Algorithms for Boundary Labeling of Horizontal Line Segments

A Project Report Submitted

in Partial Fulfilment of the Requirements

for the Degree of

MASTER OF TECHNOLOGY

in

Computer Sciene

by

Abhilash Kurmi

(Roll No. CS1621)

Under the Guidance of

Dr. Sasanka Roy

Associate Professor

Advance Computing and Microelectronics Unit

to the

INDIAN STATISTICAL INSTITUTE

KOLKATA - 700108, INDIA

June 2019

Declaration

I here by declare that dissertation report entitled **"Algorithms for Boundary Labelling of Horizontal Line Segments"** submitted to Indian Statistical Institute, Kolkata, is a bonafide record of work carried out in partial fulfilment for the award of the degree of **Master of Technology in Computer Science**. The work has been carried out under the guidance of **Dr. Sasanka Roy**, Associate Professor, ACMU, Indian Statistical Institute, Kolkata.

Kolkata - 700 108
June 2019

Abhilash Kurmi
Roll No: CS1621
Indian Statistical Institute
Kolkata - 700108 , India

CERTIFICATE

This is to certify that the dissertation entitled **"Algorithms for Boundary Labeling of Line Segments"** submitted by **Abhilash Kurmi (Roll No: CS1621)** to Indian Statistical Institute, Kolkata, in partial fulfilment for the award of the degree of **Master of technology in Computer Science** is a bonafide record of work carried out by him under my supervision and guidance. The dissertation has fulfilled all the requirements as per the regulations of this institute and, in my opinion, has reached the standard needed for submission.

Sasanka Roy

Kolkata - 700 108
June 2019

Dr. Sasanka Roy
Associate Professor
Indian Statistical Institute
Kolkata - 700108 - , India

ABSTRACT

In *boundary labelling problem* the target is to labeling a set P of n points in the plane with labels that are aligned to side of the bounding box of P . In this work, we investigate a variant of this problem. In our problem, we consider a set of sites inside a rectangle $\mathbb{R}$ and label are placed in the compliment of $\mathbb{R}$ and touches the left boundary of it. Labels are axis- parallel rectangles of same size and no two labels overlaps. We introduce a set V, called *visibility* , which is a set of subsets of labels correspond to points of sites. Before connecting site (say p) at point (say $p_1 \in p$) with some label (say l), first we need to check weather subset of label correspond to p_1 is in set V or not. If it is then we check the label l belongs to that subset of label or not. If it contains that label then we can join site to the label, otherwise not. In our problem we used *po-leaders*, that is starting from site it is parallel to the side of $\mathbb{R}$ where its label resides and then orthogonal to that side of $\mathbb{R}$. We considered various geometric objects as sites, such as point, same length horizontal segment, different length horizontal segments. As a solution, we derive a dynamic algorithm that minimizes the arbitrary cost function and give us planar solution where sites connects to labels by po- leaders and induces a matching such that no two po-leader intersects, also no two leaders shares common site (or label) and every leader satisfies visibility V. For points as sites, our dynamic algorithm runs in $O(n^3)$ time and optimizes the cost function. This running time also same for the case of unit length horizontal line segments as sites. Then we taken arbitrary length horizontal segment, algorithms runs in $O(n^4)$ time. We assumed that only one end point of any horizontal line segment can be used to connect label (by po-leader).

Acknowledgements

I would like to express my thankfulness to my supervisor *Dr. Sasanka Roy*, Associate professor, ACMU, Indian Statistical Institute, Kolkata for his guidance and continuous support and encouragement.

I would also extend my deepest respect towards *Prof. Subhas Chandra Nandy* for reviewing my mid- term presentation and providing valuable inputs.

My deepest thanks to all the teachers of Indian Statistical Institute for their valuable suggestions.

My sincere thanks to *Satyabrata Jana* for his valuable support, suggestion and discussions.

Finally, I am very thankful to my parents and family for their everlasting support.

Last but not the least, I would like to thank all of my friends for their help and support. I thank all those, whom I have missed out from the above list.

Contents

Chapter 1

Introduction

A Map consists of areas having different sizes and various information. To recognize those areas, their associated information is required. For this, we label areas with their associated information. But when we need better visualization of map as well as information about areas, then arbitrary labelling of map leads to loss of information. Also, arbitrary labelling reduces the readability of map if map is dense (that is, map has too many areas and lot of information to show). To address this problem, we need algorithms to labelling of maps such that the results increases the visibility as well as readability of map with minimum loss of information.

1.1 Boundary labeling problem

In 2007, Bekos et al.[1] introduce the following *boundary labeling* problem.

Given an axis-parallel rectangle $R = [l_R, r_R] \times [b_R, t_R]$ and a set $P \subset R$ of n sites $p_i = (x_i, y_i)$, each associated with an axis-parallel rectangular open label l_i

of width w_i and height h_i , the task is to find an *optimal leader-label placement*. An optimal leader-label placement follows the following criterias:

1. Labels have to be disjoint.
2. Labels have to lie outside R.
3. Intersections of leaders with other leaders, sites or labels are not allowed.
4. Leader c_i connects site p_i with label l_i for $1 \leq i \leq n$.
5. The ports which are the endpoints of leaders at labels may be fixed.

A rectilinear leader consists of a sequence of axis-parallel segments that connects a site with its label. Bekos et al.[1] consider boundary labeling problem for various types of leaders and optimal leader-label placements according to the following two objective functions:

(I) short leaders (minimum total length) and

(II) simple leader layout (minimum number of bends while considering rectilinear leader).

1.2 Our problem definition

We describe our problem definition with the notation used in kindermann et al.[3].

We are given an axis parallel rectangle $\mathbb{R} = [0, W] \times [0, H]$, which is called the *enclosing rectangle*, a set $P \subset \mathbb{R}$ of n sites $s_1, s_2, ..., s_n$ within the rectangle $\mathbb{R}$ and a set $\mathbb{L}$ of n axis-parallel rectangles $l_1, l_2, ..., l_n$ of equal size (same width and same height), called *labels*. The labels lie in the complement of $\mathbb{R}$ and touch the left boundary of $\mathbb{R}$. No two labels overlap.

We introduce a set V, called *visibility* , which is a set of subsets of labels correspond to points of sites. Before connecting site (say s_i) at point (say $s_{i_a} \in s_i$)

with some label (say l), first we need to check weather subset of label correspond to s_{i_a} is in set V or not. If it is then we check the label l belongs to that subset of label or not. If it contains that label then we can join site to the label, otherwise not.

We define *visibility* $V = \{V_{s_{1_1}}, V_{s_{1_2}}, .., V_{s_{2_1}}, V_{s_{2_2}}, \ldots\ldots, V_{s_{n_1}}, V_{s_{n_2}}, \ldots\}$ where $V_{s_{i_a}} \subseteq \mathbb{L}$ and $s_{i_a} \in s_i$ and s_{i_a} is a point. If we wish to join some point (say $s_{j_b} \in s_j$ to label (say l_k) where s_j is a site, we check weather subset of labels (say $V_{s_{j_b}}$) corresponding to s_{j_b} is in V or not. If $V_{s_{j_b}} \in V$ and $l_k \in V_{s_{j_b}}$ then we have a choice of leader (say c) which connects site s_j at s_{j_b} to label l_k, otherwise not.[1]

We denote an instance of the problem by the quadruplets $(\mathbb{R}, \mathbb{L}, P, V)$. A *feasible solution* of a problem instance $(\mathbb{R}, \mathbb{L}, P, V)$ is a set of n non- intersecting *leaders* $C = \{c_1, c_2, ..., c_n\}$ in the interior of $\mathbb{R}$, if they satisfies the following conditions:

(1): each leader connects a site to label and no two leaders of set C share a common label (or site).

(2): for each leader except their end points they should not intersect with any site.

(3): leaders must satisfy visibility V.

In our problem, we consider only *po-leader*. A po-leader consists of a two axis-parallel segments that connects a site with its label. The first line segment of a po-leader starting at a site p_i is parallel (p) to the side of $\mathbb{R}$ where its label touches and the second line segment which joins label is orthogonal (o) to that side of $\mathbb{R}$. For an illustration of po-leader see 1.1. The endpoint of a leader at a label is said to be *port*. Two leaders are same if and only if their corresponding

[1] In theory $|V|$ possibly infinite, But when we use it for our algorithms $|V| < \infty$.

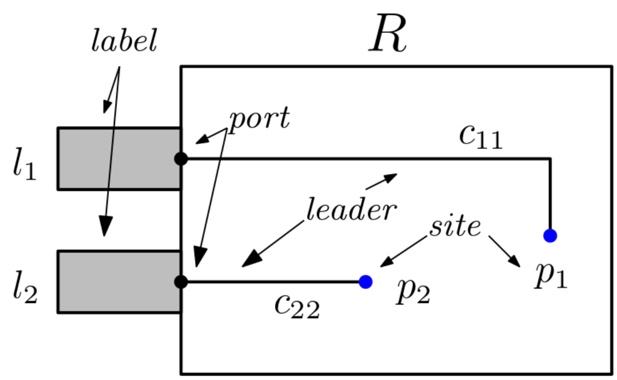


Figure 1.1: *Example of po-leaders*

end points are same. We define this type of labelling as *planar boundary labelling by po-leader.* We define cost function which takes input as leader and returns a real value. The optimal solution of a problem is a planar boundary labelling by po-leader with minimum cost.

We consider some variation of this problem which are as follows:

(Prob 1): P is a set of n points, i.e. we take points as sites.

(Prob 2): P is a set of n same length horizontal (or unit length) line segments where only end points of line segment can be used to connect label.

(Prob 3): P is a set of n arbitrary length horizontal line segments where only end points of line segment can be used to connect label.

We say a site p is *labelled* with a label l if there exist a po-leader connecting p with label l. Similarly, a horizontal line segment h is *labelled* with a label l if there exist a po-leader connecting one of the end point of h with label l. We call an instance $(\mathbb{R}, \mathbb{L}, P, V)$ is *solvable*, if there exist a feasible solution for it.

Chapter 2

Related Work

In this chapter, we will discuss some algorithms and related work on some variation of boundary labelling with po-leader problem.

2.1 One-side labeling with uniform labels

In [1], Bekos et al. proved the following theorem.

Theorem 2.1.1. *Given a rectangle* $\mathbb{R}$*, a side* s *of* $\mathbb{R}$*, a set* $P \subset \mathbb{R}$ *of* n *sites in general position and a rectangular uniform label for each site, there is an* $O(n^2)$*-time algorithm that produces a legal type-po labeling of minimum total leader length.*

Now we describe the alorithm given by Bekos et al. [1] that produces a legal type-po labeling of minimum total leader length.

2.1.1 Algorithm

- **Input** : $(\mathbb{R}, \mathbb{L}, P)$, where $\mathbb{R}$ is a rectangle, P is a set of n sites inside $\mathbb{R}$ and $\mathbb{L}$ is a set of n labels to the right side r_2 of $\mathbb{R}$
- **Output** : Planar leader label placement with minimum length po-leaders C.

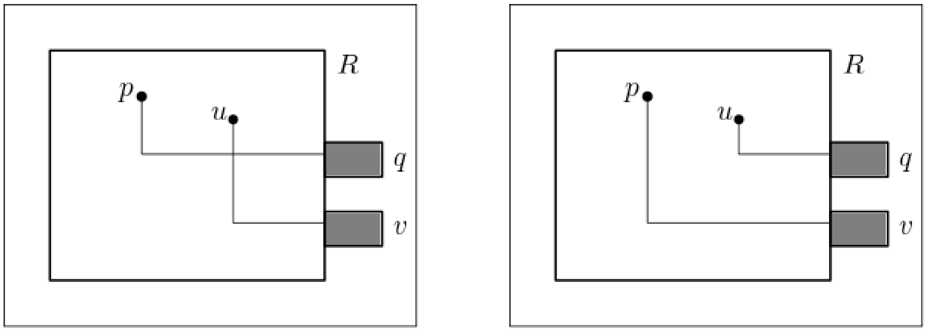
2.1.2 Proof of Correctness

In Algorithm (1), we have two while loops. Since, we can't decrease horizontal length of *po-leader*. So, we have only choice to decrease vertical length. We sort the labels and sites in increasing order (with respect to Y-axis). By connecting the i^{th} smallest site p_i with i^{th} smallest label l_i using leader c_{ii} gives us guarantee that, we have minimum total leader length. But, the solution may contain the intersection between leaders.

To remove intersection between leaders, we use another while loop in algorithm. Suppose, we have leader c_{pq} and set of leaders C_l intersects horizontal part of it, where leader $c_{wx} \in C_l$ if and only if (Y-coordinate of site p) $\geq$ (Y-coordinate of w). Let's assume that $c_{uv} \in C_l$ be the right most leader intersect at horizontal part of the leader c_{pq}. If we connect site u to label q, resulting leader is c_{uq} (Similarly we obtain leader c_{pv}), then without increasing the length between leaders, we have leader c_{uq} not intersect to any other leader as shown in figure 2.2. Since, there are finite number of leaders and each time we get a leader which doesn't intersect to any other leader. Eventually, we have set of leaders (where no two leaders intersects), without increasing the minimum length.

Algorithm 1: Quadratic Algorithm for Leader Length minimization

Sort the sites P and labels $\mathbb{L}$ with respect to increasing order of their Y-coordinate, store the sorted sites in array A, and sorted labels in array B;
We use 2-D range search tree *T* to store the sites, first level of *T* arranged with respect to Y-axis while auxiliary are arranged with respect to X-axis;
An array $C[n][n+1]$ is used to store the leaders and it's corresponding intersected leaders;
for $(i = 1; i \leq n; i++)$ **do**
    $a = A[i]$;
    $b = B[i]$;
    Connect site a with label b using leader c_{ab} and store at $C[i][1]$. Each of $A[i]$, $B[i]$ and $C[i][1]$ contains two pointers, pointing towards other two. Site a is at coordinate (a_x, a_y) and *port* of label b is at coordinate (b_x, b_y);
    Perform Range Query $[-\infty, a_y] \times [a_x, \infty]$ on *T* and store resulting points (or sites) from $C[i][2]$. Store position of last point (or site) at $C[i][1]$. (Here we assume that query reports the sites in increasing order with respect to X-axis);
end
for $(i = 1; i \leq n; i++)$ **do**
    $l =$ position stored of last point at $C[i][1]$;
    $c_{pq} =$ leader at $C[i][1]$;
    while $l \neq 1$ **do**
        c_{uv} is leader correspond to $C[i][l]$, stored at $C[k][1]$;
        if $c_{pq} \cap c_{uv} \neq \varnothing$ **then**
            Change leader c_{uv} to c_{uq} and c_{pq} to c_{pv};
            Store c_{uq} at $C[k][1]$ and c_{pv} at $C[i][1]$;
            Update related pointers;
            $c_{pq} = c_{pv}$;
        end
        $l = l - 1$;
    end
end



(a) Leaders before swap (b) Leaders after swap

Figure 2.1: After swapping combined leader length is same

2.1.3 Time and Space Complexity

In algorithm (1), sorting takes $O(n \log n)$ time. Construction of Range search tree T take $O(n \log^2 n)$ (can be reduces to $O(n \log n)$). First loop runs up to n iteration, each iteration connects site to label with leader, uses $O(1)$ time, and perform range query, takes $O(\log^2 n + k)$ time (using *fractional cascading* range query time reduces to $O(\log n + k)$), where k is number of points reported by the query. So, first loop runs in $O(n \log^2 n)$ time (can be reduces to $O(n \log n)$.

While, second loop also takes n iteration. In that, each iteration takes $O(n)$ time, because number of intersection, for leader c_{ii} correspond to i^{th} smallest site p_i (with respect to Y-axis), can have at most $(i - 1)$ to other leaders, (where sites correspond to these leaders have Y-coordinate less than (or equal to), Y-coordinate of site p_i). In each iteration, we get a leader which not intersects to any other leader. So, to remove all intersection for the leader c_{ii}, we need at most $(i - 1)$ iteration. This, leads total time complexity for second loop to $O(n^2)$.

By adding we get $O(n^2)$ time complexity for the algorithm. Space complexity is also $O(n^2)$, this we can easily conclude from algorithm.

2.2 Two-side labeling with uniform labels of maximum height

The next result deals with two-side placement of uniform labels of maximum height. The author [1] consider type-po leaders and their goal is to minimize the total leader length. They obtain the following theorem:

Theorem 2.2.1. *Given a rectangle* $\mathbb{R}$ *with* $n/2$ *uniform labels of maximum height on each of its left and right sides, and a set* $P \subset \mathbb{R}$ *of* n *sites in general position, there is an* $O(n^2)$*-time algorithm that attaches each site to a label with non-intersecting type-po leaders such that the total leader length is minimized.*

2.3 Multi-Sided Boundary Labeling problem

Kindermann et al. [3] study the Multi-Sided Boundary Labeling problem, where labels lying on at least two sides of the enclosing rectangle. They consider the problem of finding efficient algorithms for testing the existence of a crossing-free leader layout that labels all sites and also for maximizing the number of labeled sites in a crossing-free leader layout. They also give an algorithm for minimizing the total leader length for two-sided boundary labeling with adjacent sides. More importantly, they restrict their solutions to po-leaders. Author consider two versions of the Boundary Labeling problem: either the position of the ports on the boundary of $\mathbb{R}$ is fixed and part of the input, or the ports slide, i.e., their exact location is not prescribed. For example, in sliding ports, we can simply fix all ports to the bottom-left corner of their corresponding labels. They [3] obtain the following theorems:

Theorem 2.3.1. *Two- sided boundary labeling with adjacent sides can be solved in $O(n^2)$ time using $O(n)$ space.*

Theorem 2.3.2. *Two- sided boundary labeling with adjacent sides and sliding ports can be solved in $O(n^2)$ time using $O(n)$ space.*

Theorem 2.3.3. *Two- sided boundary labeling with adjacent sides can be solved in $O(n^3 \log n)$ time using $O(n)$ space such that the number of labeled sites is maximized.*

Theorem 2.3.4. *Two- sided boundary labeling with adjacent sides can be solved in $O(n^8 \log n)$ time using $O(n^6)$ space such that the total leader length is minimized.*

Theorem 2.3.5. *Three- sided boundary labeling can be solved in $O(n^4)$ time using $O(n)$ space.*

Theorem 2.3.6. *Three- sided boundary labeling can be solved in $O(n^9)$ time using $O(n)$ space.*

2.4 Multi-Criteria Boundary Labeling

Benkert et al. [2] study labeling a set P of n points in the plane with labels that are aligned to one side of the bounding box $\mathbb{R}$. They prove the following theorem.

Theorem 2.4.1. *Given a set of n points and a set of n labels on the left side of a bounding box $\mathbb{R}$, computing a crossing-free labeling with po-leaders with minimum total length takes $\Theta(n \log n)$ time and $\Theta(n)$ space in the worst case.*

Below we describe their [2] $O(n \log n)$-time algorithm to compute a crossing-free labeling with po-leaders of minimum total length. They provide a sweep line algorithm for boundary labelling problem. The algorithm we discuss here, uses same problem definition, as we defined for the Quadratic time algorithm. (Here, we assume that labels are on the left side of $\mathbb{R}$.)

2.4.1 Preliminaries

Consider the subdivision of the plane into $O(n)$ *strips* by horizontal lines, through the sites and horizontal edges of the label. Let's we denote a strip by 'σ'.

- "p_{a_σ}" be number of sites above σ (including site at top edge of σ).
- "p_{b_σ}" be number of sites below σ (including site at bottom edge of σ).
- "l_{a_σ}" be number of label above σ (including label intersects with σ).
- "l_{b_σ}" be number of labels below σ (including label intersects with σ).

Strip σ categories as follows:

- *downward* strip, if $p_{a_\sigma} > l_{a_\sigma}$.
- *upward* strip, if $p_{b_\sigma} > l_{b_\sigma}$.
- *neutral* strip, if $(p_{a_\sigma} = l_{a_\sigma})$ and/or $(p_{b_\sigma} = l_{b_\sigma})$.

Maximal set of consecutive upward strips is *upward set*. Similar way, we can define *downward set* and *neutral set*.

2.4.2 Algorithm

- **Input** : Set of site P, set of labels $\mathbb{L}$ and rectange $\mathbb{R}$.
- **Output** : Set of crossing free minimum length po-leaders leaders C.

Algorithm 2: Sweep Line Algorithm for Leader Length minimization

---

Subdivide plane $\mathbb{R}$ into horizontal strips, and store it on array A in increasing fashion (with respect to Y-axis);
Traverse each strip σ on array A, identify it's category and store it on A along with σ;
We use an array B_u to store upward sets of A. In similar way, B_d for downward sets and B_n for neutral sets.
while *there is unvisited upward set in* B_u **do**
    Sweep the horizontal line bottom to top, if we encounters site (while sweeping), store it in waiting list W, and if we encounters label, connect the site (which has minimum X-coordinate in list W) to the label using po-leader;
    Mark the upward set visited;
end
while *there is unvisited downward set in* B_d **do**
    Sweep the horizontal line top to bottom, if we encounters site (while sweeping), store it in waiting list W and, if we encounters label, connect the site (which has minimum X-coordinate in list W) to the label using po-leader;
    Mark the downward set visited;
end
while *there is unvisited neutral set in* B_n **do**
    Sweep the horizontal line top to bottom, if we encounters site connect it with direct leader;
    Mark the neutral set visited;
end

2.4.3 Time and Space Complexity

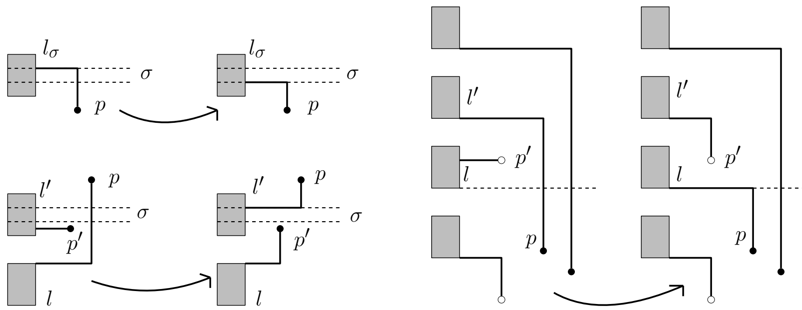
Since, there are $O(n)$ strips, sorting takes $O(n \log n)$ time. It is easy to say that, we can identify the category of each strip of array A in $O(n)$ time. Storing data to array B_u, B_d and B_n will also uses $O(n)$ time. We know that, strip σ can be either upward or downward or neutral. So, strip σ process in exactly one of the three while loops. If, list W uses min heap data structure, then insertion and deletion both will take $O(\log n)$ time. That is, total time while loops uses $O(n \log n)$.

So, over all time complexity is $O(n \log n)$, while space complexity is $O(n)$.

2.4.4 Proof of Correctness

Benkert et al. [2] observe that in any optimal labelling, no leader crosses a neutral strip σ. It can be easily figure out, in figure 2.2(a). So, site belongs to neutral set have direct leader to label. Between any downward strip and upward strip, both $(p_{a_\sigma} - l_{a_\sigma})$ and $(p_{b_\sigma} - l_{b_\sigma})$ differ by at least two. When going from a strip to adjacent strip, the value of each of these expression changes by at most one. Hence downward strip and upward strip always separated by neutral strip. It follows that in any optimal labeling, the points in any upward (or downward) set S must be labeled by leaders that lie entirely within S.

Consider an upward set S. Suppose σ be bottom most strip in S and strip below σ is β. Note that strip β is a neutral strip. It is clear that $(p_{b_\beta} \leq l_{b_\beta})$ while, strip σ have $(p_{b_\sigma} > l_{b_\sigma})$. Hence, $(p_{b_\beta} = l_{b_\beta})$, this implies that, first event must be a site p in the upward set. It is possible that, β and σ may intersect at label l, but site p can't connect to label l, because $(p_{b_\beta} = l_{b_\beta})$ and no leader can cross strip β. So we must label all sites in (and on the boundary of) S with labels that lie entirely



(a) Eliminating intersections with neutral strips

(b) The black points are in W when the sweep line reaches l.

Figure 2.2: Illustration to proof of correctness, for Sweep line algorithm

above σ. Now, it is easy to conclude from figure 2.2(b), that algorithm gives us the optimal labelling.

In [2], Benkert et al. also prove the following theorem:

Theorem 2.4.2. *Assume we are given a set of n points P, a set of n labels on the left side of a bounding box $\mathbb{R}$, and a badness function bad() such that we can determine, in $O(n)$ time, the badness and the location of an optimal po-leader to a given point with its arm in a given height interval (independent of the location of other leaders). We can compute a crossing-free labeling with po-leaders for P with minimum total badness in $O(n^3)$ time and $O(n^2)$ space.*

Benkert et al.[2] also consider the case for two-sided crossing-free labeling and obtain the following result.

Theorem 2.4.3. *Assume we are given a set of n points P, a set of n labels on the left side of a bounding box $\mathbb{R}$, and a badness function bad() such that we can determine, in $O(n)$ time, the badness and the location of an optimal po-leader to a given point with its arm in a given height interval (independent of the location* *of other leaders). Then a two-sided crossing-free labeling with po-leaders can be computed with minimum total badness in $O(n^8)$ time and $O(n^6)$ space.*

Chapter 3

Our Contribution

3.1 Introduction

In this section, we consider the problem defined in Introduction (Chapter 1) under section "Problem definition". Benkert et al.[2] provide a dynamic based solution where labels at one side (in particular left side of $\mathbb{R}$). Our algorithm almost follows the same idea but we introduce a new set called *visibility* V. That is, before connecting site to a label first we need to check whether we have permission to connect it (by leader) or not. Later we consider same length horizontal line segments(or of unit length) rather than points as sites. We say a horizontal line segment h is connected with a label l if there exists a po-leader connecting one of the end point of h with l. Our goal is to find a planar boundary labelling with minimum cost function. After that, we investigate this problem for horizontal line segments with arbitrary length.

3.2 Dynamic Approach for Points as Sites

In this section, we consider sites as point.

3.2.1 Preliminaries

We have a problem instance $(\mathbb{R}, \mathbb{L}, P, V)$, where P is set of n sites contained in $\mathbb{R}$, also no two sites have same X (and Y-coordinate) and there are n labels $\mathbb{L}$(uniform rectangle) touching left side of $\mathbb{R}$. V is a visibility for sites P. We assume that elements of $P = \{p_1, p_2, ..., p_n\}$are arranged according to their increasing order of Y- coordinates. Similarly, $\mathbb{L} = \{l_1, l_2, ..., l_n\}$ also be arranged according to their increasing order of Y- coordinate of bottom-right corner points.

We sub-divide the plane $\mathbb{R}$ into $O(n)$ strips (excluding top most strip) by horizontal lines through the sites and horizontal edges of the label such that part of label does not belongs to strip (say σ_i), if it intersects with bottom most horizontal line of σ_i. We assume that set of labels $\mathbb{L}$ and sites P are arranged in such a way that any horizontal line bounds the boundary of strip, contains either horizontal side (of label) or site, but not both. Let us assume that set of strips $\sigma = \{\sigma_1, \sigma_2, ..., \sigma_m\}$ are arranged such a way that if σ_j, σ_k are two distinct strips in σ where $j < k$ and $\sigma_{j_{t_y}}, \sigma_{k_{t_y}}$ are the Y-coordinate of top most horizontal line of σ_j, σ_k, respectively then $\sigma_{j_{t_y}} < \sigma_{k_{t_y}}$.

We define a set of possible ports for label l_i and strip σ_j as, $S_{ij} = \{s_k : \ s_k \cap l_i \cap \sigma_j \cap \mathbb{R} \neq \phi$ and $s_k \in \mathbb{R},\ l_i \in \mathbb{L},\ \sigma_j \in \sigma\}$. Let $S = \{S_{ij} :\ |S_{ij}| > 0,\ i \in [1, n]\ ,\ j \in [1, m],\ \{i, j\} \in \mathbb{Z}\}$.

We assume that $\mathcal{L} = \{C_1, C_2, C_3, ...\}$ is a feasible solutions of an instance $(\mathbb{R}, \mathbb{L}, P, V)$. It is possible that, cardinality of set $\mathcal{L}$ is infinite.

Now we define cost function of leaders denoted by $cost(leader)$ which takes leader as an input and returns $x \in \mathbb{R}$ as output. We say $cost(leader) = \infty$ if leader not satisfies the visibility V or except it's end point intersects with the site(s). We derive cost of $C_k \in \mathcal{L}$ and optimal labelling for instance $(\mathbb{R}, \mathbb{L}, P, V)$ in equation (3.1) and (3.2), as follows:

$$Cost(C_k) = \sum_{c_{k_1k_2} \in C_k} cost(c_{k_1k_2}) \tag{3.1}$$

$$Opt\_labelling(\mathcal{L}) = \{\, C_a \mid \exists C_a \, \forall C_b,\; Cost(C_a) \leq Cost(C_b),\; and \, \{C_a, C_b\} \in \mathcal{L}\} \tag{3.2}$$

Since $|\mathcal{L}|$ is infinite, it is practically impossible to get $Opt\_labelling(\mathcal{L})$. So it means that some how we need a set of labelling $\mathcal{L}_f \subseteq \mathcal{L}$, where, $Opt\_labelling(\mathcal{L}) \cap \mathcal{L}_f \neq \phi$ and $|\mathcal{L}_f| < \infty$.

Consider $Q_{ijk} = \{q_{i'_1j'_1k'_1}, q_{i'_1j'_1k'_2}, \ldots, q_{i'_aj'_bk'_c}, \ldots\}$ is the set of po-leaders, where $q_{i'_aj'_bk'_c} \in Q_{ijk}$ if and only if $q_{i'_aj'_bk'_c}$ starts at site p_i, and ends label l_j at port $s_{k'}$ where $s_{k'} \in S_{jk}$ and $S_{jk} \in S$. Let $Q = \{Q_{ijk} \;:\; |Q_{ijk}| \geq 1,\; i, j \in [1, n],\; k \in [1, m],\; and\; \{i, j, k\} \in \mathbb{Z}\}$. We denote a optimal leader of Q_{ijk} by $q_{i'_{a_o}j'_{b_o}k'_{c_o}}$, if it belongs to set $T = \{t : cost(t) \leq cost(q_{i'_aj'_bk'_c}),$ and $t < \infty$ and $t \in Q_{ijk},\; \forall\, q_{q_{i'_aj'_bk'_c}} \in Q_{ijk}\}$. Let $Q_{opt\_inf} = \{q_{i'_{a_o}j'_{b_o}k'_{c_o}} \;:\; q_{i'_{a_o}j'_{b_o}k'_{c_o}} \in Q_{ijk}, Q_{ijk} \in Q,\; i, j \in [1, n],\; k \in [1, m],\; and\; \{i, j, k\} \in \mathbb{Z}\}$. It may possible that, $|Q_{opt\_inf}| = \infty$. To make it finite, we allow exactly one optimal leader from each set of leaders Q_{ijk} if it contains optimal leader. We call this new set is Q_{opt}. Clearly, Q_{opt} is finite.

To get set of labelling $\mathcal{L}_f$, we use the concept of strips. We know that, any site always at the boundary of some strip. Suppose, $Q_{ijk} \in Q$ is a set of leaders. If $|Q_{ijk}| = 1$, then only one leader (say l_1, if $cost(l_1) < \infty$, then $l_1 = q_{i'_{a_o} j'_{b_o} k'_{c_o}}$, else l_1 never be in any optimal labelling) is possible which starts at site $p_i \in P$, ends label $l_j \in \mathbb{L}$ at port $s_{j'} \in S_{jk}$. If $C_{opt} \in \mathcal{L}$ is a optimal labelling and have leader from site p_i, to label l_j intersects at strip σ_k, then $q_{i'_{a_o} j'_{b_o} k'_{c_o}} \in C_{opt}$ (because there is only one choice of leader) and we done. So we assume that, $|Q_{ijk}| > 1$ and $\{q_{i'_{a_o} j'_{b_o} k'_{c_o}}, q_{i'_a j'_b k'_c}, ...\} \in Q_{ijk}$. Let us assume, we have set of leaders W for sites P before joining $q_{i'_{a_o} j'_{b_o} k'_{c_o}}$ or $q_{i'_a j'_b k'_c}$, such that for any leader $l \in W$, then l does not intersect with any leader used so far and also does not intersect to any site as well. It is obvious that, we can use atmost one leader of Q_{ijk}, if problem instance $(\mathbb{R}, \mathbb{L}, P, V)$ is solvable (or feasible). So if we use leader $q_{i'_{a_o} j'_{b_o} k'_{c_o}}$, and after joining it W changes to W_1. But if, we use $q_{i'_a j'_b k'_c}$ instead of $q_{i'_{a_o} j'_{b_o} k'_{c_o}}$, then after joining it, W changes to W_2. Since, ports of both the leader are in same strip σ_k, so $W_1 = W_2$. Implies that if, we have optimal labelling $C_{opt} \in \mathcal{L}$, such that, $Q_{ijk} \cap C_{opt} = q_{i'_a j'_b k'_c}$. Then either $q_{i'_a j'_b k'_c} = q_{i'_{a_o} j'_{b_o} k'_{c_o}}$, or $cost(q_{i'_a j'_b k'_c}) = cost(q_{i'_{a_o} j'_{b_o} k'_{c_o}})$. In 'either' case it is obvious and in 'or' case, we can replace $q_{i'_a j'_b k'_c}$ to $q_{i'_{a_o} j'_{b_o} k'_{c_o}}$ without increasing the $Cost(C_{opt})$. That is, $\mathcal{L}_f = \{\mathcal{L}' : \mathcal{L}' \subseteq Q_{opt}$ & $|\mathcal{L}'| = n$ where $\mathcal{L}'$is a feasible solution for instance$(\mathbb{R}, \mathbb{L}, P, V)\}$.

So if $(\mathbb{R}, \mathbb{L}, P, V)$ have a feasible solution then $\mathcal{L}_f \cap Opt\_labelling(\mathcal{L}) \neq \phi$. If cardinality of sites, labels and strips is finite, then $|\mathcal{L}_f| < \infty$.

3.2.2 Recurrence Relation

Suppose, σ_α and σ_β are two strips. Sub-plane induced by strips between σ_α and σ_β (including σ_α and σ_β) is $r(\sigma_\alpha, \sigma_\beta)$, where $\{\sigma_\alpha, \sigma_\beta\} \in \sigma$. Assume, $P_{\alpha\beta} \subseteq P$ be a set of unlabelled sites in region $r(\sigma_\alpha, \sigma_\beta)$ and $\mathbb{L}_{\alpha\beta}$ be a set of unlabelled labels in $r(\sigma_\alpha, \sigma_\beta)$, (that is, $\mathbb{L}_{\alpha\beta} = \{l_j \mid (l_j \in \mathbb{L}) \& \ (l_j \cap r(\sigma_\alpha, \sigma_\beta) = l_j)\}$). If $\alpha > \beta$, then we assume $|P_{\alpha\beta}| = |\mathbb{L}_{\alpha\beta}| = 0$.

Let, $p_r = (p_{r_x}, p_{r_y})$ be the right most site in region $r(\sigma_\alpha, \sigma_\beta)$ to label. Consider, set of leaders Q_{rjk} where $Q_{rjk} \in Q$, and $(\alpha \leq k \leq \beta)$. Let, $q_{r'_{a_o} j'_{b_o} k'_{c_o}} \in Q_{rjk}$ be the optimal leader, joins the site p_r to the label l_j at port $s'_k \in S_{jk}$. It is clear that, no site belongs to above σ_k and to the left of p_r can join the label below l_j and vice versa. Implies, after joining leader $q_{i'_{a_o} j'_{b_o} k'_{c_o}}$, we can sub-divide the problem in two regions $r(\sigma_\alpha, \sigma_{k-1})$ and $r(\sigma_{k+1}, \sigma_\beta)$ where $\sigma_\alpha \leq \sigma_{k-1}$ and $\sigma_{k+1} \leq \sigma_\beta$. Let us suppose, $\mathbb{R}_{p_{r_x}} = [0, p_{r_x}) \times [0, H]$.

For region $r(\sigma_\alpha, \sigma_\beta)$, we call an strip σ_k, a *feasible strip* for right most unlabelled site p_r, only if, $|\mathbb{L}_{\alpha(k-1)}| = |P_{\alpha(k-1)} \cap \mathbb{R}_{p_{r_x}}|$ and $|\mathbb{L}_{(k+1)\beta}| = |P_{(k+1)\beta} \cap \mathbb{R}_{p_{r_x}}|$, and $q_{r'_{a_o} j'_{b_o} k'_{c_o}} \in Q_{rjk}$, $Q_{rjk} \in Q$, $\{r, j\} \in [1, n]$, $\{r, j\} \in \mathbb{Z}$ and $\alpha < \beta$. Let us assume, $F_{\alpha\beta_r}$ be the set which have all feasible strips for region $r(\sigma_\alpha, \sigma_\beta)$ corresponds to rightmost unlabelled site p_r in $r(\sigma_\alpha, \sigma_\beta)$. Since we have $\beta - \alpha + 1$ strips, so $|F_{\alpha\beta_r}| \leq (\beta - \alpha + 1)$.

$$opt\_labelling(\alpha, \beta) = \begin{cases} \min\limits_{\sigma_k \in F_{\alpha\beta_r}} \{cost(q_{r'_{a_o} j'_{b_o} k'_{c_o}}) + opt\_labelling(\alpha, k-1) + \\ \qquad opt\_labelling(k+1, \beta)\} & \text{if } F_{\alpha\beta_r} \neq \phi \\ \infty, & \text{if } F_{\alpha\beta_r} = \phi. \\ \text{check all possible labelling and return} \\ \text{the optimum labelling, if no labelling} \\ \text{return } \infty, & \text{if } |\alpha - \beta| \leq 1 \end{cases} \quad (3.3)$$

Now, we call for $opt\_labelling(1, m)$ for problem instance $(\mathbb{R}, \mathbb{L}, P, V)$. If it returns value $x < \infty$, then $(\mathbb{R}, \mathbb{L}, P, V)$ is *solvable (or feasible)*, and we get a cost of minimum optimal labelling. But if $x = \infty$, then $(\mathbb{R}, \mathbb{L}, P, V)$ is *not solvable (or infeasible)*.

Proof of Correctness :

Base Condition : Since, we apply brute force if $r(\alpha, \beta)$ have at most two strips. That is, we have optimal labelling for $r(\alpha, \beta)$.

Induction Hypothesis : Suppose, for all smaller problems, we have optimal labelling.

Since, recurrence in equation (3.3), takes care of all feasible strips in $r(1, m)$ and their associated sub problems. And, we takes the minimum among all. So, we have optimal labelling for region $r(1, m)$.

3.2.3 Dynamic Algorithm

We assume that, calculation of a optimal leader on set of leaders $Q_{ijk} \in Q$ and cost of optimal leader both takes $O(n)$ time. An strip $\sigma_k \in \sigma$ intersects at at most one label. So we have atmost $O(n \times m)$ optimal leaders. A strip σ_k, contains information of, site p_i (if p_i intersects with σ_k) (and / or) range of S_{jk} (if $(|S_{jk}| > 0)$).

- **Input** : $(\mathbb{R}, \mathbb{L}, P, V)$, where $\mathbb{R}$ is a rectangle, P is a set of n sites inside $\mathbb{R}$ and $\mathbb{L}$ is a set of n labels to the right side r_2 of $\mathbb{R}$

- **Output** : minimum cost po-leaders of planar leader label placement.

Algorithm 3: Recursive Algorithm to calculate optimal labelling

OPT_LABELLING-initialization()

Sub divide the plane into $O(n)$ strips (as we defined earlier) and store strips on array $A[m]$ in increasing order (with respect to Y-coordinate of top most horizontal line of strip), along with strips in $A[m]$, store the Y-coordinate of top most and bottom most horizontal line of strip;

An Array $B[m][m]$ is used to store the sub-problems, such that, $B[i][j]$ stores the cost of optimal labelling for $r(\sigma_i, \sigma_j)$, initially, $B[i][j] = \phi$, $\forall \{i, j\} \in [1, m]$;

We use 2-D range search tree *T* to store the sites and pointer to their corresponding strips, first level of *T* arranged with respect to Y-axis of sites, while auxiliary trees are arranged with respect to X-axis (of sites);

An array $C[n][m]$ is used to store the leaders and it's corresponding intersected leaders;

Let us assume, initially, an array $D[n][n]$ stores the visibility V (that is, if $D[i][j] = 1$, site p_i can join the label l_j(by po-leader), otherwise, not);

for $(i = 1; i \leq n; i++$ **do**
    for $(k = 1; k \leq m; k++)$ **do**
        if *S_{jk} stored at $A[k]$ and $D[i][j] = 1$* **then**
            Compute optimal leader $q_{i'_{ao} j'_{bo} k'_{co}} \in Q_{ijk}$ and cost $cost(q_{i'_{ao} j'_{bo} k'_{co}})$;
            Store port s_k of leader $q'_{i'_{ao} j'_{bo} k_{co}}$ in $C[i][k]$;
            And also $cost(q_{i'_{ao} j'_{bo} k'_{co}})$ at $C[i][k]$;
        else
            Store cost of leader, ∞, at $C[i][k]$;
            And port $s_k = \phi$, at $C[i][k]$;
        end
    end
end

$Cost\_of\_optimal\_labelling$ = OPT-labelling$(A, B, C, D, T, 1, m)$

```
OPT-labelling(A, B, C, D, T, i, j)
if i ≤ j then
    if B[i][j] = φ then
        if |i − j| ≤ 1 then
            if there is a feasible solution then
                Check for all possible solution;
                Store value in B[i][j] of minimum cost labelling among all
                  possible labelling;
                return cost of minimum among all possible labelling;
            else
                Store value '∞' in B[i][j];
                return (∞);
            end
        else
            B[i][j] = ∞;
            FIND-B[i][j](A, B, C, D, T, i, j);
        end
    else
        return value of minimum optimal labelling stored at B[i][j];
    end
else
    return 0;
end
```

3.2.4 Proof of Correctness for Dynamic Algorithm

Since, we are checking manually for small problems in our algorithm so it is obvious that result will be optimal for them.

We assumed that, for the set Q_{ijk}, we get optimal leader in $O(n)$ time. In initialization phase, since array D stores the visibility set V. So in for loop, along with strip σ_k stores any label or not, we are also checking weather we have permission to connect label l_j with site p_i or not. So, in this way we satisfies the

FIND-B[i][j] (A, B, C, D, T, i, j)

Let us suppose Y-coordinate of bottom most horizontal line of strip stored at $A[i]$ is $\sigma_{i_{b_y}}$ and top most horizontal line of strip stored at $A[j]$ is $\sigma_{i_{t_y}}$;

Perform range query on $[\sigma_{i_{b_y}}, \sigma_{i_{t_y}}] \times [0, W]$ on Range Tree *T* and store on some temporary array $E[j-i+1]$ starting from position $E[1]$ (We assume that sites reported by query in increasing order with respect to X-axis);

Check for site (which is not labelled yet) in $E[(j-i+1)]$ which have maximum value of X-coordinate (Assume, we get site $p_r = (p_{r_x}, p_{r_y})$);

Let, n_q be the number of sites reported by range query which have X-coordinate less than p_{r_x} and l_q be the number of unlabelled label in $r(\sigma_i, \sigma_j)$. Assume, σ_t be a strip in between σ_i and σ_j, $n_{q_{tb}}$ be the number of sites reported by query in $r(i, t-1)$, and $l_{q_{tb}}$ be the number of unlabelled labels in $r(\sigma_i, \sigma_{t-1})$. And $n_{q_{ta}}$ be the number of sites reported by query in $r(\sigma_{t+1}, j)$, and $l_{q_{ta}}$ be the number of unlabelled labels in $r(\sigma_t, \sigma_j)$;

for $t = 1, t \leq j;\ t++$ **do**

  if $C[r][t] < \infty$ *and satisfy equality of number of unlabelled sites and labels in both separated regions by strip* σ_t **then**

    $min\_opt = C[r][t] +$ OPT-labelling(*A, B, C, D, T, i, t-1*) $+$ OPT-labelling(*A, B, C, D, T, t + 1, j*);

    if $min\_opt < \infty$ **then**

      if $min\_opt < B[i][j]$ **then**

        $B[i][j] = min\_opt$;

      end

    end

  end

  if $A[t]$ *stores site whose X-coordinate less than* p_{r_x} **then**

    $n_{q_{ta}} = n_{q_{ta}} - 1$ and $n_{q_{tb}} = n_{q_{tb}} + 1$;

  end

  if *strip at* $A[t]$ *have unlabelled label and* $A[t+1]$ *have no label* **then**

    $l_{q_{tb}} = l_{q_{tb}} + 1$;

  end

  if *strip at* $A[t]$ *have no label and* $A[t+1]$ *have unlabelled label* **then**

    $l_{q_{ta}} = l_{q_{ta}} - 1$;

  end

end

visibility V.

Since we are looking for right most site (which is not labelled by any leader) By performing range query in range $[\sigma_{i_{b_y}}, \sigma_{i_{t_y}}] \times [0, W]$ we get all the points in increasing order (with respect to X-axis) for the region $r(\sigma_i, \sigma_j)$. By traversing reported sites by query we get the correct site which is right most and not labelled by any label. Now, we have right most site $p_r = (p_{r_x}, p_{r_y})$ in region $r(\sigma_i, \sigma_j)$.

We calculate number of sites in region $r(\sigma_i, \sigma_j)$ whose X- coordinate less than p_{r_x}. These sites must be unlabelled because we are labelling from right to left. And since we are labelling the right most unlabelled site P_r, so site which left unlabelled must be to the left of right most site p_r. By traversing label from position i to j in array A we get number of unlabelled labels.

Since we are traversing strips bottom to top one by one. So, number of unlabelled site reduce by atmost 1 on switching from $r(\sigma_{t+1}, \sigma_j)$ to $r(\sigma_{t+2}, \sigma_j)$ (because a strip contains atmost one site), while on switching region from $r(\sigma_i, \sigma_{t-1})$ to $r(\sigma_i, \sigma_t)$ number of unlabelled sites increase by atmost 1. If $A[t]$ stores site whose X- coordinate less than p_{r_x} the number of unlabelled site in $r(\sigma_{t+2}, \sigma_j)$ one less than $r(\sigma_{t+1}, \sigma_j)$ and number of unlabelled sites in $r(\sigma_i, \sigma_t)$ one more than unlabelled sites in $r(\sigma_i, \sigma_{t-1})$, else number of unlabelled sites remain same.

It is not possible that both strips at $A[t+1]$ and $A[t]$ stores different labels, because $A[t+1]$ and $A[t]$ stores consecutive strips and no two consecutive strip can store two different labels because no two label overlaps.

If strips at $A[t+1]$ have unlabelled label and $A[t]$ have no label, then label in region $r(\sigma_i, \sigma_t)$ and $r(\sigma_i, \sigma_{t-1})$ remain same, but numbers of labels in $r(\sigma_{t+1}, \sigma_j)$ one less than $r(\sigma_t, \sigma_j)$.

If strips at $A[t+1]$ have no labels and $A[t]$ have unlabelled label, then labels in region $r(\sigma_i, \sigma_t)$ have one more than labels in $r(\sigma_i, \sigma_{t-1})$, but numbers of labels in $r(\sigma_{t+1}, \sigma_j)$ and $r(\sigma_t, \sigma_j)$ remain same.

If strips at $A[t+1]$ and $A[t]$ have same label, then labels in region $r(\sigma_i, \sigma_t)$ and $r(\sigma_i, \sigma_{t-1})$ reamain same, and also labels in $r(\sigma_{t+1}, \sigma_j)$ and $r(\sigma_t, \sigma_j)$ will also remain same.

for loop in function '$find - B[i][j]'$ runs for all strips in between σ_i and σ_j(including σ_i and σ_j one by one. 'If' condition checks weather we can join leader starts from site p_r to label at strip σ_t, along with that it also checks equality of number of unlabelled sites left of p_{r_x} and number of unlabelled labels in $r(\sigma_i, \sigma_{t-1})$ as well $r(\sigma_{t+1}, \sigma_j)$[1]. If condition satisfies, algorithm adds the cost of optimal leader starts at site p_r and ends at label l'_t (where l'_t stores at σ_t) and optimal cost of sub- problems $r(\sigma_i, \sigma_{t-1})$ and $r(\sigma_{t+1}, \sigma_j)$. Since, p_r is the right most site which is to be labelled, then no unlabelled site have X- coordinate less than p_{r_x} in $r(\sigma_i, \sigma_{t-1})$ can join the label belongs to $r(\sigma_{t+1}, \sigma_j)$ and vice versa. So, by checking optimality in region $r(\sigma_i, \sigma_{t-1})$ and $r(\sigma_i, \sigma_{t-1})$ separately will not make any difference. Since, for loop iterates over all all strips between σ_i to σ_j (including σ_i, σ_j) and picking the minimum among all we have optimal solution.

Since, $r(\sigma_i, \sigma_j)$ algorithm checks for all possible choices of leader from strip σ_i to σ_j and maintains the minimalitiy at $B[i][j]$. So, if $B[i][j] = \infty$ then there is no feasible solution possible for $r(\sigma_i, \sigma_j)$, else we have optimal solution with respect to cost function we have.

[1]if number of unlabelled site and unlabelled labels are different, then it is obvious that there will not exist any feasible solution. but if number of sites and labels are equal then there is a possibility for feasibility.

3.2.5 Time and Space Complexity

Sub-division of plane into $O(n)$ strips takes $O(n)$ time. Storing them on Array A in increasing order takes $O(n \log n)$ time. Since, each strip stores atmost one label (and/ or site). So, size of A is $O(n)$. We have atmost m^2 sub-problems (because, rightmost unlabelled site connects to any label divides the problem in two sub problems). We use $2D$-Array B to store the sub-problem results to reuse. We have to store atmost m^2 sub-problems, so B is of m^2 (that is, $O(n^2)$). Initialization of B takes $O(n^2)$ time. We have n is the number of sites, so range tree T, construction time for $2D$-plane is $O(n \log n)$ and space complexity is to store range search tree T is $O(n \log n)$. Array $C[n][m]$ is used to store optimal leader of Q_{ijk} if $|Q_{ijk}| > 0$. Since, we assume that, optimal leader in Q_{ijk} can be computed in $O(n)$ time. So, calculate array C (that is, all possible optimal leader) time uses $O(n^2m)$.

Small problem takes $O(1)$ time. So, we look for 'FIND-B[i][j]' function. Range Query on tree T takes $(\log n + k')$ time, where k' are the number of points reported by query. Checking right most site (which is not labelled) on points reported by range query will take $O(n)$ time. Calculation of number of unlabelled labels in region $r(\sigma_i, \sigma_j)$ will take $O(j - i + 1)$ time. For loop runs $(j - i + 1)$ time while in each iteration, it takes $O(1)$ time. So total time taken by for loop is $O(j - 1 + 1)$.

That is, initialization phase takes $O(n^3)$ time. Since, we have m^2 (m is number of strips) sub-problems and each sub- problem takes $O(n)$ time (as we seen above). So, total time complexity is $O(n^3)$, While space complexity is $O(n^2)$.

3.3 Recurrence Relation for Horizontal Line Segments as sites

In this section, we consider horizontal line segments as sites.

3.3.1 Preliminaries

We have an problem instance $(\mathbb{R}, \mathbb{L}, P, V)$, where P is set of n sites contained in R, also no two sites have same Y-coordinate (and endpoints of them have same X- coordinate). There are n labels $\mathbb{L}$ (uniform rectangle) touching left side of $\mathbb{R}$. V is a visibility for sites P. We assume that only end point of line segment can be used to connect label. Let $P = \{p_1, p_2, ..., p_n\}$, are arranged according to their increasing order of Y- coordinates. Similarly, $\mathbb{L} = \{l_1, l_2, ..., l_n\}$ also be arranged according to their increasing order of Y- coordinate of bottom-right corner points.

We sub-divide the plane $\mathbb{R}$ into $O(n)$ strips (excluding top most strip) by horizontal lines through the sites and horizontal edges of the label such that part of label does not belongs to strip (say σ_i), if it intersects with bottom most horizontal line of σ_i. We assume that, set of labels $\mathbb{L}$ and sites P arranges, such that, any horizontal line bounds the boundary of strip, contains either horizontal side (of label), or site, but not both. Let us assume that set of strips $\sigma_h = \{\sigma_1, \sigma_2, ..., \sigma_m\}$ are arranged such a way that if σ_j, σ_k are two distinct strips in σ where $j < k$ and $\sigma_{j_{t_y}}, \sigma_{k_{t_y}}$ are the Y-coordinate of top most horizontal line of σ_j, σ_k, respectively then $\sigma_{j_{t_y}} < \sigma_{k_{t_y}}$.

We define a set of possible ports for label l_i and strip σ_j as, $S_{ij} = \{s_k : \ s_k \cap l_i \cap \sigma_j \cap \mathbb{R} \neq \phi$ and $s_k \in \mathbb{R}, \ l_i \in \mathbb{L}, \ \sigma_j \in \sigma\}$. Let $S = \{S_{ij} : \ |S_{ij}| > 0, \ i \in [1, n] \ , \ j \in [1, m], \ \{i, j\} \in \mathbb{Z}\}$.

We assume that $\mathcal{L} = \{C_1, C_2, C_3, ...\}$ is a feasible solutions of an instance $(\mathbb{R}, \mathbb{L}, P, V)$. It is possible that, cardinality of set $\mathcal{L}$ is infinite.

Now we define cost function of leaders, denoted by $cost(leader)$, which takes leader as an input and returns $x \in \mathbb{R}$ as output. We say $cost(leader) = \infty$, if leader not satisfies the visibility V or except it's endpoint it intersect with the site(s). We derive cost of C_k and optimal labelling for instance $(\mathbb{R}, \mathbb{L}, P, V)$ in equation (3.1) and (3.2), as follows:

$$Cost(C_k) = \sum_{c_{k_1 k_2} \in C_k} cost(c_{k_1 k_2}) \tag{3.4}$$

$$Opt\_labelling(\mathcal{L}) = \{\, C_a \mid \exists C_a \, \forall C_b,\, Cost(C_a) \leq Cost(C_b),\, and \,\{C_a, C_b\} \in \mathcal{L}\} \tag{3.5}$$

Since, $|\mathcal{L}|$ is infinite, it is practically impossible to get $Opt\_labelling(\mathcal{L})$. So it means that some how we need a set of labelling $\mathcal{L}_f \subseteq \mathcal{L}$, where, $Opt\_labelling(\mathcal{L})$ $\cap\, \mathcal{L}_f \neq \phi$ and $|\mathcal{L}_f| < \infty$.

Let suppose, $p_i \in P$ be a line segment (or site) and p_{i_L} is a left end point in p_i while p_{i_R} is right end point in p_i. Consider $Q_{i_L jk} = \{q_{i'_{1_L} j'_1{}' k'_1}, q_{i'_{2_L} j'_1 k'_1}, \ldots, q_{i'_{a_L} j'_b k'_c},$ $\ldots\}$ is the set of po-leaders, where $q_{i'_{a_L} j'_b k'_c} \in Q_{i_L jk}$ if and only if $q_{i'_{a_L} b'_j c'_k}$ starts at left end point p_{i_L} of site (or line segment) p_i, and ends label l_j at port s_{k_1} where $s_{k_1} \in S_{jk}$ and $S_{jk} \in S$. In the same way we can define $Q_{i_R jk}$ with respect to right end point p_{i_R} of p_i . Let assume that $Q_L = \{Q_{i_L jk} \;:\; |Q_{i_L jk}| \geq 1,\; i, j \in [1, n],\; k \in [1, m],\; and\; \{i, j, k\} \in \mathbb{Z}\}$ and similarily Q_R is defined also in the same way as Q_L but with respect to right end points of line segments (or sites). We denote a optimal leader of $Q_{i_L jk}$ by $q_{i'_{a_{oL}} j'_{b_o} k'_{c_o}}$, if it it belongs to set $T = \{t : cost(t) \leq cost(q_{i'_{a_L} j'_b k'_c}),$ and $(cost(t) < \infty),$ and $t \in Q_{i_L jk},\ \forall\ q_{q_{i'_{a_L} j'_b k'_c}} \in Q_{i_L jk}\}$. Same way optimal leader for $Q_{i_R jk}$ is $q_{i'_{a_{oR}} j'_{b_o} k'_{c_o}}$. Let us assume, $Q_{opt\_inf_L} = \{q_{i'_{a_{oL}} j'_{b_o} k'_{c_o}} : q_{i'_{a_{oL}} j'_{b_o} k'_{c_o}} \in Q_{i_L jk}, Q_{i_L jk} \in Q_L,\ i, j \in [1, n],\ k \in [1, m],\ and\ \{i, j, k\} \in \mathbb{Z}\}$. It may possible that, $|Q_{opt\_inf_L}| = \infty$. To make it finite, we allow exactly one optimal leader from each set of leaders $Q_{i_L jk}$ if it contains optimal leader. We call this new set is Q_{opt_L}. Clearly, Q_{opt_L} is finite. Same way we can define Q_{opt_R}, but with respect to right end points of sites P. We say, $Q = Q_L \cup Q_R$ We say, $Q_{opt} = Q_{opt_L} \cup Q_{opt_R}$.

To get set of labelling $\mathcal{L}_f$, we use the concept of strips. As we seen earlier in section (3.2.1).that, $\mathcal{L}_f = \{\mathcal{L}' : \mathcal{L}' \subseteq Q_{opt}\ \&\ |\mathcal{L}'| = n$ and $\mathcal{L}'$ is a feasible solution of instance$(\mathbb{R}, \mathbb{L}, P, V)\ \}$.

So if $(\mathbb{R}, \mathbb{L}, P, V)$ have a feasible solution then $\mathcal{L}_f \cap Opt\_labelling(\mathcal{L}) \neq \phi$. If cardinality of sites, labels and strips is finite, then $|\mathcal{L}_f| < \infty$.

3.3.2 Recurrence Relation for Unit Length Horizontal Line Segments

Here we assume sites are Unit Length Horizontal Line Segments. Suppose, σ_α and $\sigma_\beta, (\alpha \leq \beta)$ are two strips, and sub-plane induced by strips between σ_α and σ_β (including σ_α and σ_β) is $r(\sigma_\alpha, \sigma_\beta)$, where $\{\sigma_\alpha, \sigma_\beta\} \in \sigma$. Assume, $P_{\alpha\beta} \subseteq P$ be a set of unlabelled sites in region $r(\sigma_\alpha, \sigma_\beta)$ and $\mathbb{L}_{\alpha\beta}$ be a set of unlabelled labels in $r(\sigma_\alpha, \sigma_\beta)$, (that is, $\mathbb{L}_{\alpha\beta} = \{l_j \mid (l_j \in \mathbb{L})\&\ (l_j \cap r(\sigma_\alpha, \sigma_\beta) = l_j)\}$). If $\alpha > \beta$, then we assume $|P_{\alpha\beta}| = |\mathbb{L}_{\alpha\beta}| = 0$. Let, p_r be the right most site in region $r(\sigma_\alpha, \sigma_\beta)$ to label. Assume left end points of site p_r is $p_{r_L} = (p_{r_{Lx}}, p_{r_{Ly}})$ and right end point of site p_r is $p_{r_R} = (p_{r_{Rx}}, p_{R_{Ry}})$. Consider, set of leaders $Q_{r_L jk}$ where $Q_{r_L jk} \in Q_L$ and $\alpha \leq k \leq \beta$. Let, $q_{r'_{a_{oL}} j'_{b_o} k'_{c_o}} \in Q_{r_L jk}$ be the optimal leader, joins the left end point of site p_r to the label l_j at port $s_{k_1} \in S_{jk}$ and site p_r is on the strip σ_t and $q_{r'_{a_{oR}} j'_{b_o} k'_{c_o}} \in Q_{r_R jk}$ be the optimal leader, joins right end point of site p_r to the label l_j at port $s_{k_2} \in S_{jk}$ and site p_r is on the strip σ_t.

Suppose $\mathbb{R}'_{p_{r_{Lx}}} = (p_{r_{Lx}}, p_{r_{Rx}}] \times [0, H]$. Since, as we know p_r is the right most site to label and sites as unit length horizontal line segments[2]. Then $(\mathbb{R}'_{p_{r_{Lx}}} \cap r(\sigma_{t+1}, \sigma_{k-1}) \cap P) = \phi$. That is, if we join site p_r to label l_j in strip σ_t using leader $q_{r'_{a_{oL}} j'_{b_o} k'_{c_o}}$. Then, no unlabelled site below strip σ_k and to the left of site p_r (with respect to the right end point of site p_r), can join the label above σ_k (ignoring the label which intersects with σ_k because label at σ_k is labelled with site p_r). The case where right end point of site p_r connects label is obvious. Case if $t > k$ and $t = k$ is similar.

Implies, if we joins leader $q_{i'_{a_{oL}} j'_{b_o} k'_{c_o}}$ (or $q_{i'_{a_{oR}} j'_{b_o} k'_{c_o}}$), we can sub-divide the problem in two regions $r(\sigma_\alpha, \sigma_{k-1})$ and $r(\sigma_{k+1}, \sigma_\beta)$.

Let suppose, $S_{p_{rx}} = [0, p_{r_{Rx}}) \times [0, H]$. We assume site p_r is labelled. For region $r(\sigma_\alpha, \sigma_\beta)$, we call an strip σ_k is a *feasible strip* for right end point p_{r_R} of right most site p_r, only if, $|\mathbb{L}_{\alpha(k-1)}| = |P_{\alpha(k-1)} \cap S_{p_{rx}}|$ and $|\mathbb{L}_{(k+1)\beta}| = |P_{(k+1)\beta} \cap S_{p_{rx}}|$, and $q_{r'_{a_{oR}} j'_{b_o} k'_{c_o}} \in Q_{r_R jk}$, $Q_{r_R jk} \in Q_R$, $\{r, j\} \in [1, n]$, $\{r, j\} \in \mathbb{Z}, \alpha < \beta$. We define, $F_{\alpha\beta_{r_R}}$ be the set which have all feasible strips for region $r(\sigma_\alpha, \sigma_\beta)$ with respect to the right end point p_{r_R} of right most unlabelled site p_r. In the similar way, $F_{\alpha\beta_{r_L}}$ be the set which have all feasible strips for region $r(\sigma_\alpha, \sigma_\beta)$ with respect to the left end point p_{r_L} of right most unlabelled site p_r. We say, $F_{\alpha\beta_r} = F_{\alpha\beta_{r_R}} \cup F_{\alpha\beta_{r_L}}$. Since, $|F_{\alpha\beta_{r_R}}| \leq \beta - \alpha + 1$ and $|F_{\alpha\beta_{r_L}}| \leq \beta - \alpha + 1$, because we have atmost $\beta - \alpha + 1$ strips. So, $|F_{\alpha\beta_r}| \leq 2 \times (\beta - \alpha + 1)$. Let

[2] If horizontal lines are of arbitrary length then it is possible that $(\mathbb{R}'_{p_{r_{Lx}}} \cap r(\sigma_{t+1}, \sigma_{k-1}) \cap P) \neq \phi$, so site below σ_t can also join the label which is above σ_t. Implies that we can't sub-divide the plane in $O(n)$ strips for arbitrary length horizontal line segment.

us assume, $\mathfrak{L}_{\alpha\beta_r} \subseteq Q_{opt}$ be a set of optimal leaders which starts from right most unlabelled site p_r in $r(\sigma_\alpha, \sigma_\beta)$ and ends at each feasible strip of endpoints belongs to p_r. Cardinality of set $|\mathfrak{L}_{\alpha\beta_r}| \leq 2 \times (\beta - \alpha + 1)$.

$$opt\_labelling(\alpha, \beta) = \begin{cases} \min\limits_{q'_{r'_{a_o} j'_{b_o} k_{c_o}} \in \mathfrak{L}_{\alpha\beta_r}} \{cost(q_{r'_{a_o} j'_{b_o} k'_{c_o}}) + \\ \quad opt\_labelling(\alpha, k-1) + \\ \quad\quad opt\_labelling(k+1, \beta)\} & \text{if } \mathfrak{L}_{\alpha\beta_r} \neq \phi \\ \infty, & \text{if } \mathfrak{L}_{\alpha\beta_r} = \phi. \\ \text{check all possible labelling and return} \\ \text{the optimum labelling, if no labelling} \\ \text{return } \infty, & \text{if } |\alpha - \beta| \leq 1 \end{cases} \tag{3.6}$$

Now, we call for $opt\_labelling(1, m)$ for problem instance $(\mathbb{R}, \mathbb{L}, P, V)$. If it returns value $x < \infty$, then $(\mathbb{R}, \mathbb{L}, P, V)$ is *solvable (or feasible)*, and we get a cost of minimum optimal labelling. But if $x = \infty$, then $(\mathbb{R}, \mathbb{L}, P, V)$ is *not solvable (or infeasible)*.

Algorithm and Time Complexity

Algorithm is similar to the case when sites are points and satisfies the visibility V. So, we are omitting the algorithm part for this case.

When joining a right most unit length line segment to label it sub- divide the region into two parts. Since we have $O(n)$ strips, so there will be $O(n^2)$ different sub- problems. By picking a right most unlabelled unit length line segment (as we did for points in section (3.2)) and going through all possible feasible leaders (which is atmost $2 \times (\beta - \alpha + 1)$ for a line segment). To solve a sub- problem we need $O(n)$ time complexity (as we seen for the case when sites are points in the section (3.2)). So, total time complexity will be $O(n^3)$. And space complexity will be $O(n^2)$.

3.3.3 Recurrence Relation for Arbitrary Length Horizontal Line Segments as sites

Here we assume sites are arbitrary length horizontal line segments. Let assume that $\mathcal{V}$ be a set of end points of sites arranged in increasing order according to the X-coordinate. Suppose, σ_α and $\sigma_\beta, (\alpha \leq \beta)$ are two strips and sub-plane induced by strips between σ_α and σ_β (including σ_α and σ_β) is $r(\sigma_\alpha, \sigma_\beta)$, where $\{\sigma_\alpha, \sigma_\beta\} \in \sigma$. Let $v_\gamma = (v_{\gamma_x}, v_{\gamma_y}) \in \mathcal{V}$ and $\mathbb{R}_{v_{\gamma_x}} = (-\infty, v_{\gamma_x}] \times [0, H]$. Then region $r(\sigma_\alpha, \sigma_\beta, v_\gamma) = \mathcal{R}_{v_{\gamma_x}} \cap r(\sigma_\alpha, \sigma_\beta)$.

Let $P_{\alpha\beta\gamma} \subseteq P$ be the set of unlabelled sites in (or intersects) $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$ and $\mathbb{L}_{\alpha\beta\gamma}$ be the set of unlabelled labels in $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$, that is, $\mathbb{L}_{\alpha\beta\gamma} = \{l_j \mid (l_j \in \mathbb{L})\ \&\ (l_j \cap r(\sigma_\alpha, \sigma_\beta, v_\gamma) = l_j)\}$. For $\alpha > \beta$, we assume $|P_{\alpha\beta\gamma}| = |\mathbb{L}_{\alpha\beta\gamma}| = 0$.

Let $v_r \in \mathcal{V}$ be the right most point in region $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$ such that the corresponding site $p_u \in P$ is unlabelled. Now we consider following two possible cases.

- **Case 1 : $\boldsymbol{v_r}$ is the right most end point of $\boldsymbol{p_u}$.** Let $v_r = (v_{r_{Rx}}, v_{r_{Ry}})$. Consider, set of leaders Q_{r_Rjk} where $Q_{r_Rjk} \in Q_R$ and $\alpha \leq k \leq \beta$. Let, $q_{r'_{a_oR} j'_{b_o} k'_{c_o}} \in Q_{r_Rjk}$ be the optimal leader, joins v_r to the label l_j at port $s_{k_1} \in S_{jk}$ and site p_r is on the strip σ_t. Then no endpoint(s) of site(s) in

$r(\sigma_\alpha, \sigma_\beta, v_r)$ and below strip σ_k can join the label which is above σ_k (ignore the label intersects to strip σ_k) and vice versa. So, by adding $q_{r'_{a_oR} j'_{b_o} k'_{c_o}}$ we can subdivide the problem in two region $r(\sigma_\alpha, \sigma_{k-1}, v_r)$ and $r(\sigma_{k+1}, \sigma_\beta, v_r)$. But optimal solution possibly may not have leader which joins v_r to any label. That means, left end point of site p_u must join to some $l_a \in \mathbb{L}$. We have horizontal lines, so left end point of p_u must be in $r(\sigma_\alpha, \sigma_{k-1}, v_r)$. That means if we consider site p_u as unlabelled in $r(\sigma_\alpha, \sigma_\beta, v_{r-1})$, problem can be restricted to $r(\sigma_\alpha, \sigma_\beta, v_{r-1})$, if $r > 1$.

- **Case 2 : v_r is the left most end point of p_u.** In this case we assume, $v_r = (v_{r_{Lx}}, v_{r_{Ly}})$. Consider, set of leaders $Q_{r_L jk}$ where $Q_{r_L jk} \in Q_L$ and $\alpha \leq k \leq \beta$. Let, $q_{r'_{a_oL} j'_{b_o} k'_{c_o}} \in Q_{r_L jk}$ be the optimal leader, joins v_r to the label l_j at port $s_{k_2} \in S_{jk}$ and site p_u is on the strip σ_t. Then no endpoint(s) of site(s) in $r(\sigma_\alpha, \sigma_\beta, v_r)$ and below strip σ_k can join the label which is above σ_k (ignore the label intersects to strip σ_k) and vice versa. So, by adding $q_{r'_{a_oL} j'_{b_o} k'_{c_o}}$ we can subdivide the problem in two region $r(\sigma_\alpha, \sigma_{k-1}, v_r)$ and $r(\sigma_{k+1}, \sigma_\beta, v_r)$. Since site p_u is unlabelled and v_r is the left most end point of p_u then $p_u \notin r(\sigma_\alpha, \sigma_\beta, v_{r-1})$. If we consider site p_u as unlabelled, restrict the problem in $r(\sigma_\alpha, \sigma_\beta, v_{r-1})$, then we never get feasible solution.

  If no optimal leader possible for v_r in $r(\sigma_\alpha, \sigma_\beta, v_r)$, then problem is infeasible.

For the case where, v_r is a right endpoint of p_u then we say a strip σ_k is a *feasible strip* for v_r in the region $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$, only if, $|\mathbb{L}_{\alpha(k-1)r}| = |P_{\alpha(k-1)r}|$ and $|\mathbb{L}_{(k+1)\beta r}| = |P_{(k+1)\beta r}|$, and $q_{r'_{a_oR} j'_{b_o} k'_{c_o}} \in Q_{r_R jk}$, $Q_{r_R jk} \in Q_R$, $\{r, j\} \in [1, n]$, $\{r, j\} \in \mathbb{Z}, \alpha < \beta$. We define, $\mathcal{F}_{\alpha\beta_{\gamma_R}}$ be the set which have all feasible strips for region $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$ with respect to right most end point v_r whose corresponding site p_u is unlabelled and v_r is right end point of p_u. Similar way, for the case where v_r is a left endpoint of p_u, we define $\mathcal{F}_{\alpha\beta_{\gamma_L}}$ be the set which have all feasible strips in the region $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$ with respect to v_r. $|\mathcal{F}_{\alpha\beta_{\gamma_R}}| \leq \beta - \alpha + 1$ and $|\mathcal{F}_{\alpha\beta_{\gamma_L}}| \leq \beta - \alpha + 1$, because we have atmost $\beta - \alpha + 1$ strips. Let us assume, $\mathfrak{L}_{\alpha\beta_{\gamma_R}} \subseteq Q_{opt_R}$ be a set of optimal leaders such that for each leader starts from right most point v_r in $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$ and ends at each feasible strip of v_r if v_r is the right end point of p_u, else $|\mathfrak{L}_{\alpha\beta_{\gamma_R}}| = \phi$. Similar way, $\mathfrak{L}_{\alpha\beta_{\gamma_L}}$ can be defined if v_r is the left end point of p_u. Cardinality of set $|\mathfrak{L}_{\alpha\beta_{\gamma_R}}| \leq (\beta - \alpha + 1)$ and $|\mathfrak{L}_{\alpha\beta_{\gamma_L}}| \leq (\beta - \alpha + 1)$. Let $\mathcal{V}_R \subset \mathcal{V}$ be the set of all right end point of sites P and $\mathcal{V}_L \subset \mathcal{V}$ be the set of all left end point of sites P. Let suppose, $v_{\lambda_{\alpha\beta}}$ be the second most right end point in $r(\sigma_\alpha, \sigma_\beta, v_\gamma)$ where site correspond to it is unlabeled. If there is no such point exist then $v_{\lambda_{\alpha\beta}} = v_\gamma$.

$$
opt\_labelling(\alpha,\beta,\gamma) = \begin{cases}
\min\limits_{q_{r'_{a_o} j'_{b_o} k'_{c_o}} \in \mathfrak{L}_{\alpha\beta_{\gamma_L}}} \{cost(q_{r'_{a_o} j'_{b_o} k'_{c_o}}) + \\
\quad opt\_labelling(\alpha, k-1, \gamma) + \\
\quad\quad opt\_labelling(k+1, \beta, \gamma)\} & \text{if } \mathfrak{L}_{\alpha\beta_{\gamma_L}} \neq \phi \text{ \& } v_\gamma \in V_L \\
\infty, & \text{if } \mathfrak{L}_{\alpha\beta_{\gamma_L}} = \phi \text{ \& } v_\gamma \in \mathcal{V}_L. \\
\min\{ \min\limits_{q_{r'_{a_o} j'_{b_o} k'_{c_o}} \in \mathfrak{L}_{\alpha\beta_{\gamma_R}}} \{cost(q_{r'_{a_o} j'_{b_o} k'_{c_o}}) + \\
\quad opt\_labelling(\alpha, k-1, \gamma) + \\
\quad opt\_labelling(k+1, \beta, \gamma)\}, \\
\quad opt\_labelling(\alpha, \beta, \lambda)\} & \text{if } \mathfrak{L}_{\alpha\beta_{\gamma_R}} \neq \phi, v_\gamma \in V_R \text{\& } v_\gamma \neq v_{\lambda_{\alpha\beta}} \\
\infty, & \text{if } \mathfrak{L}_{\alpha\beta_{\gamma_R}} = \phi,\ v_\gamma \in V_R \\
\min\limits_{q_{r'_{a_o} j'_{b_o} k'_{c_o}} \in \mathfrak{L}_{\alpha\beta_{\gamma_R}}} \{cost(q_{r'_{a_o} j'_{b_o} k'_{c_o}}) + \\
\quad opt\_labelling(\alpha, k-1, \gamma) + \\
\quad opt\_labelling(k+1, \beta, \gamma)\} & \text{if } \mathfrak{L}_{\alpha\beta_{\gamma_R}} \neq \phi, v_\gamma \in V_R \text{\& } v_\gamma = v_{\lambda_{\alpha\beta}} \\
\text{check all possible labelling and return} \\
\text{the optimum labelling, if no labelling} \\
\text{return } \infty, & \text{if } |\alpha - \beta| \leq 1
\end{cases} \tag{3.7}
$$

Now, we call for $opt\_labelling(1, m, 2 \times n)$ for problem instance $(\mathbb{R}, \mathbb{L}, P, V)$. If it returns value $x < \infty$, then $(\mathbb{R}, \mathbb{L}, P, V)$ is *solvable (or feasible)*, and we get a cost of minimum optimal labelling. But if $x = \infty$, then $(\mathbb{R}, \mathbb{L}, P, V)$ is *not solvable (or infeasible)*.

Algorithm and Time Complexity

Algorithm is similar to the case when sites are points and satisfies the visibility V. So, we are omitting the algorithm part for this case.

When joining a right most end point of arbitrary length horizontal line segments to label it sub- divide the region into two parts. Since we have $O(n^3)$ different regions. So we check the time taken by each sub- problem. By arranging the labels and site in increasing order, when we initialize the problem. Traversing top to bottom of the strip and checking equality between number of unlabelled sites and unlabelled label can be done in $O(n)$ time. Ckecking the second right most end point whose corresponding site is unlabelled can be done in $O(n)$ time. Minimality checking also takes $O(n)$ time. So, time taken by to solve a sub- problem is $O(n)$. we are not going through much details because these sub-problem and the sub- problem we solved when we considered sites as points are very similar. So total time complexity to solve for the optimality of problem instance $(\mathbb{R}, \mathbb{L}, P, V)$ will be $O(n^4)$ and space complexity will be $O(n^3)$.

Chapter 4

Conclusion and possible future work

In this report, we have studied algorithms for solving the one-sided as well as multi sided boundary labeling problem with po-leaders. The algorithms either minimize the total leader length or optimize a general badness function. Considering same length horizontal segment rather than points as sites, we obtain some results for one sided boundary labelling problem. Here is a list of interesting future task:

- A polynomial time algorithm to find a minimum length solution of three-sided and four- sided boundary labeling when sites are points.
- Labelling various type of line segments such as different length horizontal segment, axis parallel line segment with same length, general line segment with same length as well as arbitrary length for one sided boundary labelling problem with minimum leader length.
- Labelling various type of line segments for opposite sided boundary labelling problem with minimum leader length.

Bibliography

[1] Michael A Bekos, Michael Kaufmann, Antonios Symvonis, and Alexander Wolff. Boundary labeling: Models and efficient algorithms for rectangular maps. *Computational Geometry*, 36(3):215–236, 2007.

[2] Marc Benkert, Herman J Haverkort, Moritz Kroll, and Martin Nöllenburg. Algorithms for multi-criteria boundary labeling. *J. Graph Algorithms Appl.*, 13(3):289–317, 2009.

[3] Philipp Kindermann, Benjamin Niedermann, Ignaz Rutter, Marcus Schaefer, André Schulz, and Alexander Wolff. Multi-sided boundary labeling. *Algorithmica*, 76(1):225–258, Sep 2016.