the corridor. The connection for the ncpe n_2 is charged to the horizontal distance from ncpe n_1 to ncpe n_2, and the vertical distance from ncpe n_2 to ncpe n_3 because one can show that the area includes at least one rectangle already connected to the corridor. Similarly, the cost of connecting ncpe n_3 can be charged to the horizontal distance from ncpe n_2 to ncpe n_3 and the vertical distance from ncpe n_3 to ncpe n_4. And so forth until ncpe n_9, where its connection is charged to the corridor after it because there is a rectangle inside the box in the center of Figure 24.



(a) 2 (b) 1 (c) 0

Figure 23: Region Types.

Other complex cases are given in Figure 25(a-b) which indicate how to deal with the sequence of adjacent rectangles of types 001000 and sequence 00111. By a sequence $Y_1 Y_2 \ldots Y_k$ we mean that the first pair of ncpe's is type Y_1 and the second pair is type Y_2, and so on. There are more critical cases that need to be considered. It is possible to characterize all sequences that need to be solved, but it is quite complex. That is why we only present the analysis for the selector function $S(2OC+)$, which is significantly simpler.

We claim without stating any details that the approximation ratio of the parameterized algorithm Alg$(S(4C+))$ is 30.

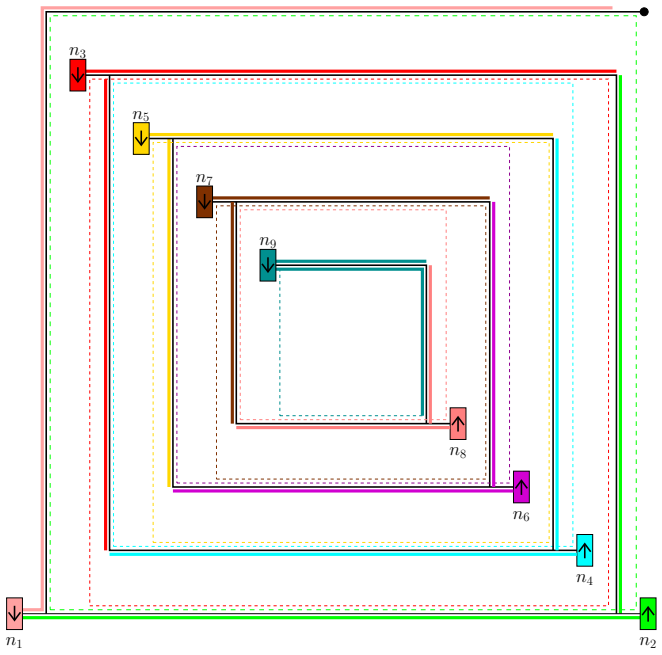
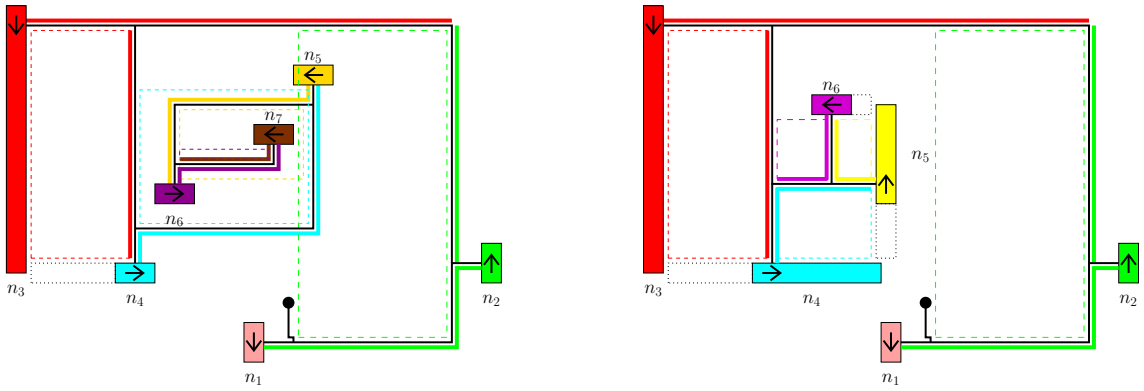


Figure 24: Sequence of type-0 adjacent ncpe rectangles.

6 Additional Results and Discussion

The analysis of our approximation algorithm also applies (with the same time complexity and approximation ratio) to the version of the MLC-R problem when not all the rectangles in R need to access the corridor. This corresponds to the "Steiner" version of the problem, rather than the "spanning tree" version of the problem which we call the MLC-R problem. The analysis of our approximation algorithm also applies (with the same time complexity and approximation ratio) when the boundary of the MLC-R problem is a rectilinear polygon rather than the rectangle F, or when the problem is to find a tree that is not necessary joined to the boundary of F.

Our approximation algorithm can also be applied to the MLC$_n$ problem, but the approximation ratio depends on n. When n is bounded by a constant, the algorithm is a constant ratio approximation algorithm. The selector function $S(C+)$, which includes all the corner points plus other points, and a special point of each rectilinear polygon $R_i \in R$, is more complex. The idea is to introduce for each rectilinear polygon R_i the least number of horizontal line segments (all of which are completely inside R_i) that partition the interior of R_i into rectangles (see Figure 26). All the corners of these rectangles are on the boundary of R_i and are the fixed points for R_i. Then we add a special point for R_i. The total number of critical points for each R_i is at



(a) Sequence 001000 (b) Sequence 00111

Figure 25: Assignment of regions between adjacent ncpe rectangles.

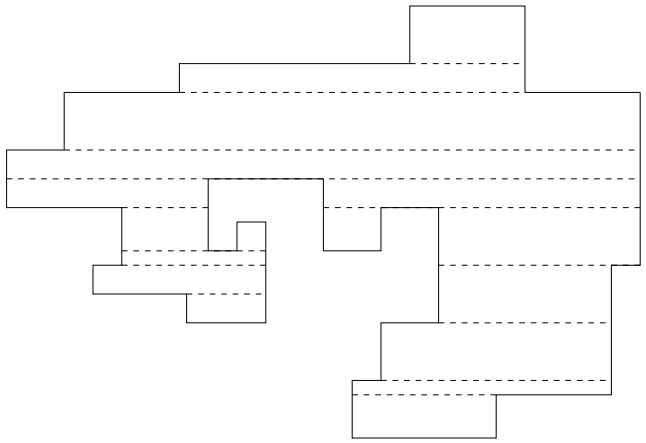


Figure 26: Partition of R_i into rectangles by introducing horizontal line segments completely inside R_i.

most $\frac{3}{2}n - 1$. Therefore, $k_{S(C+)} = \frac{3}{2}n - 1$.

We now need to determine $r_{S(C+)}$. The process follows the same lines as the one for the MLC-R problem. The only difference is when considering the ncpe's. Instead of selecting for each ncpe n_i the rectilinear polygon, we just take the rectangle that the tour intersects first. So the set of ncpe's are simply rectangles with four fixed points. The special point is for the whole rectilinear polygon. Our analysis for the MLC-R problem also applies for this case. Though, it can be simplified since the rectangles include at least four critical points. Note that the special point may or may not be part of the rectangles. But since it is associated with the rectilinear polygon, that is enough for our analysis. The approximation bound is therefore $2k_{S(C+)} \cdot r_{S(C+)} = 2 \cdot (\frac{3}{2}n - 1) \cdot 5 = 15n - 10$. For n bounded above by a constant, the approximation ratio is a constant. Note that with a more careful

introduction of critical points we can decrease the approximation ratio. For example, we just need the bottom-left and top-right corner of each rectangle. But at this point we are only interested in showing that our approximation algorithm takes polynomial time and it is a constant ratio approximation algorithm for the MLC_n problem.

We have applied our algorithm (with the same constant ratio) to restricted versions of the MLC problem, but so far we have not been able to apply it to all cases. The existence of a constant ratio approximation algorithm for the MLC problem remains a challenging open problem. An equally challenging problem is to develop approximation algorithm for the MLC-R with a significant smaller approximation ratio, for example two.

When we restrict the MLC-R problem to $S(2OC+)$ or $S(4C+)$, we use Slavik's algorithm to generate a suboptimal solution. This is the most time consuming part of our procedures. The first question is develop a faster approximation algorithm for this problem. The second question is to develop an algorithm with a smaller approximation ratio or even one that generates an optimal solution. But since the latter problem is NP-hard for the $S(2OC+)$ or $S(4C+)$ restrictions, it is unlikely one can find an efficient algorithm for its solution. The NP-hardness proof follows the same lines as the one in [12], but we need to do some modifications to show that the the MLC-R restricted to $S(2OC+)$ ($S(4C+)$) (i.e. every rectangle must intersect the corridor at a critical point defined by $S(2OC+)$ (resp. $S(4C+)$)) is NP-complete.

The MLC problem (and all its subproblems) can be easily solved in polynomial time when the objective function is to find a tree such that the maximum length of the path from the root to a leaf is least possible. A class of interesting open problems are ones with dual-criteria objective functions.

Another interesting problem is the group-TSP problem when restricted to rectangles as in the case of the of the MLC-R problem, which we call the *rectangular group-TSP*. In this version of the TSP one may visit the same edge or vertex more than once. The PTAS for the geographically clustered version of the MLC problem also applies to the rectangular group-TSP [5].

We claim that the same approach that we use for the MLC-R problem can also be applied to the rectangular group-TSP. It is simple to show that the selector function S that do not generate constant ratio approximations for the MLC-R problem do not generate constant ratio approximations for the rectangular group-TSP. However the selector function $S(2OC+)$ also generates a constant ratio approximation to the group-TSP. In fact we can

just use the tour of the corridor (traversing each edge twice) as the solution to the rectangular TSP problem. The approximation ratio in this case will be 60 time the length of an optimal tour. However there is a better algorithm for this case. instead of using Slavik approximation algorithm for the Tree Errand problem, we use the one for the group TSP. In this case the approximation ratio is $\frac{3}{2}\rho$ [20, 21]. By applying the same approach as in the previous section we know that the approximation ratio is $\frac{3}{2} \cdot k_{S(2OC+)} \cdot r_{S(2OC+)}$, where $k_{S(2OC+)} = 3$ and $r_{S(2OC+)} = 5$. This results in the approximation ratio 22.5.

Approximation algorithms for other versions of the group-TSP problem are discussed in [5]. The same type of approach used for the MLC_n can be used for the rectilinear n-gon group-TSP, resulting in a polynomial time constant ratio approximation. For brevity we do not discuss additional results.

References

[1] Arkin, E. M., Halldorsson, M. M., and Hassin, R. Approximating the tree and tour covers of a graph. *Information Processing Letters 47*, 6 (1993), 275–282.

[2] Bafna, V., Berman, P., and Fujito, T. Constant ratio approximations of feedback vertex sets in weighted undirected graphs. *Proceedings of the 6th. International Symposium on Algorithms and Computation* (1995), 142–151.

[3] Bateman, C. D., Helvig, C. S., Robins, G., and Zelikovsky, A. Provably good routing tree construction with multi-port terminals. In *ISPD '97: Proceedings of the 1997 International Symposium on Physical Design* (1997), ACM Press, pp. 96–102.

[4] Becker, A., and Geiger, D. Approximation algorithms for the loop cutset problem. *Proceedings of the 10th. Conference on Uncertainty in Artificial Intelligence* (1994), 60–68.

[5] Bodlaender, H. L., Feremans, C., Grigoriev, A., Penninkx, E., Sitters, R., and Wolle, T. On the minimum corridor connection and other generalized geometric problems. In *4th. Workshop on Approximation and Online Algorithms (WAOA)* (Zurich, Switzerland, September 2006), T. Erlebach and C. Kaklamanis, Eds., vol. 4368 of *Lecture Notes in Computer Science*, Springer, pp. 69–82.

[6] Chudak, F. A., Goemans, M. X., and Hochbaum, D. A primal-dual interpretation of two 2-approximation algorithms for the feedback vertex set problem in undirected graphs. *Operations Research Letters 22* (1998), 111–118.

[7] de Berg, M., Gudmundsson, J., Katz, M. J., Levcopoulos, C., Overmars, M. H., and van der Stappen, A. F. TSP with neighborhoods of varying size. *J. Algorithms 57*, 1 (2005), 22–36.

[8] Demaine, E. D., and O'Rourke, J. Open problems from CCCG 2000. In *Proceedings of the 13th Canadian Conference on Computational Geometry (CCCG 2001)* (2001), pp. 185–187.

[9] Dumitrescu, A., and Mitchell, J. S. B. Approximation algorithms for TSP with neighborhoods in the plane. In *SODA '01: Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms* (Philadelphia, PA, USA, 2001), Society for Industrial and Applied Mathematics, pp. 38–46.

[10] Elbassioni, K., Fishkin, A., Mustafa, N. H., and Sitters, R. Approximation algorithms for euclidean group TSP. In *Automata, languages and programming : 32nd International Colloquim, ICALP 2005* (Lisbon, Portugal, 2005), L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, Eds., vol. 3580 of *Lecture Notes in Computer Science*, Springer, pp. 1115–1126.

[11] Eppstein, D. Some open problems in graph theory and computational geometry. PDF file (3.89 MB), World Wide Web, http://www.ics.uci.edu/∼eppstein/200-f01.pdf, November, 2001.

[12] Gonzalez-Gutierrez, A., and Gonzalez, T. F. Complexity of the minimum-length corridor problem. *Comput. Geom. Theory Appl. 37*, 2 (2007), 72–103.

[13] Helvig, C. S., Robins, G., and Zelikovsky, A. An improved approximation scheme for the group Steiner problem. *Networks 37*, 1 (2001), 8–20.

[14] Ihler, E. Bounds on the quality of approximate solutions to the group Steiner problem. In *WG '90: Proceedings of the 16th International Workshop on Graph-theoretic Concepts in Computer Science* (1991), Springer-Verlag New York, Inc., pp. 109–118.

[15] Jin, L. Y., and Chong, O. W. The minimum touching tree problem. PDF file (189 KB), World Wide Web, http://www.yewjin.com/research/MinimumTouchingTrees.pdf, National University of Singapore, School of Computing, 2003.

[16] Karp, R. M. Reducibility among combinatorial problems. *Complexity of Computer Computations* (1972), 85–103.

[17] Mitchell, J. S. B. *Geometric shortest paths and network optimization,* in Handbook of Computational Geometry. Elsevier, North-Holland, Amsterdam, 2000, pp. 633–701.

[18] Reich, G., and Widmayer, P. Beyond Steiner's problem: a VLSI oriented generalization. In *WG '89: Proceedings of the Fifteenth International Workshop on Graph-theoretic Concepts in Computer Science* (1990), Springer-Verlag New York, Inc., pp. 196–210.

[19] Safra, S., and Schwartz, O. On the complexity of approximating TSP with neighborhoods and related problems. *Proc. of the 11th. Annual European Symposium on Algorithms* (2003), 446–458.

[20] Slavik, P. The errand scheduling problem. Tech. Rep. 97-02, Department of Computer Science and Engineering, University of New York at Buffalo, http://www.cse.buffalo.edu/tech-reports/, 1997.

[21] Slavik, P. Approximation Algorithms for Set Cover and Related Problems. Ph. D. Thesis 98-06, Department of Computer Science and Engineering, University of New York at Buffalo, http://www.cse.buffalo.edu/tech-reports/, 1998.